BMECLEARING

MEFFGate Clearing

FIX INTERFACE SPECIFICATIONS

Version C1.8

4 October 2018

Changes made in the latest revision

Outlined below are the main changes made in the version C1.8 (since the version C1.7 on 18 May 2018):

- Margins calculation portfolio:
    - New chapter "14 - Margins calculation portfolio"
    - New messages: Margin Requirement Inquiry and Margin Requirement Report

Contents

1. Introduction

1.1 Scope of this manual

This document contains the definition of the interface provided by MEFF for developing applications related with the clearing scope. The interface is based on version 4.4 of the FIX Protocol standard (Financial Information eXchange). More detailed information about the standard can be found in reference document 1 (see 1.5) or on the website www.fixprotocol.org.

The interface follows the FIX 4.4 specifications, as far as possible. In the majority of cases the structure and semantics of the messages are identical to the standard.

In some cases, the protocol has been extended to cover functions not considered by the standard. These extensions are clearly detailed in the document.

In other cases, the standard is ambiguous or indicates that the details should be mutually defined by the parties. In these cases the manual provides a detailed description to avoid any possible ambiguity.

All annotations and adaptations of the standard have been done in accordance with the recommendations in the standard.

To avoid possible duplication in the sources of information, this document does not include explanations of those matters that comply exactly with the standard. Therefore, the standard documentation should be considered as the main source of information for any matter that is not explicitly covered in this manual.

This is a reference document for those Members and ISVs that wish to develop software that can communicate with the CCP using the MEFFGate server FIX interface.

1.2 Public and private information

The functions covered by MEFFGate are grouped into public and private information.

The following table displays the public functions and their related messages.

Public function	Related messages	Msg Type
Obtain information on contracts	Security List Request	x
	Security List	y
	Market Data Request	V
	Market Data Request Reject	Y
	Market Data – Snapshot / Full Refresh	W

The following table displays the private functions and their related messages.

Private function	Related messages	Msg Type
Monitoring and management position	Request For Positions	AN
	Request For Positions Ack	AO
	Position Report	AP
	Trade Capture Report	AE
	Position Maintenance Request	AL
	Position Maintenance Report	AM
Request for trades	Trade Capture Report Request	AD
	Trade Capture Report Request Ack	AQ
	Trade Capture Report	AE
Trades management	Allocation Instruction	J
	Allocation Instruction Ack	P
	Confirmation	AK
	Confirmation Ack	AU
	Trade Capture Report	AE
Exercise instruction	Position Maintenance Request	AL
	Position Maintenance Report	AM
	Request For Positions	AN
	Request For Positions Ack	AO
	Position Report	AP
	Trade Capture Report	AE
Send messages to the CCP supervisor and Reception of administrator's messages	News	B
Give-up references and filters management	Registration Instructions	o
	Registration Instructions Response	p

1.3 Structure of manual

The manual is divided into two parts. The first part, containing the first four chapters, gives a description of generic features of this interface.

This first chapter describes the scope of the document, its structure and introduces the related documents.

Chapter 2 "Implementation decisions" presents those annotations or restrictions arising from the implementation of the protocol defined in this manual.

Chapter 3 "FIX Session" describes those aspects related to the session level, including the detailed description of the corresponding messages.

Chapter 4 "General conventions in application messages" describes in detail specific aspects that affect the majority of the messages described in this manual.

Given the generic nature of the content, which affects all the messages, it is recommended to read chapters 2, 3 and 4 before considering other chapters.

The second part of the manual, containing the remainder of the chapters, describes the different functions supported by MEFFGate. Each of these chapters deals with a specific function, describing specific matters of interest.

Each of these chapters contains the following sections:

- **Introduction**. A brief description of the function covered in the chapter

- **List of messages**. List of the different messages implemented by the function

- **Message flow**. Description of the different scenarios for message exchange that may arise, with the corresponding message flow diagrams

- **Annotations and adaptations of FIX 4.4**. Details the annotations and adaptations that has made to the standard protocol to meet its needs

- **Definition of messages**. Contains a table for each message in the chapter, describing the message fields in detail

Finally, various tables providing information referred to throughout the document are included as appendices.

1.4   Format of the message definition tables

As explained in the previous section, a table for each message is included in those chapters where it is necessary, describing the component fields in detail.

These tables contain one field per row and have the following columns:

Column	Meaning
Tag	Field number. The fields added to the message in this implementation have an asterisk ("*") after the number
Name	Name of field according to the FIX standard
Req	"Y" indicates that the field is required; "N" means that the field is optional. "Y*" means that the field is required in this implementation, but it is optional in the FIX 4.4 standard
Valid values	Accepted values for the field in the context of the message. It may be a list of values, or a range of numeric values, e.g. ">=3, <= 10". The default value for the field is also indicated in this column. To avoid confusions with the terms, the original FIX value description has been respected in the values associated with codes.
Format	Type of data in the field. It is one of the types defined by FIX, or one of these types with some additional restriction. String(n) is a String type with a maximum of n characters, or in some cases with exactly n characters. For more information on the String type, See 2.4
Description	Description of the field in the context of the message

1.5   Related documents

#	Title	Author
1	Financial Information Exchange Protocol (FIX) 4.4 with errata 20030618	FIX Committee
2	Financial Information Exchange Protocol (FIX) 5.0 Service Pack 2 (9 December 2013) EP98-222 enhancing FIX 5.0 SP2	FIX Committee

2. Implementation decisions

2.1 Description

This chapter presents the implementation decisions made by MEFF. Those aspects that the standard leaves open and have been defined in its implementation are detailed here.

2.2 Fields ignored

In some cases, the content of certain fields of the entering messages may be ignored by MEFFGate. When this is the case, it is clearly stated in the field description.

2.3 Unsupported fields

The unsupported fields of a message are not included in its description.

Messages sent to MEFFGate should not contain unsupported fields. Messages sent by MEFFGate never contain unsupported fields.

No required fields have been declared unsupported.

2.4 Length of String type

The FIX standard does not place any restriction on the maximum length of the String type. In this implementation the maximum length is 255 characters.

In some fields, a shorter maximum length has been established. In these cases, the type is presented as String(n), where "n" is the maximum number of characters of the field. In certain cases "n" indicates the exact length of the field, in which case it will be explicitly stated in the valid values column.

2.5 Maximum length of message

The maximum length of the messages sent or received by MEFFGate is 4096 bytes.

2.6 Encryption

MEFFGate does not use the encryption defined in the FIX standard (using the SecureData and SecureDataLen fields in the message header). The encryption is implemented through the use of SSL (*Secure Socket Layer*).

2.7 Identification of the MEFFGate FIX protocol

MEFFGate implements an additional function that allows both parties to agree on the MEFFGate FIX version that they are going to use.

It is important to distinguish between the version of the FIX protocol (in this case "4.4") and the version of the MEFFGate FIX protocol ("C1.8" in this edition).

The MEFFGate FIX protocol is the annotation and adaptation of the standard by MEFF to meet the specific needs of its business. All the annotations and adaptations made by MEFF have been done in accordance with the recommendations specified in the standard.

More than one version of the MEFFGate FIX protocol may exist for the same version of FIX. Similarly, one version of the MEFFGate FIX protocol may work with more than one version of the FIX protocol.

The MEFFGate FIX protocol version used as default by MEFFGate is determined in the MEFFGate user configuration. The client program may also indicates which MEFFGate FIX protocol version it is going to use in the ProprietaryFixProtocolVersion of the Logon message. This field has been added by MEFF and has been implemented as an optional field.

If the version requested by the client program is not available in the MEFFGate server in use, it will return a Logout Message with the corresponding explanatory message.

3. FIX Session

3.1 Introduction

The level of the FIX session guarantees the complete delivery of messages between both parties, without errors. MEFFGate implements the majority of the functions of the session level defined in the FIX 4.4 standard.

3.2 FIX session and communication session

As explained in the standard, there are two types of session:

- **Communication session**. This begins when a request to start a session (Logon message) is accepted. It ends when the communication is completed, preferably with the exchange of Logout messages

- **FIX session**. This is a combination of two-way messages identified by a sequence of consecutive numbers. A FIX session begins when the sequence numbers of both parties are restarted with the value 1. There is no explicit way of ending a FIX session; a session ends when a new one begins. A FIX session can encompass more than one communication session

In addition to the two mentioned types of sessions, the Clearing Contract Group session should also be considered. A Clearing Contract Group session begins each day when the MEFFGate server loads the Clearing Contract Group data again and accepts connections for that session, following the procedure defined by MEFF.

The client program must begin a new FIX session in the first connection of the Clearing Contract Group session.

Given that MEFFGate does not provide 24-hour support for the service, the ResetSeqNumFlag field is not required in the Logon message.

3.3 Identification of the FIX session

Once a communication session has been established, MEFFGate identifies the associated FIX session using four fields in the Logon message sent by the initiator:

- SenderCompID

- SenderSubID

- TargetCompID

- TargetSubID

SenderCompID identifies the member and SenderSubID identifies the user. TargetCompID together with TargetSubID identify the Clearing Contract Group.

No more than one FIX session can exist at a time with the same values for these four fields.

The SenderCompID, SenderSubID, TargetCompID and TargetSubID fields are present in all the FIX messages. All the messages belonging to the same FIX session must have the same values in these fields. If a message is received with values that do not correspond with those of the session, it will be rejected with a Reject message.

It should be noted that the values of these fields are inverted when the message is sent by MEFFGate, with respect to those sent by the client. Suppose that user "001" of member "A001" has a session established with Financial Contract Group at MEFF. The messages will be those shown below:

Client message to MEFFGate:

- o SenderCompID = "A001"

- o SenderSubID = "001"

- o TargetCompID = "XMEF"

- o TargetSubID = "C2" [*]

MEFFGate message to client:

- o SenderCompID = "XMEF"

- o SenderSubID = "C2"

- o TargetCompID = "A001"

- o TargetSubID = "001"

If the same client wants to operate with two Clearing Contract Groups simultaneously, or wishes to make two different connections with the same Clearing Contract Group, he will have to establish various connections as explained in 3.4.

3.4   Client software and FIX sessions

A MEFFGate client is a software development that connects to MEFF through a MEFFGate server.

As noted in 3.3, a FIX session is limited to one user and one Clearing Contract Group. A client will be able to establish various FIX sessions simultaneously to access more than one Clearing Contract Group or trade in one Clearing Contract Group with various user codes.

A MEFFGate server can provide service to various sessions simultaneously, be they of the same client or various clients.

When a FIX client tries to connect with a Clearing Contract Group that is not available, his Logon message is answered with a Logout message with the appropriate explanation.

3.5   Synchronisation of the FIX session

On initiating a communication session (sending Logon message), the client can opt to initiate a new FIX session or continue with a previous session from the same Clearing Contract Group session. The procedure to follow in each case is described below.

- **Start a new FIX session**. To start a new FIX session the value to be used in the MsgSeqNum field of the Logon message must be 1.

- **Continuation with previous FIX session**. To continue a previous FIX session the value to be used in the MsgSeqNum field is the consecutive number to the one used in the last message of the session to be continued.

Note that on starting a new FIX session, it will replace the previous session and therefore it cannot be continued later.

A client application can only continue a FIX session if it connects to the same MEFFGate server with which the FIX session had been initiated. In the case that it has to connect to a different MEFFGate server a new FIX session will have to be started.

---

[*] See table 18 in document "Codification Tables" for a list of available Clearing Contract Groups

When a Logon message has a MsgSeqNum other than 1, the number sent might not coincide with the number that MEFFGate was expecting for this session (MEFFGate expects the consecutive number after the number of the last message received for that session). The different situations that can arise with respect to the sequence number of the Logon message and the message expected by MEFFGate are shown below.

- **MsgSeqNum is the same as the number expected by MEFFGate**. There is no discrepancy and MEFFGate accepts the connection.

- **MsgSeqNum lower than the number expected by MEFFGate**. MEFFGate rejects the connection by sending a Logout message with the sequence number 1. In this case the connection cannot be accepted as the sequence number used in the Logon message was previously used by another message in the same session.

- **MsgSeqNum higher than the number expected by MEFFGate**. MEFFGate accepts the connection, and requests that the missing messages between the sequence number expected and the number used by the client be sent. The request to send these messages will be made using one of the following mechanisms:

    o **Logon message with NextExpectedMsgSeqNum**. This method will be used whenever the client has used the NextExpectedMsgSeqNum field in his Logon message

    o **Resend Request message**. In the event that the client has not used the NextExpectedMsgSeqNum field (Logon message), the server will not use it. In this case the server, after the Logon message reply, will notify the client of the sequence error using the Resend Request message

When continuing with a FIX session it is advisable to use the NextExpectedMsgSeqNum field (Logon message), indicating the message number expected to facilitate the synchronisation between both parties. In this way, the server can detect if the client did not receive some of the messages, and proceed to resend them. When this value is not specified, the server continues sending messages from the last number in the sequence sent.

MEFFGate uses the NextExpectedMsgSeqNum field in the reply to the Logon message if the client used it in his start message.

MEFFGate only uses the Resend Request message as described above, that is, immediately after a Logon request. Any other time when there is a discrepancy between the expected sequence number and the number received, MEFFGate assumes that there is a serious problem and closes the connection with a Logout message.

When MEFFGate requests a lower sequence number from the client than the one that the client proposes, the client can decide whether to repeat the same messages that he sent with the sequence numbers that are missing, send one or more GapFills, or simply give them a new use. In any case the client must take into account that the messages will be processed by MEFFGate in the moment that it is received like any other message. It is recommended to set up controls to prevent orders being sent that could have become obsolete whilst the disconnection lasted.

Like MEFFGate, the client can only use the Resend Request message in the case of detecting that the Logon message sent by the server is out of the sequence and does not contain the NextExpectedMsgSeqNum field. If MEFFGate receives a Resend Request message at any other moment it will be considered as an error and the connection will be terminated.

Independent of whether it was requested through the Resend Request message or using the NextExpectedMsgSeqNum field of the Logon message, the repetition of messages by the server is limited to messages belonging to the Clearing Contract Group Session underway. The server repeats all the messages requested, except those providing public information. Given the volume of public information that may exist, it cannot be repeated and will be replaced by a "Gapfill" messages. The client must use query functionalities available to get up to date on this information. See section 3.6 for more information on this matter.

A) The FIX session continues without request for repetition of messages
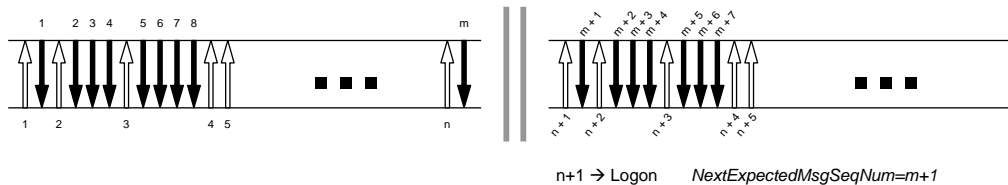
B) The FIX session continues with messages request

n+1 → Logon   *NextExpectedMsgSeqNum=m+1*

n+1 → Logon   *NextExpectedMsgSeqNum=**m-50***

Message of server to client

Message of client to server

Menssage SequenceReset-GapFill sent from server to the client to skip the public messages (unrepeatable)
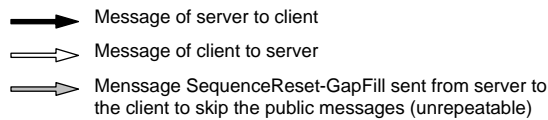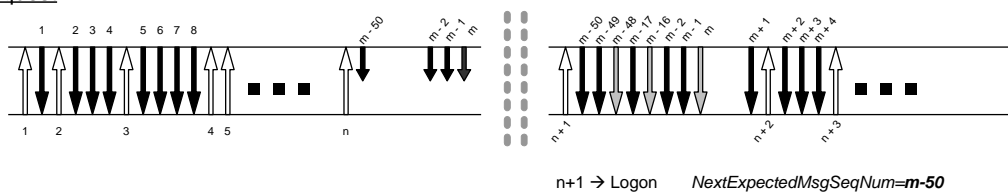
- Scenarios for start or continuation of a FIX session -

When the start of a communications' session is rejected, the server answers with a Logout message. The client application should take into account that this Logout message may come with a sequence number 1 or with the sequence number corresponding to its late communication. The sequence number used by the client in its Logon message is not considered by MEFFGate and therefore does not alter the expected sequence number in subsequent communications.

3.6 Synchronisation at application level

When a client starts a communication session (Logon message accepted), it receives a series of information related with the current Clearing Contract Group session. The information to be received depends on whether it is the start of a new session or the reconnection to an existing FIX session:

- **Start of a FIX session**. The client receives all the private messages, not associated to subscriptions, corresponding to the entire Clearing Contract Group session. All or part of these messages may have been previously received in a previous FIX session.

- **Reconnection to an existing FIX session**. The client receives all the private messages, not associated to subscriptions, corresponding to the current Clearing Contract Group session, not received previously. In this case MEFFGate ensures that there is no duplication of messages (except when it is explicitly requested).

The messages that come from an explicit request for repetition (requested using a Resend Request message or a Logon message with NextExpectedMsgSeqNum lower than the last message received) will contain the value "Y" in the PossDupFlag field indicating this situation. This does not apply when it is a new FIX session (start of the session with sequence number 1).

In both cases, both the private information associated with subscriptions and the public information can be requested using the messages that implement the corresponding functionalities.

It should be taken into account that any subscription to information is cancelled when the communication session ends. If this service is required when reconnecting to a FIX session, it must be requested again.

The private messages not associated to subscriptions referred to in this section correspond to these messages:

- News

- Confirmation messages with the SecondaryConfirmStatus [5683] value different that "P" (Pending)

- Position Maintenance Report with PosTransType = 4 (Delivery)

3.7 High availability

To improve the availability of access to MEFF there will be various instances of the MEFFGate server executing in different computers.

All the instances of MEFFGate will be connected with the central systems of MEFF. Therefore, they will have all the necessary information.

When a MEFFGate server fails, the client can continue working with another MEFFGate. In this case, the client must start a new FIX session (beginning with the sequence number 1), as the status of the sessions is not replicated between different MEFFGate servers. The client must carry out the necessary processes to synchronise at the application level (see 3.5).

When a client application that has established a FIX session fails, the client application can restart in another computer that continues with the same session (using the same MEFFGate server). In this case, the client is responsible for restoring the status of the failed application. If it is not possible to restore this status, it is recommended that a new FIX session be started, although it is possible to maintain the current session and consult all the necessary data at the application level to restore the status.

3.8 PossResend field

The implementation of FIX in MEFF does not allow the use of the PossResend field (see 3.4 for more information on unsupported fields).

3.9 Administrative messages that the FIX client must manage

The FIX client must be able to manage all the administrative messages as described in this chapter, including the Resend Request message. The client can decide whether to respond to this by resending messages or simply sending a GapFill (Sequence Reset message).

3.10 List of messages

The functionality at the session level is implemented in FIX 4.4 using seven administrative messages. All these are fully supported by the MEFFGate FIX protocol.

Message	Description
Logon (Msg Type = A)	Request or confirmation of the start of a communication session
Logout (Msg Type = 5)	Request or confirmation of the end of a communication session

Heartbeat (Msg Type = 0)	Periodic notification that the connection continues to be live
Test Request (Msg Type = 1)	Request to send a Heartbeat message to confirm that the connection is alive
Resend Request (Msg Type = 2)	Request to resend messages that have not been received
Sequence Reset (Msg Type = 4)	Fill a gap in messages re-establishing a higher sequence number
Reject (Msg Type = 3)	Reject a message at session level

3.11 Message flow

Start of communication session and start of FIX session

A request to start a communication session (Logon message) that is accepted is replied to by the receiver with another Logon message. The initiator must not send another message until it has received this confirmation of acceptance.

MEFFGate Client   MEFFGate Server

Logon ("A")

MsgSeqNum [34] = 1

Logon ("A")

MsgSeqNum [34] = 1
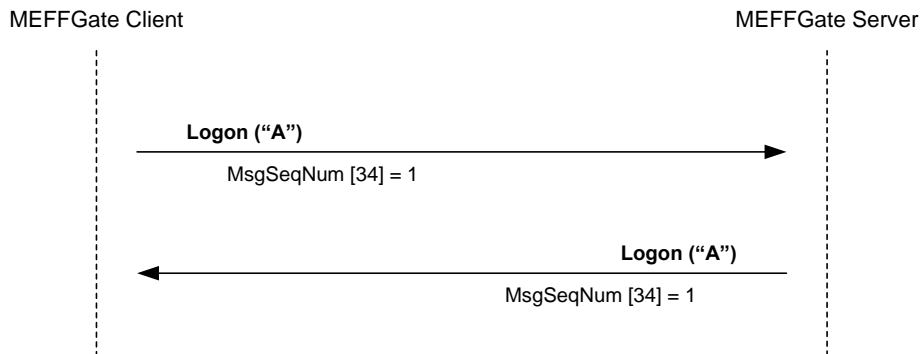
Start of communication session and continuation of FIX session

A request to start a communication session (Logon message) that is accepted is replied to by the receiver with another Logon message. The initiator must not send another message until it has received this confirmation of acceptance.

MEFFGate Client   MEFFGate Server

Logon ("A")

MsgSeqNum [34] = m
NextExpectedMsgSeqNum [789] = n

Logon ("A")

MsgSeqNum [34] = n
NextExpectedMsgSeqNum [789] = m+1
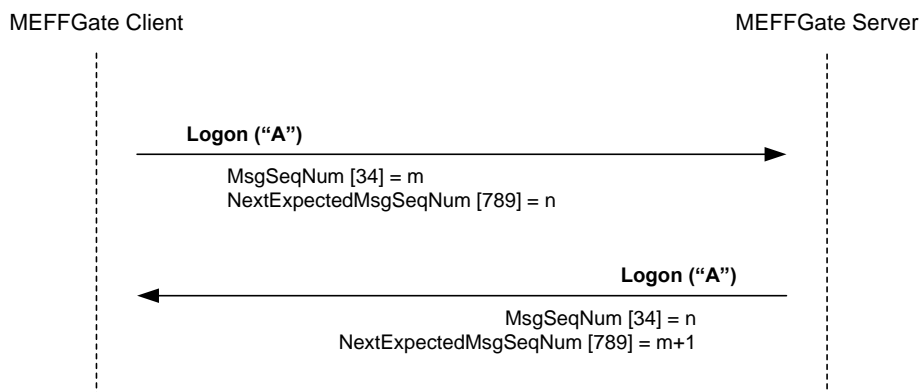
Start of communication session rejected

When the start of a communication session (Logon message) is not accepted, MEFFGate will reply with a Logout message.

For more details on the behaviour of sequence numbers of both parties see section 3.5.

MEFFGate Client   MEFFGate Server

Logon ("A")
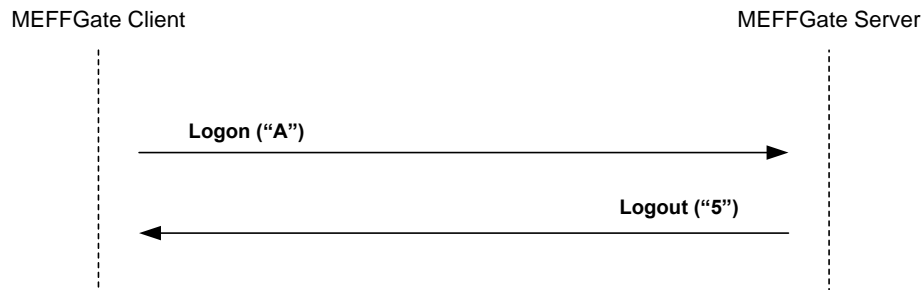
Logout ("5")

End of a communication session started by the sender

The client can end the communication session by sending a Logout message at any time.

MEFFGate Client   MEFFGate Server
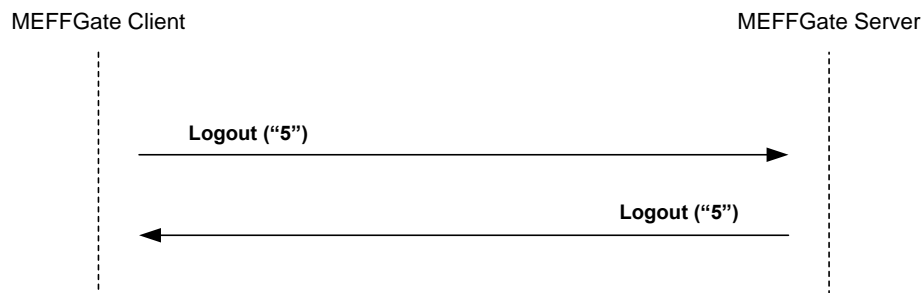
Logout ("5")

Logout ("5")

End of a communication session started by the receiver

In exceptional circumstances, the server can end the communication session with a Logout message.

MEFFGate Client   MEFFGate Server

Logout ("5")

Logout ("5")
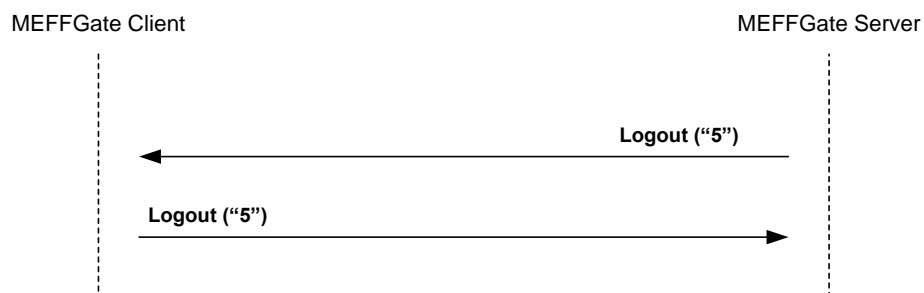
Sending messages with identification fields of session (SenderCompID, SenderSubID, TargetCompID and TargetSubID) with different values from those associated to the current FIX session

All the messages associated to a FIX session must include the same identifying values of the session (SenderCompID, SenderSubID, TargetCompID and TargetSubID). If a message differs from the values indicated in the Logon of the session, it is rejected with a Reject message.

MEFFGate Client   MEFFGate Server

Any message with wrong ID

Reject ("3")

SessionRejectReason [373] = 9 (CompID problem)
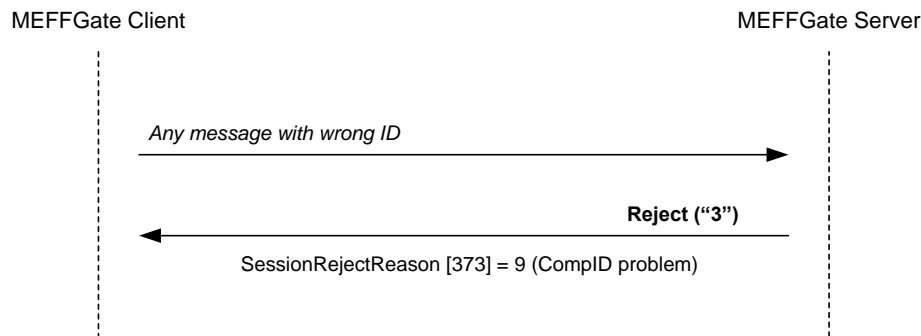
3.12 Annotations and adaptations of FIX 4.4

- The optional field ProprietaryFixProtocolVersion has been added to the Logon message to identify the version of the MEFF protocol

- The optional field FixEngineName has been added to the Logon message. It contains a descriptive text of the software application

- When a request to start a session (Logon message) is rejected, the receiver (MEFF) will always send a Logout message in reply

- The SenderSubID and TargetSubID fields in the header of messages (Standard Message Header) are now required

- The PossResend field is not supported

- The FIX method of encryption is not supported

- MEFFGate only accepts the Resend Request message after the Logon message

- The valid values of the ResetSeqNumFlag field in the Logon message are limited to the value "N"

3.13 Definition of messages

3.13.1 Standard Message Header

Tag	Name	Req	Valid values	Format	Description
8	BeginString	Y	FIX.4.4	String	Indicates the start of a new message. Contains the version of the FIX protocol. It is always the first field of the message
9	BodyLength	Y		Int	Length of message in bytes, from the end of this field up to and including the delimiter before the Checksum field. It is always the second field of the message
35	MsgType	Y	All message types supported by MEFF	String	Identifies the type of message. It is always the third field of the message
49	SenderCompID	Y		String	Identifier of the entity that sends the message. Contains "XMEF" when the message is sent by MEFFGate. It must contain the member code in the messages sent by the client application
56	TargetCompID	Y		String	Identifier of the entity that the message is sent to. Must contain "XMEF" when the message is sent to MEFFGate. Contains the member code in the messages sent by MEFFGate
34	MsgSeqNum	Y		Int	Sequence number of the message within the current FIX session
50	SenderSubID	Y*	See table 18 in document "Codification Tables" for details of the Market codes	String	The messages sent from MEFFGate to the client contain the code assigned to the Contract Group with which the connection was established. Messages sent to MEFFGate must contain the trader code with which the FIX session was started
57	TargetSubID	Y*	See table 18 in document "Codification Tables" for details of the Contract Group codes	String	The messages sent from MEFFGate contain the code of the trader which it is to be sent to. Messages sent to MEFFGate must contain the code of the Contract Group with which the connection was established
43	PossDupFlag	N	N = Original message sent (default value) Y = Duplicate possible	Boolean	Indicates if it is the first time within a FIX session that a message is sent ("N") or if the same message is being sent again ("Y"), either due to an explicit request from the other party or due to uncertainty about the reception of the original message
52	SendingTime	Y		UTC Timestamp	Time message sent
122	OrigSendingTime	N		UTC Timestamp	Time when original message was sent. Required in a resend. A message is considered to be a resend if the field PossDupFlag = "Y" and the field MsgType is not "4" (Sequence Reset)

3.13.2 Standard Message Trailer

Present in all FIX messages.

Tag	Name	Req	Valid values	Format	Description
10	CheckSum	Y		String(3)	Checksum of the message, calculated in accordance with the standard. It is always the last field of the message and its length is exactly 3 bytes

3.13.3 Logon (Msg Type = A)

The Logon message is used to start a session by the client application and to accept it by the MEFFGate.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = A		
98	EncryptMethod	Y	0 = None	Int	Ignored by MEFFGate
108	HeartBtInt	Y	>= 5	Int	Interval at which messages are sent to verify the connection (Heartbeat message) expressed in seconds.
141	ResetSeqNumFlag	N	N	Boolean	Only allows the value "N", as it is not required in the implementation of the protocol
789	NextExpectedMsgSeqNum	N		Int	Indicates the next sequence number (MsgSeqNum) that it expects to receive
464	TestMessageIndicator	N	Y = Test N = Production	Boolean	Indicates whether it is a test or production session. The client can use it optionally to indicate if it wants to connect to the production or test environment. The start of a session is accepted only if this environment is valid for the MEFFGate If the client does not indicate anything, this parameter is not taken into account. In any event MEFFGate always informs this field
553	Username	N		String	User password. Required when the message is sent by the client application. It is currently comprised of the combination of the member code and the user code assigned by MEFF
554	Password	N		String	Password assigned by MEFF. Required when the message is sent by the client application
5680*	ProprietaryFixProtocolVersion	N	C1.8	String	Exact identification of the version of the protocol used and expected by the client application. MEFFGate always compliments this field, independently if the client application informed it or not.
5679*	FixEngineName	N		String	Optional field, where the client can include a descriptive string of the software name used by the FIX connection. Only used for informative purposes. MEFFGate never sends this field
	Standard Trailer	Y			

3.13.4 Logout (Msg Type = 5)

The Logout message is used by both parties to request the end of a communication session and to accept said request.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = 5		
58	Text	N		String	Explanatory text
	Standard Trailer	Y			

3.13.5 Heartbeat (Msg Type = 0)

The Heartbeat message is used by both parties to indicate that the connection is active.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = 0		
112	TestReqID	N		String	If the message is the reply to a Test Request message, it must contain the same value as the original TestReqID field. Otherwise, this field should be omitted.
	Standard Trailer	Y			

3.13.6 Test Request (Msg Type = 1)

The Test Request message is used by both parties to request that a Heartbeat message be sent.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = 1		
112	TestReqID	Y		String	Identifier of the request. It must be included in the Heartbeat message reply
	Standard Trailer	Y			

3.13.7 Resend Request (Msg Type = 2)

The Resend Request message can be used by both parties to request the resending of messages that have not been received.

MEFFGate only makes use of this functionality after a Logon message. In any other case the reception of a message out of sequence is considered a serious error and the connection is terminated.

MEFFGate will only accept this type of message immediately after receiving a Logon message in which the NextExpectedMsgSeqNum field has not been specified.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = 2		
7	BeginSeqNo	Y	Valid sequence number	Int	Sequence number of the first message in the range of messages requested to be resent. Its value must be less than the last sequence number received
16	EndSeqNo	Y	0 = Infinite Valid sequence Number	Int	Sequence number of the last message of the range of messages requested to be resent. Its value must be lower than the last sequence number received. If the request is only for one message, then EndSeqNo = BeginSeqNo If the request is for all the messages starting from a given one, then EndSeqNo = 0
	Standard Trailer	Y			

3.13.8 Sequence Reset (Msg Type = 4)

The Sequence Reset message is used by both parties to fill a gap in the messages that are being sent, by reassigning the sequence number (see 3.5). The FIX standard allows other uses of this message that are not supported by MEFF (note that the GapFillFlag field is required and must always contain the value "Y").

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = 4		Note that the PossDupFlag must contain the value "Y"
123	GapFillFlag	Y*	Y = Indicates that the message is to fill a gap	Boolean	More information is available in the FIX 4.4 specifications documentation
36	NewSeqNo	Y		Int	Sequence number of the message that will now be sent
	Standard Trailer	Y			

3.13.9 Reject (Msg Type = 3)

The Reject message is used by MEFFGate to reject a message that does not comply with the FIX protocol specified by MEFF.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = 3		
45	RefSeqNum	Y		Int	Sequence number of the rejected message
373	SessionRejectReason	N	0 Invalid tag number 1 Required tag missing 2 Tag not defined for this message type 3 Undefined Tag 4 Tag specified without a value 5 Value is incorrect (out of range) for this tag 6 Incorrect data format for value 9 CompID problem 11 Invalid MsgType 13 Tag appears more than once 14 Tag specified out of required order 15 Repeating group fields out of order 16 Incorrect NumInGroup count for repeating group 17 Non "data" value includes field delimiter (SOH character) 99 Other	Int	Code indicating the rejection motive
58	Text	N		String	Contains a more detailed explanation of the reason for the rejection
	Standard Trailer	Y			

4. General conventions in application messages

4.1 Trade identification

4.1.1 SecondaryExecID

The SecondaryExecID field has the trade register number. This is the code assigned by the central trading system to the trade or cross trade referred to in the message. The period in which the uniqueness of this field is guaranteed is determined by each central trading host.

4.1.2 SecondaryTradeReportID

The SecondaryTradeReportID field has the CCP register number. This is the code assigned by the MEFF central clearing system to the trade referred to in the message. Both sides to the trade receive the same identifier.

Each trade in the trading system has a corresponding trade in the clearing system, but some trade types are specific to the clearing system, such as assignments, exercise trades, etc.

The SecondaryTradeReportRefID and TrdMatchID fields are also containing a CCP register number of the previous and initial trade respectively.

4.2 Parties block

The Parties block (or the NestedParties block) is used in many application messages to specify the parties involved in the transaction.

In the detailed definition of the messages that this block contains, the block is incorporated exactly as shown below. The list of possible values is restricted by the specific characteristics of the message.

Tag	Name	Req	Valid values	Format	Description
	Start <Parties>				
453	NoPartyIDs	N		NumInGroup	
→ 448	PartyID	N		String	Member code assigned by MEFF
→ 447	PartyIDSource	N	D = Proprietary/ Custom code	Char	Indicates the codification used in the PartyID field. MEFF's own codification is always used
→ 452	PartyRole	N		Int	Indicates the role taken by the party indicated in the PartyID field
	End <Parties>				

Various roles are used in the messages contained in this manual. The interpretation of the PartyID field depends on the value of the PartyRole, as explained below:

- **1 (Executing Firm)**

    - **Send.** The implemented protocol does not allow the use of this role in sending

    - **Receive.** When this value is specified, the PartyID field corresponds to the Executing Broker code of the Give-up treated

- **3 (Give-out Internal Reference)**

  - **Send**. When this value is specified, the PartyID field corresponds to the reference assigned by the Executing Broker for internal purposes

  - **Receive**. When this value is specified, the PartyID field corresponds to the Give-out internal reference

- **4 (Clearing Firm)**

  - **Send**. Used as a selection criteria. When this value is specified, it indicates that only information related to the accounts cleared by the member specified in the PartyID is requested

  - **Receive**. When this value is specified, the PartyID field corresponds to the member code which acts as a clearer of the account treated

- **12 (Executing Trader)**

  - **Send** The implemented protocol does not allow the use of this role in sending

  - **Receive.** When this value is specified, the PartyID field corresponds to the user that requested the transfer or Give-up

- **13 (Order Origination Firm)**.

  - **Send**. Used as a selection criteria. When this value is specified, the request to the member accounts regarding the PartyID field is being restricted

  - **Receive**. When this value is specified, the PartyID field corresponds with the member code of the account being handled.

- **14 (Give-up Clearing Firm)**

  - **Send**. When this value is specified, the PartyID field corresponds to the Clearing Broker code of the Give-up requested

  - **Receive**. When this value is specified, the PartyID field corresponds to the Clearing Broker code of the Give-up treated

- **24 (Give-up Reference)**

  - **Send**. When this value is specified, the PartyID field corresponds to the Give-up reference

  - **Receive**. When this value is specified, the PartyID field corresponds to the Give-up reference

- **33 (Give-up Mnemonic)**

  - **Send**. When this value is specified, the PartyID field corresponds to the Give-out mnemonic assigned by the Executing Broker, or the Give-in mnemonic assigned by the Clearing Broker

  - **Receive**. When this value is specified, the PartyID field corresponds to the Give-out mnemonic or Give-in Mnemonic

- **36 (Clearing Broker Trader)**

    - **Send** The implemented protocol does not allow the use of this role in sending

    - **Receive.** When this value is specified, the PartyID field corresponds to the user of the Clearing Broker that accepted or rejected the Give-up

4.3   Symbol and SecurityID

Symbol and SecurityID tags have different meanings depending on how the user associated to the FIX client has been configured.

These tags can be included in the Instrument, UnderlyingInstrument and InstrumentLeg blocks.

The two possible configurations are:

Type of Configuration	Meaning of Symbol	Meaning of SecurityID
By contract	MEFF Contract code	Underlying asset (see Table 21 in "Codification tables" document)
Por asset	Underlying asset (see Table 21 in "Codification tables" document)	MEFF Contract code

All sections and messages described in this manual refer to type of configuration "By contract".

4.4   Instrument block

In some requests, the FIX client may specify selection criteria for the contracts. In these cases, they will only receive information on the contracts that meet these criteria. The possible selection criteria correspond to the fields of the Instrument block. The table below indicates which fields are accepted by MEFF and the type of request that can be made.

Field	Meaning	Contract information request	Price information request
SecurityID	MEFF Underlying asset	X	X
Symbol	MEFF Contract code	-	X

The use of these fields is explained in detail in the following sub-sections.

4.4.1   Underlying asset (SecurityID field)

This code identifies the underlying asset of a contract (see table 21 in document "Codification Tables").

4.4.2   Contract code (Symbol field)

This is the most selective of the criteria, as it refers to a specific contract. It can only be used to request contract information. If you want to use the other selection criteria and do not want to specify a particular contract, complete this field with the value "[N/A]", as indicated in the FIX standard specifications. This field is obligatory in the Instrument block.

4.5 Error format (Text Field)

The Text field is used in various messages to include the description of an error. In these cases the format of the field is:

%MFsXXXXXX-Free text

Where **s** indicates the importance of the error (I: information, W: warning, E: Error), **XXXXXX** is the error code, which is followed by an explanation. "%MF" text is always present.

5. Common Application Messages

5.1 Introduction

This chapter presents some common messages at the application level that cover two functions: the control of the communication status and the rejection of messages by MEFFGate.

5.2 Network communication status

MEFFGate includes a mechanism to inform the client application of the status of communication between MEFFGate itself and the central system. This functionality is achieved using the FIX Network Status messages.

The request can be made on demand or by subscription.

5.3 Password change

This functionality allows to change the individual user password used in the connection between the client application and MEFFGate.

The new password is valid for all the next future communication sessions between the client application and MEFFGate.

5.4 Rejection of application messages

When MEFFGate receives a supported message with correct syntax in an unsupported situation, but there is no specific rejection message, the Business Message Reject is used. In particular, this is used to reject the Confirmation Ack and Network Counterparty System Status Request messages.

5.5 List of messages

Message	Description
Network Counterparty System Status Request (Msg Type = BC)	Request of connection status between MEFFGate and the central systems
Network Counterparty System Status Response (Msg Type = BD)	Report on status of connection between MEFFGate and the central systems
User Request (Msg Type = BE)	Individual user password change request
User Response (Msg Type = BF)	Reply to a User Request message
Business Message Reject (MsgType = j)	Rejection of message at application level (used when there is no specific message)

5.6   Message flow

Check connection status on demand

MEFFGate Client   MEFFGate Server

Network CounterParty System Status Request ("BC")

NetworkRequestType [935] = 1 (Snapshot)

Network CounterParty System Status Response ("BD")
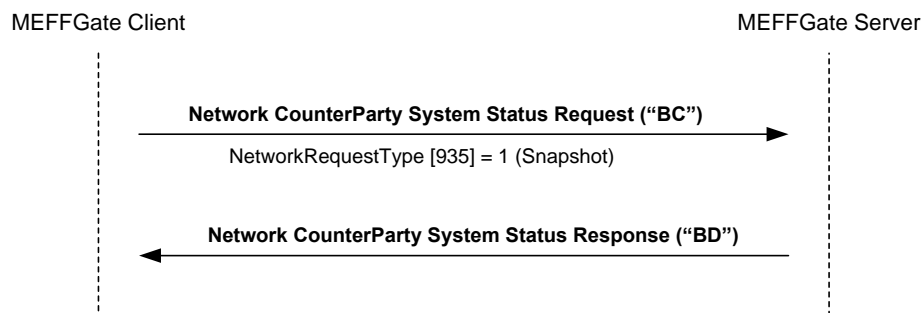
Subscription to connection status

MEFFGate Client   MEFFGate Server

Network CounterParty System Status Request ("BC")

NetworkRequestType [935] = 2 (Subscribe)

Network CounterParty System Status Response ("BD")

…
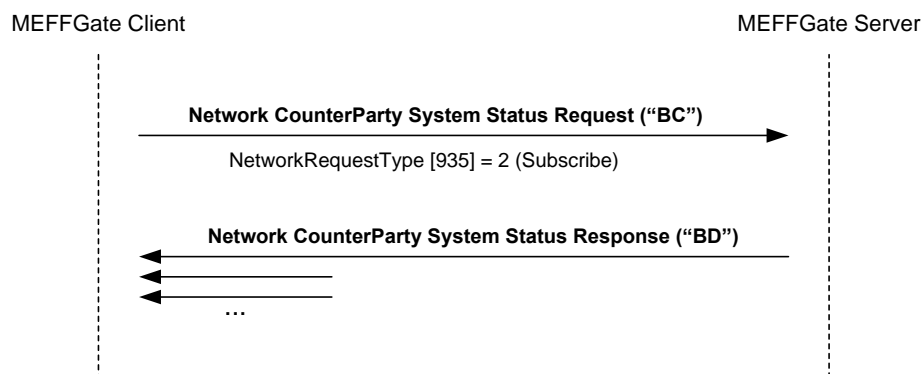
Individual password change

MEFFGate Client   MEFFGate Server

User Request ("BE")

User Response ("BF")
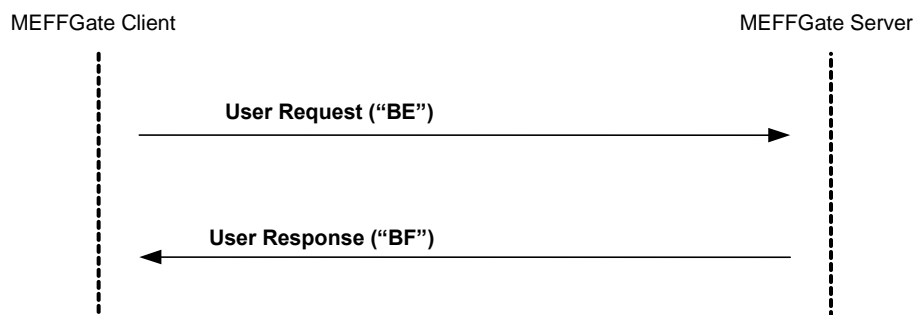
5.7   Annotations and adaptations of FIX 4.4

In the User Request message, the Password [554] and NewPassword [925] fields are now required.

5.8   Definition of messages

5.8.1   Network Counterparty System Status Request (Msg Type = BC)

Message sent by the client application to request information on the status of the connection between MEFFGate and the MEFF central systems.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = BC		
935	NetworkRequestType	Y	1 = Snapshot 2 = Subscribe 4 = Stop subscribing	Int	
933	NetworkRequestID	Y		String(10)	Message identifier
	Standard Trailer	Y			

5.8.2 Network Counterparty System Status Response (Msg Type = BD)

Message sent by MEFFGate as reply to a Network Counterparty System Status Request Message.

It has information about the connectivity between MEFFGate and the MEFF central systems.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = BD		
937	NetworkStatusResponseType	Y	1 = Full	Int	
933	NetworkRequestID	Y		String	Message identifier Network Counterparty System Status Request
932	NetworkResponseID	Y		String	Unique message identifier
936	NoCompIDs	Y	1	NumInGroup	
→930	RefCompID	N	MEFF	String	Contains the same value as the SenderCompID field in the header (see 3.3) This field is always included in the message
→931	RefSubID	N	Clearing Contract Group Code. See table 18 in document "Codification Tables" for details of the Clearing Contract Group codes	String	Contains the same value as the SenderSubID field in the header (see 3.3) This field is always included in the message
→928	StatusValue	N	1 = Connected 2 = Not connected – down expected up 3 = Not connected – down expected down	Int	Connection status This field is always included in the message. The value "3" indicates the end of the clearing session. After this moment MEFFGate can keep accepting connections and information requests. In this situation the StatusText field contains "%MFI – End of session"
→929	StatusText	N		String	Additional information
	Standard Trailer	Y			

5.8.3 User Request (Msg Type = BE)

Message sent by the client to modify the password used in their connection to the MEFFGate

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = BE		
923	UserRequestID	Y		String (10)	Unique identifier for each User Request message
924	UserRequestType	Y	3 = Change Password For User	Int	
553	Username	Y		String	Identifier of the user assigned by MEFF. It is currently comprised of the combination of the member code and the user code assigned by MEFF
554	Password	Y*		String (10)	Old Password
925	NewPassword	Y*		String (10)	New Password
	Standard Trailer	Y			

5.8.4 User Response (Msg Type = BF)

Message sent by MEFFGate to notify the status of the request initiated with the User Request message.

This message is only sent to the user who made the request.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = BF		
923	UserRequestID	Y		String	Identifier assigned by the client in the User Request message
553	Username	Y		String	User identifier
926	UserStatus	N	5 = Password Changed 6 = Other	Int	Status of the User Request message If rejected (value 6) , there is an explanation in the UserStatusText field
927	UserStatusText	N		String	When UserStatus = 6 there is an explanation of the rejection
	Standard Trailer	Y			

5.8.5 Business Message Reject (MsgType = j)

Message sent by MEFFGate when it receives a supported message that is syntactically correct in an unsupported situation, and there is no specific rejection message. It is especially used to reject the Confirmation Ack and Network Counterparty System Status Request messages.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = j		
45	RefSeqNum	Y		Int	MsgSeqNum of the rejected message
372	RefMsgType	Y	.	String	MsgType of the rejected message
379	BusinessRejectRefID	N		String	Optional Identifier of the rejected message. For a Confirmation Ack message (MsgType=AU) rejected, this field contains the ConfirmID [664].
380	BusinessRejectReason	Y	0 = Other 3 = Unsupported Message Type	Int	Reason for rejection
58	Text	N		String	Explanation of rejection
	Standard Trailer	Y			

6. Contracts Information

6.1 Introduction

This chapter distinguishes between the static and the dynamic contract information

- **Static contract information**. Definition of contracts

- **Dynamic contract information**. Settlement price, traded volume and open interest

Each of these groups is covered in a separate section of this chapter. Section 6.4 provides details of the format of the corresponding messages.

6.2   Static contract information

6.2.1   Description

This functionality allows static contract information to be obtained defined in the CCP.

6.2.2   Request contract information

The request for the definition of contracts is made using the Security List Request message, as shown in the following table:

	Contracts Snapshot	Contracts Updates
Security List Request	X	
Security List Request NewSecuritySubscription = 1	X	X

The meaning of each column is explained below:

- **Contracts - Snapshot**. One or more Security List messages are obtained with the description of the contracts available that meet the selection criteria specified in the request

- **Contracts - Update**. One Security List message is obtained with information on a new contract when it is introduced into the system. Only information on new contracts that meet the selection criteria indicated in the request is received

Note that the NewSecuritySubscription field has been added to the Security List Request message by MEFF to enable the subscription to the definition of contracts created during the session, usually new strikes for options.

MEFF recommends the use of the Security List Request message with NewSecuritySubscription = 1 and SubscriptionRequestType = 1 so that the client application will have the maximum information. In this way information is obtained on all the changes that occur, including the entry of new contracts.

6.2.3   Reception of contract definitions

The information on the contract definitions is received in the Security List message. This message gives one contract at a time. The TotNoRelatedSym field gives the total number of contracts that meet the selection criteria and the NoRelatedSym field gives the number of contracts contained in that particular message.

When the current status plus updates is requested, first all the messages associated to the current status are sent and then the updates. This avoids confusion when following the list. It should be noted that the update messages only contain one contract per message.

6.2.4   Ending subscriptions

A Security List Request message is used to end a subscription to contract definitions, where the field NewSecuritySubscription = 2.

6.2.5 List of messages

Message	Description
Security List Request (Msg Type = x)	Sent by the client to request the definition of contracts
Security List (Msg Type = y)	Sent by the server to provide the contract definitions. It is also used to inform about the rejection of requests for this information

6.2.6 Flow of messages

Request contract definitions with update

After requesting contract definitions, one or more Security List messages will be received. Each of these messages indicates the total number of contracts that meet the selection criteria in the TotNoRelatedSym field and the number of contracts contained in the particular message in the NoRelatedSym field.

MEFFGate Client   MEFFGate Server

Security List Request ("x")

SubscriptionRequestType [263] = 1 (Snapshot + Update)

Security List ("y")

Unsolicited Indicator [325] = N
TotalNoRelatedSym [393] = 350
NoRelatedSym [146] = 1
…

Security List ("y")
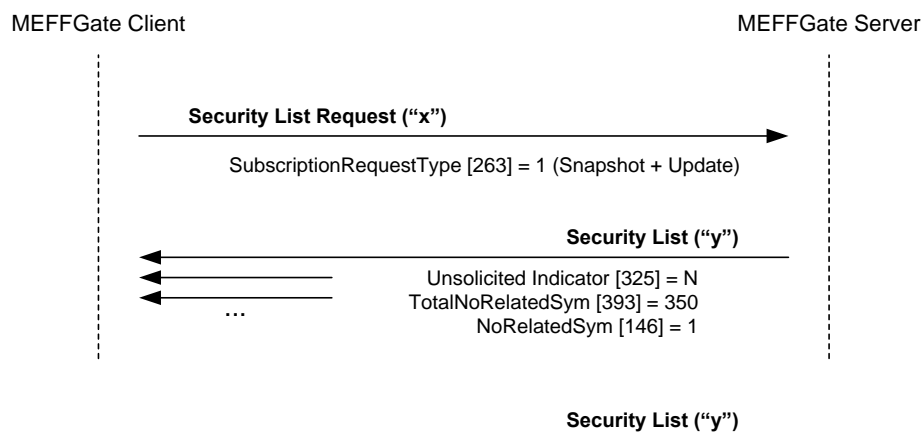
Request contract definitions with updates

If the request for contract definitions includes the updates (NewSecuritySubscription = 1) in addition to the messages detailed in the case above, when a new contract is entered in the system, a Security List message is received containing information on this contract.

MEFFGate Client   MEFFGate Server

Security List Request ("x")

SubscriptionRequestType [263] = 1 (Snapshot + Update)

Security List ("y")

Unsolicited Indicator [325] = N
TotalNoRelatedSym [393] = 350
NoRelatedSym [146] = 1
…

For every new contract

Security List ("y")

Unsolicited Indicator [325] = Y
TotalNoRelatedSym [393] = 1
NoRelatedSym [146] = 1
Symbol [55] = *New Symbol*
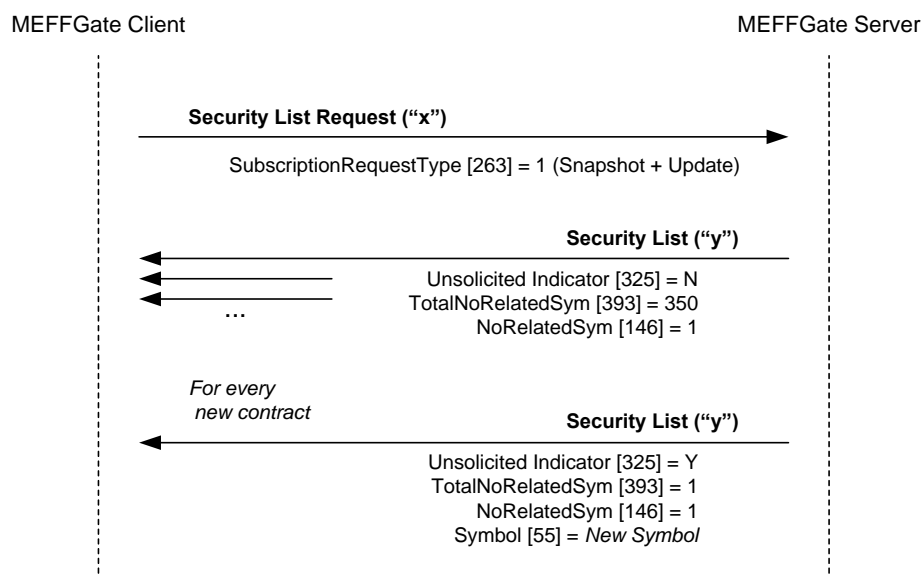
Request contract definitions, without contracts that meet the selection criteria

When there are no contracts that meet the selection criteria indicated in the contract definition request, MEFFGate will reply with a Security List message where the field SecurityRequestResult = 2. Note that in this case, if the request is made by subscription, that subscription will be active and therefore if new contracts are introduced that meet the selection criteria the corresponding messages will be received.

FIX client   FIX server

SecurityListRequest ("x")

SecurityList ("y"); SecurityRequestResult = "2"
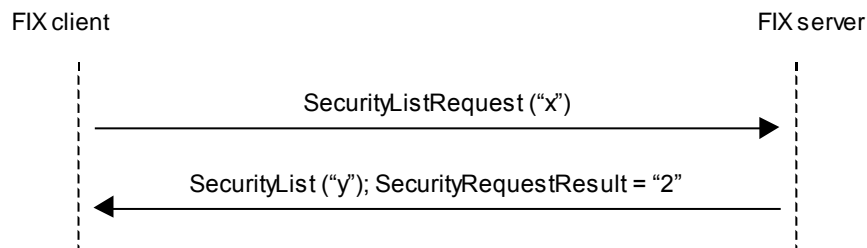
Failed contract definition request

When a contract definition request is erroneous, it is answered with a Security List message where the field SecurityRequestResult = 1.

MEFFGate Client   MEFFGate Server

Security List Request ("x")

Security List ("y")

SecurityRequestResult [560] =
2 (No instruments found that match selection criteria)
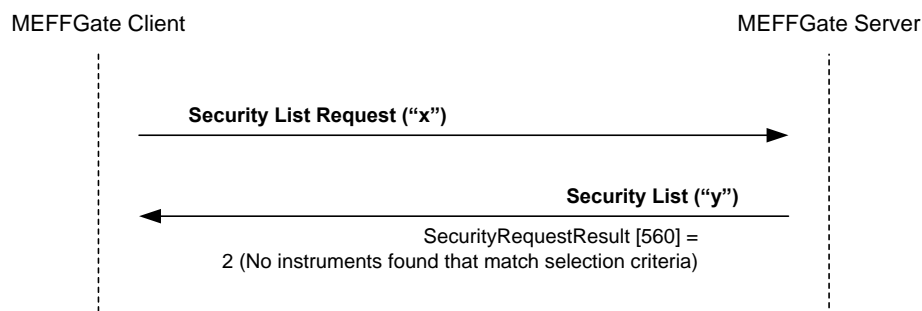
6.2.7 Annotations and adaptations of FIX 4.4

- The NewSecuritySubscription (5682) user field has been added to the Security List Request message to support the functionality of receiving the contract definition when a new contract is entered in the system

- The AccruedInterestAmt [159] has been added to the Security List message

- The UnsolicitedIndicator field has been added to the Security List message

6.3   Dynamic contracts information

6.3.1   Description

This functionality allows dynamic information to be requested on the prices for a number of contracts.

6.3.2   Information request

The request for dynamic information related to prices is made using the Market Data Request message.

A number of contracts can be selected using a combination of fields of the Instrument block as explained in 4.3. As it can be seen in the detailed message description, various Instrument blocks can be included to make more than one selection simultaneously. A contract is considered selected if any of the selection criteria are met.

The types of information offered by MEFF are listed below. A client can request a combination of these types of information in the same request.

- Settlement price

- Trade Volume

- Open position at the end of the previous session

The settlement price (Settlement Price field) information provided during the session includes the settlement price of the contract for the previous session. Once the settlement price of the current session has been fixed, this value will be used. The OpenCloseSettleFlag field shows when the closing price has been fixed.

The request may be just the current situation (snapshot) or include updates (snapshot + update).

6.3.3   Receipt of information

MEFFGate sends the information requested in Market Data Snapshot Full Refresh messages.

If information updates have been requested, a new Market Data Snapshot Full Refresh message will be received every time there is a change. This message contains both the information that has changed as well as the rest of the fields requested in the subscription.

6.3.4   List of messages

Message	Description
Market Data Request (Msg Type = V)	Sent by the client to request price information
Market Data Snapshot Full Refresh (Msg Type = W)	Sent by the server to return price information
Market Data Request Reject (Msg Type = Y)	Sent by the server to notify that a Market Data Request has been rejected

6.3.5 Message flow

Dynamic contracts information request with updates

A dynamic contracts information request, without updates, is answered by MEFFGate with a message for each of the contracts.

MEFFGate Client   MEFFGate Server

Market Data Request ("V")

SubscriptionRequestType [263] = 0 (Snapshot)

Market Data - Snapshot / Full refresh ("W")

(One message for every contract requested)
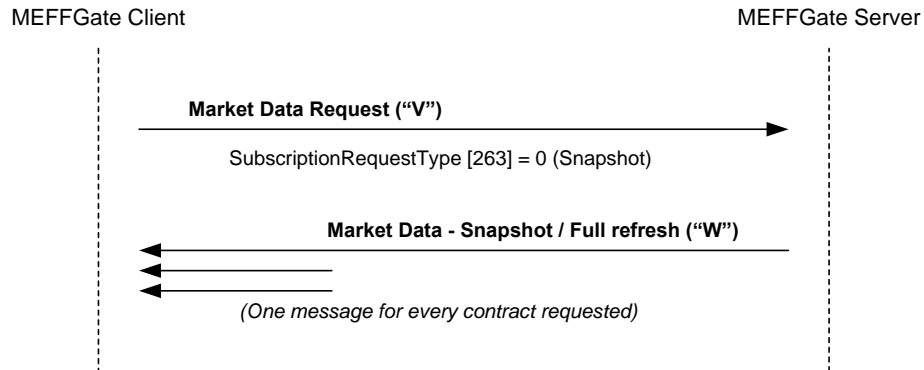
Request for dynamic contracts information with updates

A request for dynamic contracts information with updates initially receives a series of messages for the selected contracts at the time of the request. From this moment on it receives messages notifying changes that occur.

MEFFGate Client   MEFFGate Server

Market Data Request ("V")

SubscriptionRequestType [263] = 1 (Snapshot + Updates)

Market Data - Snapshot / Full refresh ("W")

(One message for every contract requested)

Snapshot

Market Data - Snapshot / Full refresh ("W")

…

Updates

Market Data Request ("V")

SubscriptionRequestType [263] = 2 (Unsubscribe)
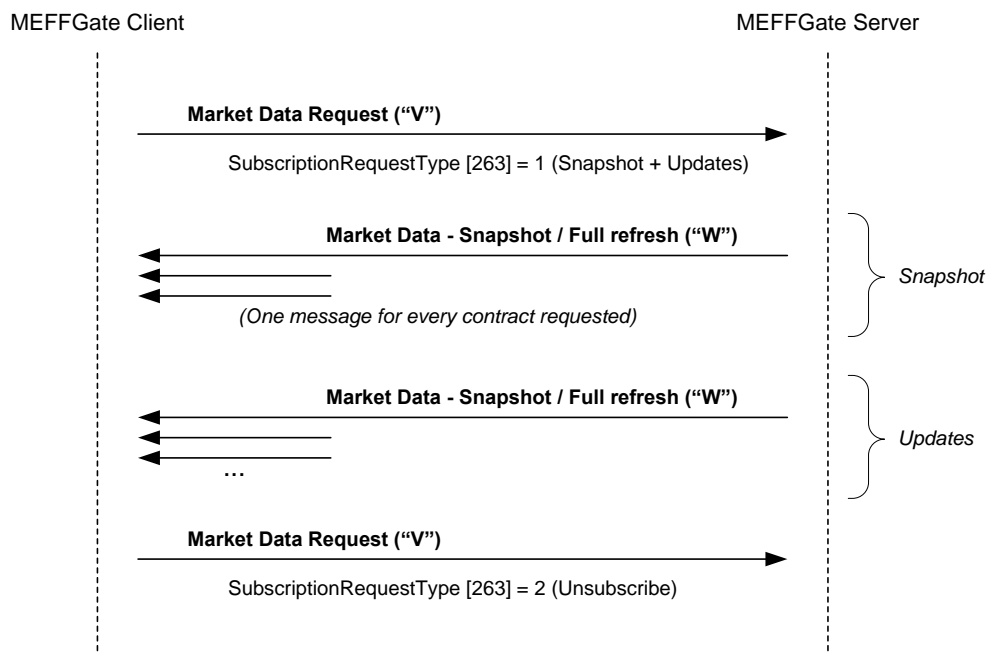
Note that a request to end the subscription only receives a reply if there is an error.

Incorrect dynamic contracts information request

When a dynamic contracts information request is incorrect the reply will be a Market Data Request Reject message.

MEFFGate Client	MEFFGate Server

Market Data Request ("V") →

← **Market Data Request Reject ("Y")**
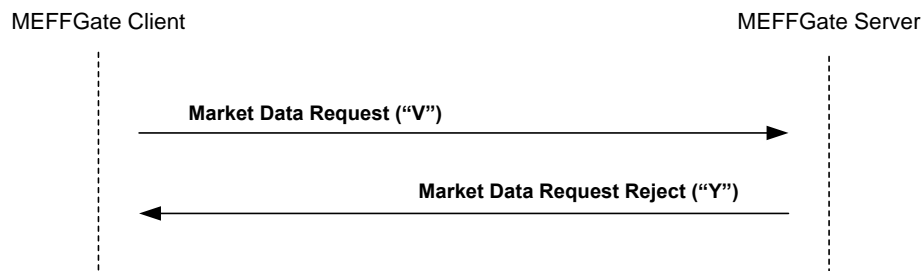
6.3.6 Annotations and adaptations of FIX 4.4

- The incremental update is not supported

- In the Market Data Request Reject message, the meaning of the value 0 (invalid symbol) in the MDReqRejReason field has been broadened to indicate an invalid selection criteria

6.4   Definition of messages

6.4.1   Security List Request (Msg Type = x)

Used by the client to request the contract definitions

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = x		
320	SecurityReqID	Y		String(10)	Unique identifier for each Security List Request message If SubscriptionRequestType = 2 or NewSecuritySubscription = 2, it must contain the value of the original request
559	SecurityListRequestType	Y	0 = Symbol 1 = Security type	Int	Selection criteria used
	Start <Instrument>				
55	Symbol	Y	[N/A] or contract code	String(22)	Contract code if SecurityListRequestType = 0, [N/A] if SecurityList-RequestType = 1
48	SecurityID	N	See table 21 in document "Codification Tables" for a list of possible values	String	Underlying asset Not allowed if SecurityListRequestType = 0
22	SecurityIDSource	N	8 = Exchange Symbol	String	Required if SecurityID is present. Not allowed if Security-ListRequestType = 0
	End <Instrument>				
5682 *	NewSecuritySubscription	N	0 = Snapshot (default value) 1= Updates (Subscribe) 2= Unsubscribe	Char	Indicates the type of contract definition request
	Standard Trailer	Y			

6.4.2 Security List (Msg Type = y)

Message sent by the server to provide the definition of one or more contracts.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = y		
320	SecurityReqID	Y		String	Identifier of Security List Request message that it is replying to
322	SecurityResponseID	Y		String	Unique identifier for each Security List message
560	SecurityRequestResult	Y	0=Valid request 1=Invalid or unsupported request 2=No instruments found that match selection criteria 4=Instrument data temporarily unavailable	Int	Result of request identified by SecurityReqID. If rejected (>0) there is an explanation in the Text [58] field
325*	UnsolicitedIndicator	N	N = The message is part of a snapshot Y = The message is sent due to an update (new contract)	Boolean	Contains "Y" when the message is sent due to a subscription
393	TotNoRelatedSym	N		Int	Total number of contracts that meet the selection criteria in the request. The number of contracts that the message contains is indicated in the NoRelated-Sym field. This field is always present when SecurityRequestResult = 0
893	LastFragment	N		Boolean	Indicates when the message is the last in a sequence in response to a single request. This field is always present when SecurityRequestResult = 0
146	NoRelatedSym	N	1	NumInGroup	Indicates the number of elements contained in this message.
	Start <Instrument>				
→55	Symbol	N	[N/A] or contract code	String(22)	Contract code. Present if NoRelatedSym has been specified and SecurityRequestResult [560] = 0
→48	SecurityID	N	See table 21 in document "Codification Tables" for a list of possible values	String	Underlying asset
→22	SecurityIDSource	N	8 = Exchange Symbol	String	
→454	NoSecurityAltID	N	1		
→→455	SecurityAltID	N		String	ISIN code for the contract
→→456	SecurityAltIDSource	N	4 = ISIN number	String	
→167	SecurityType	N	See Appendix B.1	String	Product type
→200	MaturityMonthYear	N	YYYYMM or	Month-	Contract expiration

Tag	Name	Req	Valid values	Format	Description
			YYYYMMDD or YYYYMMwW	Year	
→541	MaturityDate	N		LocalMktDate	Expiration date
→225	IssueDate	N		UTCDate	Date contract issued
→202	StrikePrice	N		Price	Exercise price. Only present for options
→231	ContractMultiplier	N		Float	Indicates the ratio or multiplier to convert "nominal" units (e.g. contracts) to total units (e.g. shares)
→1193	SettlMethod	N	C = Cash settlement required P = Physical settlement required	Char	Settlement method for this security
→1194	ExerciseStyle	N	0 = European 1 = American	Int	Type of exercise of this security
→201	PutOrCall	N	0 = Put 1 = Call	Int	Indicates whether an option contract is a put or call
→1244	FlexibleIndicator	N	Y = Flexible N = Standard (default)	Boolean	Used to indicate if this security has been defined as flexible according to "non-standard" means. When not informed, means "N = Standard "
→107	SecurityDesc	N		String	Description of the underlying asset. See table 21 in document "Codification Tables" for a list
→969	MinPriceIncrement	N		Float	Minimum amount allowed for price change. This field is always included in the message
→864	NoEvents	N		NumInGroup	
→→865	EventType	N	134 = Adjustments rule 166 = Settlement date of the expiry deferral,	Int	
→→866	EventDate	N		LocalMktDate	Settlement date of the expiry deferral, when EventType [865] = 166
→→868	EventText	N		String	It contains the adjustments rule when EventType = 134: E – Extraordinary dividend adjustments only (default) T - Total
	Start < ComplexEvents >				
→1483	NoComplexEvents	N	1	NumInGroup	
→→1484	ComplexEventType	N	16 = Foreign exchange cross currency	Int	
→→2124	ComplexEventCurrencyOne	N		Currency	Base currency code. Follows ISO 4217 standard

Tag	Name	Req	Valid values	Format	Description
→→ 2125	ComplexEventCurrencyTwo	N		Currency	Quoted currency code. Follows ISO 4217 standard
	End < ComplexEvents >				
	End <Instrument>				
	Start <StrikeRules>				
→1201	NoStrikeRules	N	1	NumInGroup	
→→122 3	StrikeRuleID	N	[N/A]	String	
	Start <MaturityRules>				
→→123 6	NoMaturityRules	N	1	NumInGroup	
→→→12 22	MaturityRuleID	N	[N/A]	String	
→→→13 02	MaturityMonthYearIncrementUnits	N	0 = Months 1 = Days 2 = Weeks 3 = Years	Int	Periodicity
→→→12 41	StartMaturityMonthYear	N	YYYYMMDD	Month-Year	Start delivery date for Energy segment contracts
→→→12 26	EndMaturityMonthYear	N	YYYYMMDD	Month-Year	End delivery date for Energy segment contracts
→→→12 29	MaturityMonthYearIncrement	N		Int	
	End < MaturityRules >				
	End <StrikeRules>				
→711	NoUnderlyings	N	1	NumInGroup	Present if the contract has another contract as its underlying
	Start <UnderlyingInstrument>				
→→311	UnderlyingSymbol	Y		String(22)	Symbol for underlying contract
	End <UnderlyingInstrument>				
→15	Currency	N		Currency	Currency code. Follows ISO 4217 standard
→555	NoLegs	N		NumInGroup	Present if the contract has deliverable securities
	Start <InstrumentLeg>				
→→600	LegSymbol	N		String	Symbol for deliverable contract
→→603	LegSecurityIDSource	N	8 = Exchange Symbol	String	
→→604	NoLegSecurityAltID	N	1		
→→→60 5	LegSecurityAltID	N		String	ISIN code for the delivery contract
→→→60 6	LegSecurityAltIDSource	N	4 = ISIN number	String	
→→253	LegFactor	N		Float	Conversion factor for deliverable contract
→→599	LegInstrRegistry	N	1 = IBERCLEAR	String	Code of the Central Security Depositary for deliverable contract
→→159*	AccruedInterestAmt	N		Amt	Accrued interest for deliverable contract
	End < InstrumentLeg >				
→58	Text	N		String	If SecurityRequestResult [560] > 0 there is an explanation of the rejection
	Standard Trailer	Y			

6.4.3 Market Data Request (Msg Type = V)

Used by the client to request price information.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = V		
262	MDReqID	Y		String(10)	Unique identifier for each Market Data Request message. If SubscriptionRequestType = 2 it must contain the identifier of the original request
263	SubscriptionRequestType	Y	0=Snapshot 1=Snapshot + Updates 2= Unsubscribe	Char	
264	MarketDepth	Y		Int	Ignored by MEFFGate
265	MDUpdateType	N	0 = Full refresh	Int	Required if SubscriptionRequestType = 1
267	NoMDEntryTypes	Y		NumInGroup	Number of MDEntryType fields that contain the message
→ 269	MDEntryType	Y	6 = Settlement Price B = Trade Volume (total volume for contract in session) C = Open Interest	Char	Type of market information requested
146	NoRelatedSym	Y	1	NumInGroup	Number of selection criteria
	Start <Instrument>				
→ 55	Symbol	Y	[N/A] or contract code	String(22)	Contract code
→ 48	SecurityID	N	See table 21 in document "Codification Tables" for a list of possible values	String	Underlying asset
→ 22	SecurityIDSource	N	8 = Exchange Symbol	String	Required if the SecurityID has been specified
	End <Instrument>				
	Standard Trailer	Y			

6.4.4 Market Data Request Reject (Msg Type = Y)

Used by MEFFGate to reject a Market Data Request.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = Y		
262	MDReqID	Y		String	Identifier of the request being rejected
281	MDReqRejReason	N	0 = Invalid selection criteria 1 = Duplicate MDReqID 4 = Unsupported SubscriptionRequestType 5 = Unsupported MarketDepth 6 = Unsupported MDUpdateType 8 = Unsupported MDEntryType	Char	Reason for rejection. This field is always present in the message
58	Text	N		String	Explanation of rejection motive
	Standard Trailer	Y			

6.4.5 Market Data Snapshot Full Refresh (Msg Type = W)

Used by MEFFGate to communicate price information requested with a Market Data Request message.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = W		
262	MDReqID	Y		String	Identifier of the Market Data Request message that is being replied to
	Start <Instrument>				
55	Symbol	Y	Contract code	String(22)	Contract code
	End <Instrument>				
268	NoMDEntries	Y		NumInGroup	Number of entries to follow
→269	MDEntryType	Y	6 = Settlement Price B = Trade Volume (total volume for contract in session) C = Open Interest	Char	Type of information that the present entry contains.
→270	MDEntryPx	N		Price	Price. Present when the MDEntryType is equal 6. When it is not present, a zero value should be understood
→236	Yield	N		Percentage	Trade price based on yield percentage
→271	MDEntrySize	N		Qty	Volume. Present when the MDEntryType is equal B or C
→286	OpenCloseSettleFlag	N	1 = Session Open / Close / Settlement entry 4 = Entry from previous business day 5 = Theoretical Price Value	MultipleValue String	When MDEntryType = 6, the values 1 and 4 are used to indicate if the closing price is from the previous session (value 4) or are the prices for the current session (value 1).
→811	PriceDelta	N		float	Maybe present if MDEntryType = 6
	Standard Trailer	Y			

7. Monitoring and Management of Positions

7.1 Introduction

In this chapter the functionalities related to the open position are covered. This includes as the request as the modification by means of position adjustment

7.2 Request of Open position

The client application can request the open position by account and contract by sending a Request for Positions message.

The open position for an account and contract reflects the situation of that account at the time of the request for the contract in question. MEFF provides this information divided into two parts:

- The open position at the beginning of the day

- The combined trades made from the beginning of the day until the time of the query

The open position at the start of the day is the position the account had that was used for the market valuation and calculation of position margins for the previous session. Only are given the account/contract combinations with open position different than zero.

The open position at the beginning of the day is reported by Position Report message.

The information on the trades made from the beginning of the day until the time of the request is reported using Trade Capture Report messages.

In addition to the position at the time of the request, the Request For Positions message allows a subscription to be made, requesting information updates for the open position. The updates of said information are made through the notification of trades made with Trade Capture Report messages. Note that when the update request is made, the trades of all the accounts will be received, whether or not they have positions at the time of the request.

The request can include a series of selection criteria, such as the clearing member, the contract type, the specific instrument, etc. The information received in the reply corresponds only to the account-contract pairs that meet the criteria specified in the request.

Should the request have been made with a subscription, once the end of the day arrives and the position is definitive, MEFFGate sends a Position Report message, for each account included in the request, informing of the open position at that moment. The same message will also be included in the answer to a request without subscription made after the end of day.

7.3 Request by trading and/or clearing member

In the request messages Request For Positions allows to make the request by trading member and/or clearing member using the Parties block.

Whether or not a user of clearing member will be able to see the trading activity of a member that it clears will depend on the user permissions.

When these fields are not specified in a request it is considered that the request only refers to the own accounts of the member to which the trader who makes the request belongs

When a clearing member wishes to query the accounts position that it clears it will have to include its own member code in the PartyID field associated with PartyRole 4 (Clearing Firm). If in addition it wishes to restrict this query per trading member, it can use the PartyID field associated to PartyRole 13 (Order Origination Firm).

In the following table different examples of selections are given and their interpretation. In these examples Member A is a clearing member that makes the query and Member B is a trading member cleared by A.

Trading Member (parties block with PartyRole = 13)	Clearing Member (parties block with PartyRole = 4)	Interpretation
-	-	Selection of accounts of Member A
Member A	-	Selection of accounts of Member A
-	Member A	Selection of accounts cleared by Member A, belonging to any trading member that it clears including its own
Member B	Member A	Selection of accounts cleared by Member A, belonging to Member B
Member B	-	Erroneous query. If a different code is given other than its trading member code, the clearing member code must be given
-	Member B	Erroneous query. If a code other than the own one is given as clearing Member, the trading member should be specified with the self trading code

In the following table some examples are given, where the request is done by the trading member B. Some of the Member B accounts have the Member A as clearing member.

Trading Member (parties block with PartyRole = 13)	Clearing Member (parties block with PartyRole = 4)	Interpretation
-	-	Selection of his own accounts (Member B)
Member B	-	Selection of his own accounts (Member B)
Member B	Member A	Selection of his own accounts (Member B) with Member A as clearing member
-	Member A	Erroneous query. If a code other than the own one is given as clearing member, the user member code must be present

7.4 Position adjustments

To request a position adjustment, the client application should send a Position Maintenance Request message indicating the account and contract that it wishes to adjust and the number of contracts desired to reduce the position.

MEFFGate responds to this request with a Position Maintenance Report message informing of the acceptance or rejection of the request. Irrespective of the acceptance or rejection, this message informs of the position for the account and contract of reference at the moment of emission of the message.

In MEFF, the position adjustment is implemented by two closing position trades of opposite sign and the same volume as the adjustment. Users subscribed to the Trade Capture Report messages will receive the messages associated with these trades.

7.5   List of messages

Message	Description
Request For Positions (Msg Type = AN)	Request for information on open position
Request For Positions Ack (Msg Type = AO)	Acknowledgement of Request For Positions message
Position Report (Msg Type = AP)	Information on open position in the CCP for a contract and account
Trade Capture Report (Msg Type = AE)	Information on a trade registered in the CCP
Position Maintenance Request (Msg Type = AL)	Request for position adjustment
Position Maintenance Report (Msg Type = AM)	Report of an acceptance or rejection of a position adjustment

7.6   Message flow

Request open position without update

The following example shows the messages corresponding to an account-contract combination. If the selection criteria refer to more than one combination, the same messages for each of them will be received.
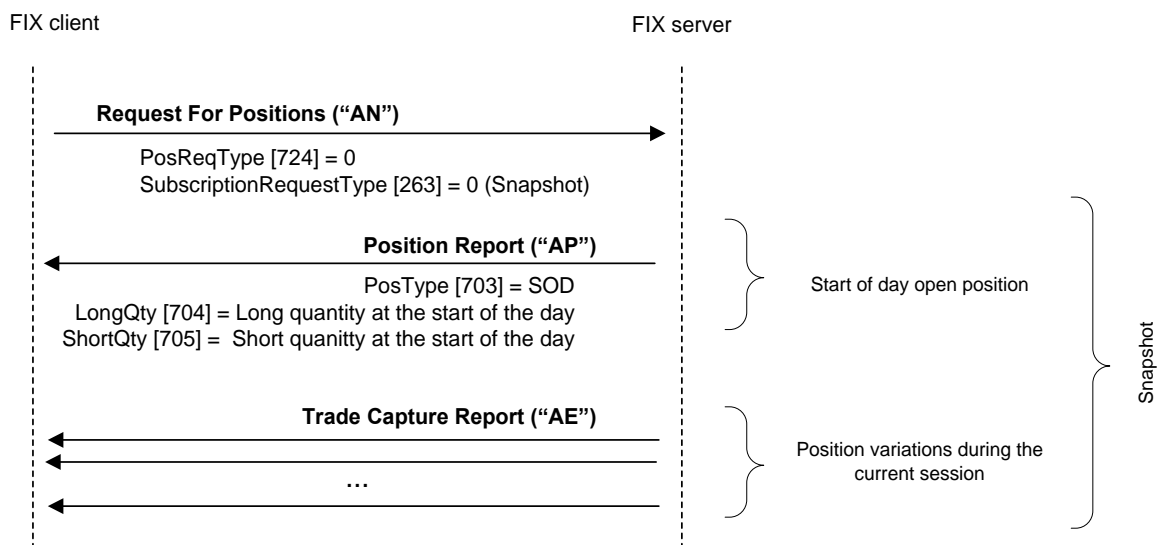
Open position request with updates.

The following example shows the messages corresponding to an account-contract combination. If the selection criteria refer to more than one combination, the same messages for each of them will be received.
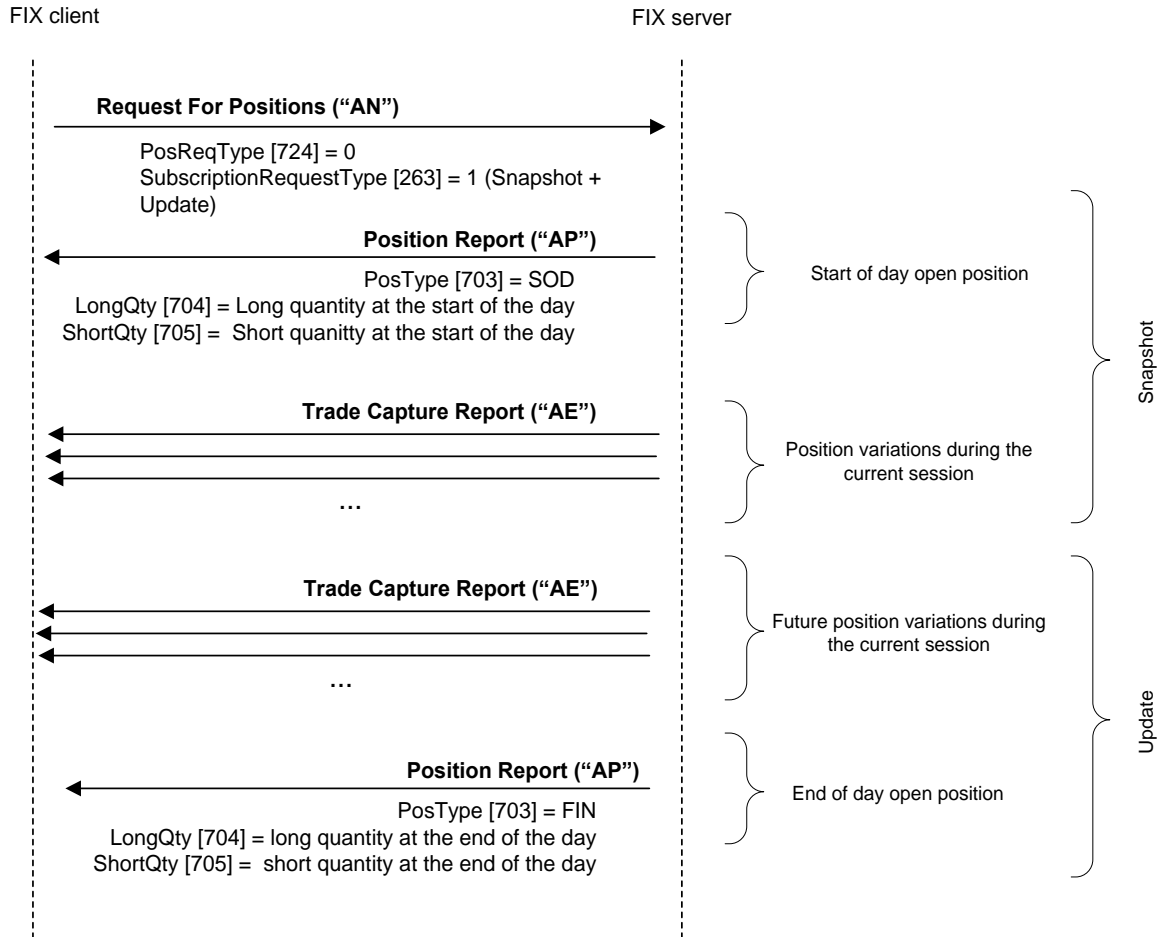
Acceptance of a request for position adjustment

Request of a position adjustment for *n* contracts. When the message is processed by the system the account has L and N contracts bought and sold respectively

```
         FIX client                                      FIX server
             |                                                |
             |     Position Maintenance Request ("AL")        |
             |----------------------------------------------->|
             |   PosTransType [709] = 3 (Position Adjustment);|
             |        PosType [703]= PA (Adjusment Qty);       |
             |              LongQty [704] = n                   |
             |                                                |
             |      Position Maintenance Report ("AM")        |
             |<-----------------------------------------------|
             |  PosTransType [709] = 3 (Position Adjustment);  |
             |       PosMaintStatus [722] = 0 (Accepted)        |
             |      PosType [703] = PA (Adjusment Qty)          |
             |              LongQty [704] = n                    |
             |              PosType [703] = TOT                  |
             |              LongQty [704] = L - n                |
             |             ShortQty [705] = S - n               |
             |                                                |
             |         Trade Capture Report ("AE")            |
             |<-----------------------------------------------|
             |              LastQty [32] = n                    |
             |              Side[54] = 1 (Buy)                  |
             |       PositionEffect[77] = "C" (Close)           |
             |                                                |
             |         Trade Capture Report ("AE")            |
             |<-----------------------------------------------|
             |              LastQty [32] = n                    |
             |              Side[54] = 2 (Sell)                 |
             |       PositionEffect[77] = "C" (Close)           |
             |                                                |
```

Rejection of Request for position adjustment

```
         FIX client                                      FIX server
             |                                                |
             |     Position Maintenance Request ("AL")        |
             |----------------------------------------------->|
             |  PosTransType [709] = 3 (Position Adjustment);  |
             |       PosType [703] =  PA (Adjusment Qty);       |
             |              LongQty [704] = n                   |
             |                                                |
             |      Position Maintenance Report ("AM")        |
             |<-----------------------------------------------|
             |  PosTransType [709] = 3 (Position Adjustment);  |
             |       PosMaintStatus [722] = 2 (Rejected)        |
             |                                                |
```
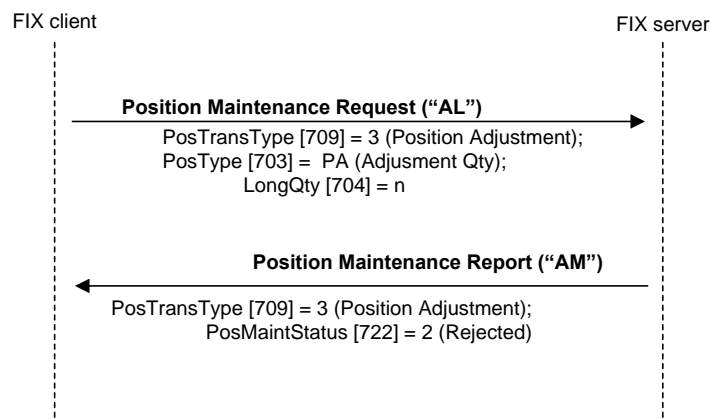
7.7   Annotations and adaptations of FIX 4.4

- The ExchangeTradeType [5681] user field has been added to the Trade Capture Report message

- The LeavesQty [151] and PosReqID [710] fields have been added to the Trade Capture Report message

- The LegSymbol [600] field has been added to the Position Maintenance Request and Position Maintenance Report messages

- The Trade Capture Report message only informs on one of the parties involved in the trade. Specifically, it informs on the party that the message is being sent to

- In the Request For Positions message the Parties Block does not have to be included. In this case the trading member's own code will be considered selected.

7.8   Definition of messages

7.8.1   Request For Positions (Msg Type = AN)

Message sent by the client to request the open position, with or without updates.

It can also be used to request the status of exercise instructions (see chapter 10).

Tag	Name	Req	Accepted values	Format	Description
	Standard Header	Y	MsgType = AN		
710	PosReqID	Y		String	Unique identifier for each Request For Positions message
724	PosReqType	Y	0 = Positions 2 = Exercises	Int	Indicates the type of transaction request
263	SubscriptionRequestType	N	0=Snapshot (default value) 1=Snapshot + Updates 2= Unsubscribe	Char	Request type
→	Start <Parties>				
→ 453	NoPartyIDs	Y		NumInGroup	The Parties block is required by the standard. MEFFGate does not require this block to be present. In this case the trading member's own code will be considered selected.
→→ 448	PartyID	Y		String	MEFF member code
→→ 447	PartyIDSource	Y	D = Proprietary / Custom code	Char	Required if NoPartyIDs has been specified
→→ 452	PartyRole	Y	13 = Order Origination Firm 4 = Clearing Firm	Int	Indicates the role taken by the code specified in PartyID. Required if NoPartyIDs has been specified
→	End <Parties>				
1	Account	Y	Fixed length	String(5)	Account code. The use of the wildcard "?" for the multiple selection is only allowed in the five positions at a time or in the last two positions, in which case it must be used for both at the same time
581	AccountType	Y	1	Int	Ignored by MEFFGate
	Start <Instrument>				
55	Symbol	Y	[N/A] or contract code	String(22)	Contract code (When PosReqType=2 must refer to an option)
48	SecurityID	N	See table 21 in document "Codification Tables" for a list of possible values	String	Underlying asset
22	SecurityIDSource	N	8 = Exchange Symbol	String	Required if the SecurityID has been specified
	End <Instrument>				
715	ClearingBusinessDate	Y		LocalMktDate	Ignored by MEFFGate

Tag	Name	Req	Accepted values	Format	Description
60	TransactTime	Y		UTCTimestamp	Time at which request was generated
	Standard Trailer	Y			

7.8.2 Request For Positions Ack (Msg Type = AO)

Used by the server to notify the result of a Request For Positions for an open position query. It precedes the information about the position itself.

Given that the Request For Positions message can be used to request the status of the exercise instructions, this message is also used to reply to this request (see chapter 10 for more information about exercise instructions).

Tag	Name	Req	Accepted values	Format	Description
	Standard Header	Y	MsgType = AO		
721	PosMaintRptID	Y		String	Unique identifier for each Request For Positions Ack message
710	PosReqID	N		String	Identifier of the Request For Positions message, as specified in the message. This field is always present in the message
727	TotalNumPosReports	N		Int	Number of messages that are going to be received in reply to the corresponding Request For Positions.
728	PosReqResult	Y	0 = Valid Request 1 = Invalid or unsupported Request 2 = No positions found that match criteria 4001 = Duplicate PosReqID 4002 = Invalid Account or Symbol	Int	Result of the Request For Position
729	PosReqStatus	Y	0 = Completed 2 = Rejected	Int	Request status
1	Account	Y	Fixed length	String(5)	Contains the same value as indicated in the request
581	AccountType	Y	1	Int	Contains the same value as indicated in the request
58	Text	N		String	Explanation if there is an error
	Standard Trailer	Y			

7.8.3 Position Report (Msg Type = AP)

Message that notifies the open position for an account and contract, in reply to a Request For Positions message with the field PosReqType = 0 (request of open position).

This message is also used to notify the exercise instructions, in reply to a Request For Positions message with the field PosReqType = 2 (request of exercise instructions). See chapter 10 for more information on exercise instructions.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = AP		
721	PosMaintRptID	Y		String	Unique identifier for each Position Report message
710	PosReqID	N		String	Identifier of the Request For Positions message, as specified in the message. This field is always present in the message
724	PosReqType	N	0 = Positions 2 = Exercises	Int	Contains the same value as indicated in the request
325	UnsolicitedIndicator	N	N = The message is part of a snapshot Y = The message is sent as the result of a subscriptcion	Boolean	Contains "Y" when the message is sent as the result of a subscription
727	TotalNumPosReports	N		Int	Number of account-contract pairs on which information is provided. Note that no information is provided for accounts without positions or exercise requests
728	PosReqResult	Y	0 = Valid Request	Int	Result of Request For Positions. Note that erroneous requests are answered with a Request For Positions Ack message
715	ClearingBusinessDate	Y		LocalMktDate	The content of this field should not be considered, and is included as requirement of the standard
	Start <Parties>				
453	NoPartyIDs	N		NumInGroup	Number of participants to be informed
→ 448	PartyID	N		String	Member code
→ 447	PartyIDSource	N	D = Proprietary/ Custom code	Char	
→ 452	PartyRole	N	13 = Order Origination Firm	Int	Indicates the role taken by the code specified in PartyID
	End <Parties>				
1	Account	Y	Fixed length	String(5)	Account code on which information is provided
581	AccountType	Y	1	Int	Contains the same value as indicated in the request
	Start <Instrument>				
55	Symbol	Y	Contract code	String(22)	Contract on which information is provided
	End <Instrument>				
730	SettlPrice	Y		Price	Closing price of the contract . It contains the value from the previous session until the value for the current session has been determined. The field SettlPriceType allows to differentiate between these two cases

Tag	Name	Req	Valid values	Format	Description
731	SettlPriceType	Y	1 = Final 2 = Theoretical	Int	Indicates if the SettlPrice fields refers to the previous session (value 2) or the current session (value 1)
734	PriorSettlPrice	Y		Price	Previous closing price in respect to SettlPrice
	Start <PositionQty>				
702	NoPositions	Y		NumInGroup	
→ 703	PosType	Y	SOD = Start of Day Position FIN = End-of-Day Qty EX = Option Exercise Qty TOT = Total Transaction Qty	String	SOD or FIN is used when the message is a reply to open positions report request. EX and TOT both are used when the message is a reply to a request for report on exercise instructions. SOD will be present in the message that reports the open position at the beginning of the day. FIN will be present in the message that reports the open position at the end of the day
→ 704	LongQty	N	>= 0, no decimals	Qty	When PosType = SOD or FIN, it indicates the number of contracts that make up the long position of the position being reported. When PosType = EX it indicates the number of options to exercise. When it is not present and PosType = EX the message informs of the cancellation of an exercise instruction. When PosType = TOT, it indicates the number of contracts that make up the long position at the time the message is sent. Only present if there is a long position.
→ 705	ShortQty	N	> 0, no decimals	Qty	When PosType = SOD or FIN, it indicates the number of contracts that make up the short position of the position being reported. Not used when PosType = EX. When PosType = TOT, it indicates the number of contracts that make up the short position at the time the message is sent. Only present if there is a short position.
	End <PositionQty>				

Tag	Name	Req	Valid values	Format	Description
	Start <PositionAmountData>				The content of this block should only be considered in Mark-to-Market contracts
753	NoPosAmt	N		NumInGroup	
→ 707	PosAmtType	N	SMTM = Start-of-Day Mark-to-Market Amount FMTM = Final Mark-to-Market Amount	String	
→ 708	PosAmt	N		Amt	When PosAmtType = SMTM is the Start-of-Day Mark-to-Market position amount. When PosAmtType = FMTM is the Final Mark-to-Market position amount.
	End <PositionAmountData>				
	Standard Trailer	Y			

7.8.4 Trade Capture Report (Msg Type = AE)

Message containing data on a CCP trade. This message can be a reply to a Trade Capture Report Request message or a Request For Positions message.

When the message is a reply to a Request For Positions and it corresponds to a trade from a previous trading session, some of the compulsory fields may contain values that should not be considered; this is due to the fields being compulsory but the information not being available to populate these fields. The explanations given for these fields below warn of this situation.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = AE		
571	TradeReportID	Y		String	Unique identifier for each Trade Capture Report message
568	TradeRequestID	N		String	Identifier of the request. Present when the request is a Trade Capture Report Request message
710*	PosReqID	N		String	Identifier of the request. Present when the request is a Request For Positions message
150	ExecType	N	F = Trade	Char	Trade type
912	LastRptRequested	N		Boolean	Indicates if it is the last message of the snapshot reply to a request made with a Trade Capture Report Request message
325	UnsolicitedIndicator	N	N = The message is part of a snapshot Y = The message is sent as result of an update	Boolean	Contains "Y" when the message is sent as the result of a subscription
881	SecondaryTradeReportRefID	N		String	Contains the CCP trade number of the previous trade
818	SecondaryTradeReportID	N		String	CCP register number. This field is always present in the message
820	TradeLinkID	N		String	Primary original trade reference. For repos, it's the common reference for both legs.
880	TrdMatchID	N		String	Contains the CCP trade number of the initial trade
17	ExecID	N		String	Secondary original trade reference. For repos, it refers to one of its legs.
527	SecondaryExecID	N		String	Trade register number of the initial trade. Coincides with the TrdMatchID [880] field of the Execution Report message.
5681*	ExchangeTradeType	N	See table 19 in document "Codification Tables" for details of the Trade Type	String	Trade type

Tag	Name	Req	Valid values	Format	Description
			codes		
	Start <RegulatoryTradeIDGrp>				
1907	NoRegulatoryTradeIDs			NumInGroup	
→1903	RegulatoryTradeID	N		String	It contains the UTI (Unique Trade Identifier)
→1905	RegulatoryTradeIDSource	N	BMCL-UTI	String	Identifies the reporting entity that originated the value in RegulatoryTradeID [1903]
	End <RegulatoryTradeIDGrp>				
570	PreviouslyReported	Y	Y	Boolean	Indicates if the trade was notified to the counterparty
	Start <Instrument>				
55	Symbol	Y	Contract code	String(22)	MEFF contract code
→864	NoEvents	N		NumInGroup	
→→865	EventType	N	101 = Nominal trade value (Par amount) 102 = Member code acting in market where delivery takes place 103 = Counterparty member code in market where delivery takes place	Int	
→→868	EventText	N		String	If EventType = 101, it contains the Nominal trade value (Par amount). Present in all spot trades (deliveries) If EventType = 102, it contains the Member code acting in market where delivery takes place. Present in spot trades broken down by member (deliveries) If EventType = 103, it contains the Counterparty member code in market where delivery takes place. Present in spot trades broken down by member (deliveries)
	End <Instrument>				
236	Yield	N		Percentage	Trade price based on yield percentage
32	LastQty	Y	>= 0, no decimals	Qty	Volume bought/sold in the trade described.
151*	LeavesQty	N		Qty	Alive volume of the trade. Only present when the message is the answer to a alive trades request
31	LastPx	Y		Price	Price of trade described.
75	TradeDate	Y		LocalMktDat	Local date of the market,

Tag	Name	Req	Valid values	Format	Description
				e	where trade was made.
60	TransactTime	Y		UTCTimestamp	Date and time, in UTC format, when transaction was made.
	Start <TrdRegTimestamps>				
768	NoTrdRegTimestamps	N	1	NumInGroup	Number of TrdRegTimestamps entries
→ 769	TrdRegTimestamp	N		UTCTimestamp	Timestamp when the initial trade was made in the Trading System
→ 770	TrdRegTimestampType	N	3 = Time Out	Int	
	End <TrdRegTimestamps>				
63	SettlType	N	0 = Regular (default value) 2 = Next Day (T + 1)	Char	Indicates the settlement type for the trade
552	NoSides	Y	1	NumInGroup	Always 1 as it only includes the buyer or seller part, depending whom the message is sent to.
→ 54	Side	Y	1=Buy 2=Sell	Char	Position that the party takes in the trade
→ 37	OrderID	Y		String	Order number, unique per trading operator. Contains "NONE" when the trade is not the result of the execution of an order
→ 198	SecondaryOrderID	N		String	Unique identifier of the order of the initial trade as assigned by the MEFF central systems or another exchange.
→	Start <Parties>				
→453	NoPartyIDs	N	1	NumInGroup	
→→ 448	PartyID	N		String	
→→ 447	PartyIDSource	N	D = Proprietary/ Custom code	Char	Present if NoPartyIDs has been specified
→→ 452	PartyRole	N	12 = Executing Trader 13 = Order Origination Firm	Int	Indicates the role taken in the PartyID code specified. Present if NoPartyIDs has been specified 12: MEFF Member Executing Trader Code 13: MEFF Member Code to which the associated account belongs
	End <Parties>				
→ 1	Account	N		String(5)/(3)	5 characters: Account associated with the trade. Present in subscriptions that are <u>not</u> using the specific option TrdType [828] = 15 in the Trade Capture Report Request message. 3 characters (only for

Tag	Name	Req	Valid values	Format	Description
					deliveries. See section 13.4 for more details): Holder associated with the trade. Used in spot trades broken down by holder (deliveries). Present in subscriptions that are using the specific option TrdType [828] = 15 in the Trade Capture Report Request message. Not informed (only for deliveries. See section 13.4 for more details): Used in spot trades broken down by member (deliveries). Present in subscriptions that are using the specific option TrdType [828] = 15 in the Trade Capture Report Request message.
→ 381	GrossTradeAmt	N		Amt	Nominal/Effective amount traded
→ 77	PositionEffect	N	"O" = Open "C" = Close		It indicates if the trade opens or closes the position
→ 58	Text	N		String	When this trade comes from a market order, contains the text entered as a reference for the order. When this trade comes from a cross-order, contains the text entered as a reference by the broker. When this trade is an allocation of daily account or a transfer, contains the same text as the original trade
→136	NoMiscFees	N	0, 1, 2	NumInGroup	
→→ 137	MiscFeeAmt	N		Amt	Fees amount. Contains a positive value when it is a payment from the Member to the Market. Contains a negative value when it is a payment form the Market to the Member
→→ 138	MiscFeeCurr	N		Currency	Currency in which fees are Codified according to Standard 4217
→→ 139	MiscFeeType	N	4=Exchange Fees 7=Other	Char	Fee type. It contains value 4 when it is a trading fee. It contains value 7 when it is a clearing fee.
→→ 891	MiscFeeBasis	N	0 = Absolute	Int	
	Standard Trailer	Y			

7.8.5 Position Maintenance Request (Msg Type = AL)

Message used by the client application to give an adjustment instruction on an account and contract

This message also allows the client application to give an exercise instruction on an account and contract, or to cancel instructions given. Sending a new instruction can be done to replace previous instructions.

Also is used to send a delivery notice.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = AL		
710	PosReqID	Y		String	Unique identifier for each Position Maintenance Request message
709	PosTransType	Y	1 = Exercise 3 = Position Adjustment 4 = Delivery	Int	Indicates the transaction type for the position
712	PosMaintAction	Y	1 = New 2 = Replace 3 = Cancel	Int	In the case of the exercise instructions indicating a specified request the value = 1 or should the request indicate a return to the automatic behaviour the value = 3. In the case of a position adjustment the value must always = 1. In the case of a delivey the value must always = 2. Please note the last message completely replaces the previous one.
713	OrigPosReqRefID	N		String	When PosTransType [709] = 4 (Delivery): Reference to the PosReqID [710] of a previous maintenance request that is being replaced. When PosTransType [709] = 1 or 3: Ignored by MEFFGate
715	ClearingBusinessDate	Y		LocalMktDate	Ignored by MEFFGate
1	Account	Y		String(5)/(3)	Account code for adjustment position or exercise. When deliveries, only the holder is informed (first three positions of the MEFF member account)
581	AccountType	Y	1	Int	Ignored by MEFFGate
	Start <Instrument>				
55	Symbol	Y		String(22)	Contract code for adjustment position, exercise or delivery notice. In case of exercise instruction should refer to an option contract
	End <Instrument>				
60	TransactTime	Y		UTCTimestamp	Ignored by MEFFGate
	Start <PositionQty>				
702	NoPositions	N		NumInGrou	This block is required except

Tag	Name	Req	Valid values	Format	Description
				p	for PosMaintAction [712] =3 (Cancel)
→ 703	PosType	N	EX = Option Exercise Qty PA= Adjustment Qty DN = Delivery Notice Qty	String	It should contain "EX" when PosTransType = 1, "PA" when PosTransType = 3 and "DN" when PosTransType = 4. Required if NoPositions has been specified
→ 704	LongQty	N	> 0, no decimals	Qty	When PosType=EX contains the maximum volume to exercise. The value 999999999 means "exercise all" and this is the default value. When PosType=PA contains the number of contracts to adjust position.
→ 705	ShortQty	N	> 0, no decimals	Qty	When PosType = DN, it indicates the number of contracts that are delivered
	Start <NestedParties>				
→ 539	NoNestedPartyIDs	N		NumInGroup	Number of participants to be informed
→→ 524	NestedPartyID	N		String	See NestedPartyRole [538]
→→ 525	NestedPartyIDSource	N	D = Proprietary/ Custom code	Char	
→→ 538	NestedPartyRole	N	13 = Order Origination Firm 4 = Clearing Firm 21 = Clearing Organization	Int	When 13 indicates a delivery notice made by the Trading Member. When 4 indicates a delivery notice made by the Clearing Member. When 21 indicates a delivery notice made by the Clearing House (MEFF).
	End < NestedParties >				
→ 600*	LegSymbol	N		String	Symbol for deliverable contract
	End <PositionQty>				
	Standard Trailer	Y			

7.8.6 Position Maintenance Report (Msg Type = AM)

Sent by MEFFGate to inform on the result of a Position Maintenance Request message.

This message is only sent to the user that makes the request. To monitor both the position adjustments, the exercise instructions and the delivery notices, the Request For Positions message with subscription, should be used.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = AM		
721	PosMaintRptID	Y		String	Unique identifier for each Position Maintenance Request message
709	PosTransType	Y	1 = Exercise 3 = Position Adjustment 4 = Delivery	Int	Contains the same value as specified in the request
710	PosReqID	Y		String	Identifier assigned by the client in the Position Maintenance Request message
712	PosMaintAction	Y	1 = New 2 = Replace 3 = Cancel	Int	Contains the same value as specified in the request
713	OrigPosReqRefID	Y		String	Contains the same value as specified in the request
722	PosMaintStatus	Y	0 = Accepted 2 = Rejected	Int	Status of Position Maintenance Request message. If the value is 2, PosMaintResult contains a more specific code
723	PosMaintResult	N	0 = Success 4001 = Duplicate PosReqID 4002 = Invalid Account or Symbol 4003 = Invalid quantity 4004 = Nothing to revoke (only for PosMaintAction = 3) 4005 = Action not allowed for this contract code 99 = Other	Int	Result of Position Maintenance Request message
715	ClearingBusinessDate	Y		LocalMktDate	This field should not be considered, and is included as requirement of the standard
1	Account	Y		String(5)	Contains the same value as specified in the request
581	AccountType	Y	1	Int	This field should not be considered, and is included as requirement of the standard
	Start <Instrument>				
55	Symbol	Y	Contract code	String(22)	Contains the same value as specified in the request

Tag	Name	Req	Valid values	Format	Description
	End <Instrument>				
60	TransactTime	Y		UTCTimestamp	This field should not be considered. Its inclusion is a requirement of the standard
	Start <PositionQty>				
702	NoPositions	Y		NumInGroup	
→703	PosType	Y	EX = Option Exercise Qty PA = Adjustment Qty DN = Delivery Notice Qty DLV = Total Delivery Qty TOT = Total Transaction Qty	String	The EX value is only included when it is an answer to an exercise instructions. Value PA is only included when it is an answer to a request of position adjustment. Values DN and DLV are only included when it is an answer to a request of deliveries.
→704	LongQty	N		Qty	When PosType=EX contains the number of contracts requested to exercise. When PosType=PA contains the number of contracts requested to adjust. When PosType=TOT contains the number of bought contracts that make the current position. In case of a request of adjustment of accepted position, this one is the value after the adjustment
→705	ShortQty	N		Qty	When PosType = DN, it indicates the number of contracts that are delivered. When PosType = DLV, it indicates the number of contracts to be delivered. When PosType=TOT. It contains the number of sold contracts that make the current position. In case of a request of adjustment of accepted position, this one is the value after the adjustment.
	Start <NestedParties>				
→ 539	NoNestedPartyIDs	N		NumInGroup	Number of participants to be informed

Tag	Name	Req	Valid values	Format	Description
→→ 524	NestedPartyID	N		String	See NestedPartyRole [538]
→→ 525	NestedPartyIDSource	N	D = Proprietary/ Custom code	Char	
→→ 538	NestedPartyRole	N	13 = Order Origination Firm 4 = Clearing Firm 21 = Clearing Organization	Int	When 13 indicates a delivery notice made by the Trading Member. When 4 indicates a delivery notice made by the Clearing Member. When 21 indicates a delivery notice made by the Clearing House (MEFF).
	End < NestedParties >				
→ 600*	LegSymbol	N		String	Symbol for deliverable contract
	End <PositionQty>				
	Start <PositionAmountData>				
753	NoPosAmt	Y	0	NumInGroup	This field should not be considered, and it is included as requirement of the standard
	End <PositionAmountData>				
58	Text	N			Explanation. Provided if PosMaintResult is not 0
	Standard Trailer	Y			

8. Trades Report

8.1 Introduction

This chapter describes the mechanisms offered by MEFF FIX interface to request the trades registered in the CCP.

The client application can request to query on the trades registered with a Trade Capture Report Request message

The whole trades reported are determined by the following selection:

- Parties Block. Allows to select by clearing member and trading member

- TradeDate field. Allows to indicate the period of reference:

    o   Session trades. In this case the request message should not contain this field.

    o   All the trades with live volume, independent of the clearing date. In this case the message should contain the value "19800101" in TradeDate field

- Side field. If this field is included in the request message, its value indicates if buying or selling trades are requested

In reply to this request a Trade Capture Report message is received for each trade that meets the selection criteria. If there are no trades that meet the criteria, a Trade Capture Report Request Ack message is received indicating this.

The request can be made in two ways, as indicated in the SubscriptionRequestType field:

- **SubscriptionRequestType = 0 (Snapshot)**. A message is received for each trade that meets the selection criteria, registered up until when the request is made

- **SubscriptionRequestType = 1 (Snapshot plus update)**. The same messages as the Snapshot request are received and from this moment on a new message for each trade that occurs that meets the selection criteria

Each message received has a unique CCP register number in the SecondaryTradeReportID field. When the trade being reported arises from a previous trade, the SecondaryTradeReportRefID field has the CCP trade number of the original trade. The CCP trade register is the same for the two parties to the trade. That is, if both the buyer and seller positions are taken, you must wait to receive two Trade Capture Report messages with the same SecondaryTradeReportID.

For transactions that result from trading, there are the SecondaryExecID, OrderID and SecondaryOrderID fields to make the front-to-back reconciliation.

When the Trade Capture Report message is the result of executing a cross trade, it will be related to the corresponding Execution Report message by the SecondaryExecID field in both, which will contain the same value.

8.2 Request by trading and/or clearing member

The Trade Capture Report Request message allows the making of a request by a trading member and/or clearing member using the Parties block.

Whether or not a user of clearing member will be able to see the trading activity of a member that it clears will depend on the user permissions.

When these fields are not specified in a request, it is considered that the request only refers to the trades of own accounts of the member that belong to the user who makes the request.

When a clearing member wants to query the trades of the accounts that it clears it will have to include its own member code in the PartyID field associated to PartyRole 4 (Clearing Firm). If in addition it wishes to restrict this query by trading member, it can use the PartyID field associated to PartyRole 13 (Order Origination Firm).

In the following table different examples of selection and the corresponding interpretations are given. In these examples Member A is a clearing member that makes the query and Member B is a trading member cleared by A.

Trading Member (parties block with PartyRole = 13)	Clearing Member (parties block with PartyRole = 4)	Interpretation
-	-	Selection of trades in accounts of Member A
Member A	-	Selection of trades in accounts of Member A
-	Member A	Selection of trades of accounts cleared by Member A belonging to any trading member that it clears including its own
Member B	Member A	Selection of trades of accounts cleared by Member A belonging to Member B
Member B	-	Erroneous query. If a different code is given other than its trading member code, the clearing member code must be given
-	Member B	Erroneous query. If a code other than the own one is given as clearing Member, then the trading member should be specified with the self trading code

In the following table some examples are given, where the request is done by the trading member B. Some of the Member B accounts have as clearing member the Member A.

Trading Member (parties block with PartyRole = 13)	Clearing Member (parties block with PartyRole = 4)	Interpretation
-	-	Selection of trades on own accounts (Member B)
Member B	-	Selection of trades on own accounts (Member B)
Member B	Member A	Selection of trades on own accounts (Member B) with Member A as clearing member
-	Member A	Erroneous query. If a code other than the own one is given as clearing member, then the user member code must be present

8.3 List of messages

Message	Description
Trade Capture Report Request (Msg Type = AD)	Request for CCP trades
Trade Capture Report Request Ack (Msg Type = AQ)	Acknowledgement of Trade Capture Report Request message
Trade Capture Report (Msg Type = AE)	Information on a trade registered in the CCP

8.4 Message flow

Request CCP trades with update subscription, followed by subscription cancellation

After the request to receive the messages corresponding to the trades registered up to that moment (snapshot) and to receive the new trades as they are registered from that point on.

MEFFGate Client   MEFFGate Server

Trade Capture Report Request ("AD")

SubscriptionRequestType [263] = 1 (Snapshot + Updates)

Trade Capture Report ("AE")

...   *One message for every trade in the snapshot*

Trade Capture Report ("AE")

...   *One message for every update*

Trade Capture Report Request ("AD")

SubscriptionRequestType [263] = 2 (Unsubscribe)
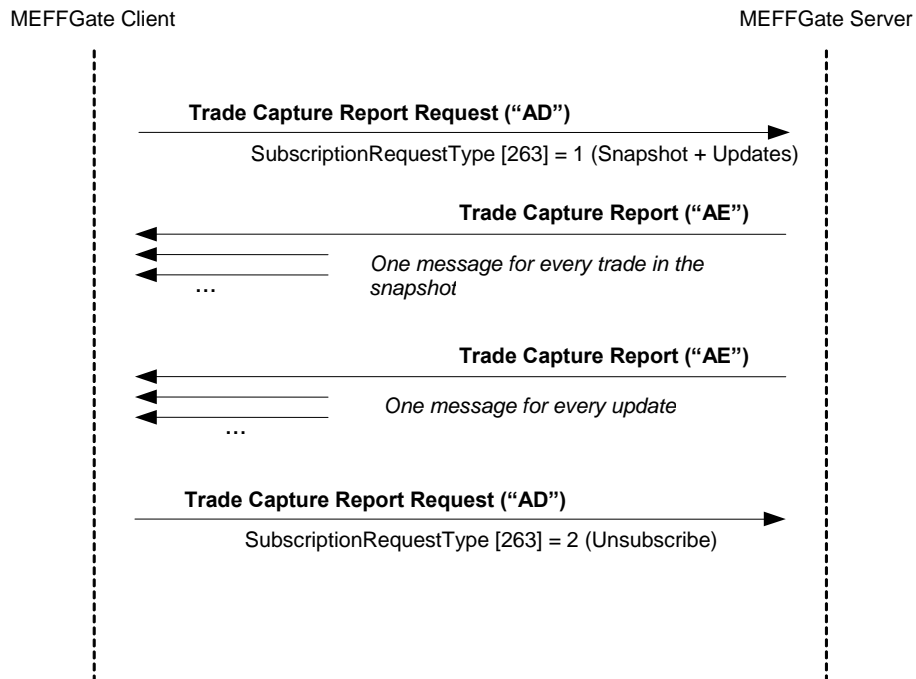
Request CCP trades without update subscription, without trades that meet selection criteria

When there are no trades that meet the selection criteria, the server sends a Trade Capture Report Request Ack message to notify the result of the request, with the value 0 in the TotNumTradeReports field.

MEFFGate Client   MEFFGate Server

Trade Capture Report Request ("AD")

SubscriptionRequestType [263] = 0 (Snapshot)

Trade Capture Report Request Ack ("AQ")

TotNumTradeReports [748] = 0
TradeRequestStatus [750] = 1 (Completed)
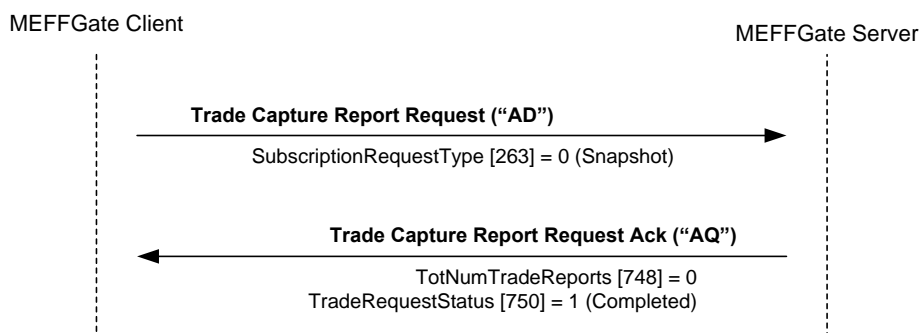
Request invalid CCP trades

When the CCP trade request (Trade Capture Report Request message) is invalid, the request is rejected in a Trade Capture Report Request Ack message.

MEFFGate Client   MEFFGate Server

Trade Capture Report Request ("AD")

SubscriptionRequestType [263] = 0 (Snapshot)

Trade Capture Report Request Ack ("AQ")

TradeRequestStatus [750] = 2 (Rejected)
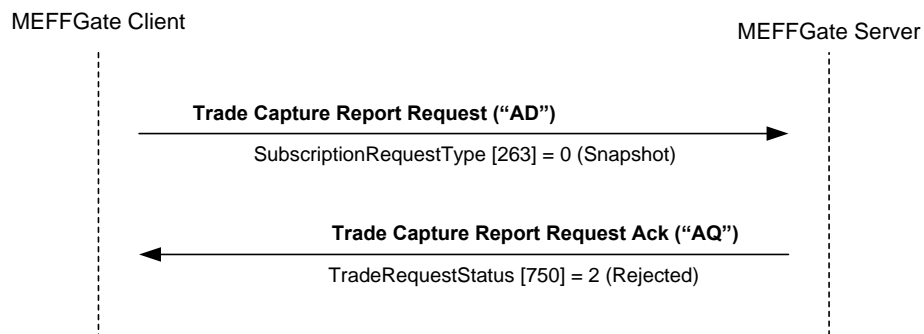
8.5 Annotations and adaptations of FIX 4.4

- The Trade Capture Report Request message only allows some of the selection criteria covered by the standard

8.6 Definition of messages

8.6.1 Trade Capture Report Request (Msg Type = AD)

Message sent by the client to request information on CCP trades; allows requests with or without new trade updates.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = AD		
568	TradeRequestID	Y		String (10)	Identifier of the request assigned by the user. It must be a unique value when it is a new request, and the value previously assigned when it is for an unsubscription
569	TradeRequestType	Y	0=All trades that match criteria	Int	
263	SubscriptionRequestType	N	0=Snapshot (default value) 1=Snapshot + Updates 2= Unsubscribe	Char	Indicates if it is a request with or without updates or the cancellation of a previous request
818	SecondaryTradeReportID	N		String	CCP register number
828	TrdType	N	15 = Deliveries	Int	Only included in the specific request of the purchase/sale trades to carry out in IBERCLEAR as a result of the delivery procedure
	Start <Parties>				
453	NoPartyIDs	N		NumInGroup	Number of selection criteria defined in the Parties block
→ 448	PartyID	N		String	MEFF code of the member or user. Required if NoPartyIDs has been specified
→ 447	PartyIDSource	N	D = Proprietary/ Custom code	Char	Required if NoPartyIDs has been specified
→ 452	PartyRole	N	13 = Order Origination Firm 4 = Clearing Firm	Int	Indicates the role that taken by the code specified in the PartyID. Required if NoPartyIDs has been specified
	End <Parties>				
580	NoDates	N	1	NumInGroup	
→75	TradeDate	N	"19800101"	LocalMktDate	Only included in the request of all trades with live volume not 0. In this case must contain the special value "19800101"
54	Side	N	1 = Buy 2 = Sell	Char	Optional selection criteria. Allows only those trades made on the buyer side or the seller side to be obtained
	Standard Trailer	Y			

8.6.2 Trade Capture Report Request Ack (Msg Type = AQ)

Message sent by MEFFGate to notify that a Trade Capture Report Request message is invalid or no trade meets the selection criteria.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = AQ		
568	TradeRequestID	Y		String	Identifier specified in the corresponding Trade Capture Report Request message
569	TradeRequestType	Y		Int	Same value as specified in the request
263	SubscriptionRequestType	N	0=Snapshot (default value) 1=Snapshot + Updates 2= Unsubscribe	Char	Same value as specified in the request
748	TotNumTradeReports	N	0	Int	Number of transactions that meet the selection criteria. The value is always 0 as this message is only sent when the request is invalid or there are no transactions which meet the selection criteria
749	TradeRequestResult	Y	0 = Successful 2 = Invalid TradeRequestType 99=other	Int	Result of request When the value is 99, there is an explanation in the Text field
750	TradeRequestStatus	Y	1 = Completed 2 = Rejected	Int	Request status. When an error is notified the value is 2. When there are no trades that meet the selection criteria the value is 1
58	Text	N		String	An explanation of the rejection reason is given
	Standard Trailer	Y			

9.   Trades Management

9.1   Introduction

This chapter covers the functionalities associated with the full or partial transfer of a trade from one account to another and the average trade price allocation.

MEFF classifies this procedure according to the source account type and the destination account type.

MEFF term	Source	Destination
Allocation of daily account	Daily account	Standard account (same member)
Transfer	Standard account	Standard account (same member)
Average trade price allocation	Standard account	Same source account
Give-up	Daily or standard account	Another member

MEFF only allows the transfer of orders that have been executed (i.e. trades).

In the following sections the different procedures are described.

9.2   Allocation of Daily Account and Transfer

From FIX interface point of view, an allocation of Daily account and transfer work in the same way. The only difference is the type of origin account and this attribute is implicit in the account and therefore it is not necessary to specify it in the message.

The request is made through the Allocation Instruction message, indicating the trade to transfer data and the destination account.

MEFFGate notifies of the status of the request, to the user who made the request, via the Allocation Instruction Ack messages.

Once the request has been accepted, MEFFGate informs all the users via a Confirmation message including the allocation data.

9.3   Average trade price allocation

Ability to group trades carried out in the same session, one member, in a single contract, in the same account, on the same side to be allocated to the weighted average price of this group.

The request is made through the Allocation Instruction message, indicating the trades to convert into a single trade at the average price.

MEFFGate notifies of the status of the request, to the user who made the request, via the Allocation Instruction Ack messages.

Once the request has been accepted, MEFFGate informs all the users via Trade Capture Report messages.

9.4   Give-up (Executing Broker)

The Give-up operation is always started by the Member holder of the account associated with the trade being transferred. In this document, this figure is referred as Executing Broker.

The Executing Broker can request the performing of a Give-up through the Allocation Instruction message. This message should contain the trade to transfer data and the Clearing Broker, and a reference for the Clearing Broker.

Before the Give-up is accepted by the Clearing Broker, any user from the Executing Broker can cancel the request with a new Allocation Instruction message.

MEFFGate notifies the reception of Allocation Instruction message through an Allocation Instruction Ack message only to the user who started the request.

In addition, MEFFGate informs all the Executing Broker users of the Give-up data and the different status that happen by Confirmation messages.

9.5   Give-up (Clearing Broker)

The Clearing Broker is chosen by the Executing Broker in the request message as a receiver of the trade being transfer.

Once the Give-up request is processed by the central systems, it can be automatically accepted or to be left of acceptance by the Clearing Broker.

The fact that a Give-up is automatic depends on the CCP rules and the possible filters configured by the Clearing Broker.

In case of the Give-up is left without acceptance, MEFFGate sends a Confirmation message to all users of the Clearing Broker informing of the Give-up data and requesting the acceptance or rejection of it. If the Clearing Broker had a destination account defined for the reference introduced in the request message, this account will be present in the message.

The acceptation or rejection of a Give-up is made by a Confirmation Ack message. In case of acceptance the message should indicate the destination account of the Give-up, irrespective of whether or not the information regarding the account associated with the reference was received. ,

If the Clearing Broker clears the selected account; its acceptance is enough for the performing of the Give-up. If the Clearing Broker does not clear the selected account, acceptance will be necessary by the Clearing Member of the destination account to perform the Give-up.

If acceptance by the Clearing Member of the destination account is necessary, once the Give-up has been accepted or rejected by the Clearing Member, MEFFGate sends a Confirmation message informing the status of the Give-up.

In the case of a Clearing Member refusing the Give-up, it will be returned to a pending state with the Clearing Broker, who may opt to refuse it definitively or specify an account another time. Both actions are carried out via the Confirmation Ack message as described above.

9.6   Give-up (Clearing Member)

When the Clearing Broker of a Give-up is not a clearer of its operation, acceptance will be necessary by the Clearing Member of the destination account to perform the Give-up

As it happens in the Clearing Broker case, this acceptation can be made automatically by MEFF central systems from the possible filters defined by the Clearing Member.

When a Give-up is left pending acceptance by the Clearing Member of the destination account, MEFFGate sends a Confirmation message to all users of that member informing of the Give-up data and requesting the acceptance or rejection of it. This acceptance or rejection is made by a Confirmation Ack message.

In the case of a Clearing Member refusing the Give-up, it will remain awaiting update on the part of the Clearing Broker. The Clearing Member will only receive a further message relating to this Give-up when the Clearing Broker opts to further select accounts that are also cleared by the same Clearing Member.

9.7 AllocID

The field AllocID, present in a request initiated by an Allocation Instruction message, is the identifier that allows the cancellation of that request (only in Give-up), as it relates to the request with Confirmation messages of the status of allocations and Give-ups.

The field AllocID assigned by the client should be ten characters length. If length is inferior, MEFFGate complete with spaces to achieve that length. MEFFGate also expects that messages sent by the client use an AllocID of 30 length, in this case only the last ten positions can be fixed free, since the 20 first should coincide with the format explained below.
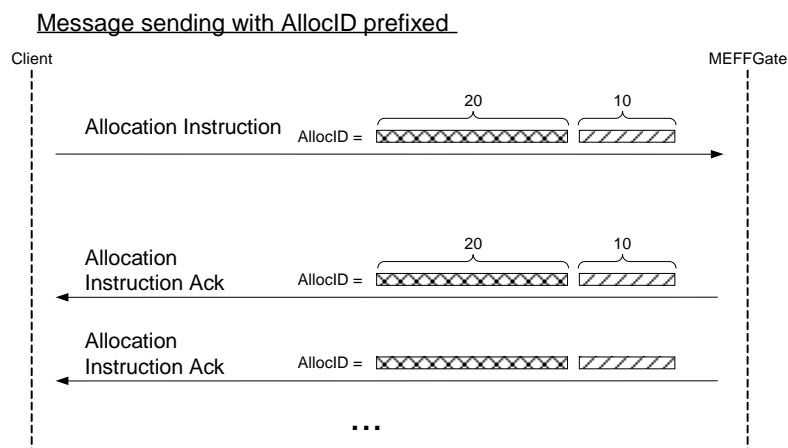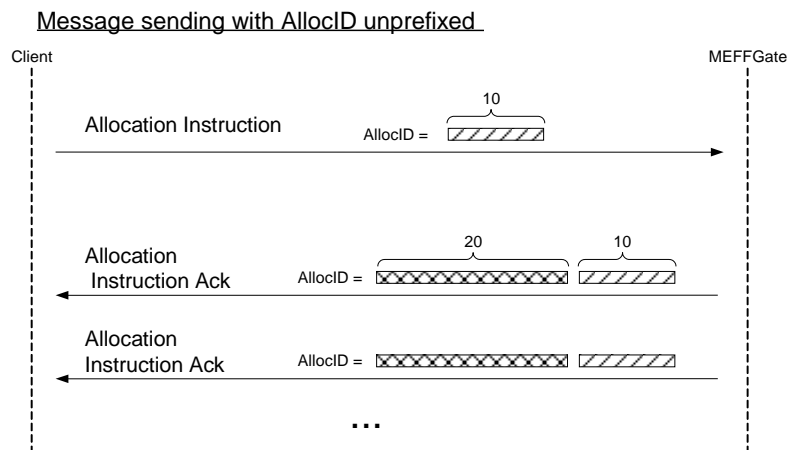
- Assigning of the prefix to the AllocID identifier -

A process in the MEFFGate of assigning a prefix to the AllocID field is performed to avoid duplicates in this identifier.

The AllocID assigned by MEFFGate in the reply message has the format AAMMDDMmmmTttMmmmTttNnnnnnnnnn, made by the following codes:

- **AAMMDD**. It is the date of the CCP session

- **MmmmTtt**. Contains the member and user code of connection from which the request was made

- **Nnnnnnnnnn**. It is the value assigned by the client application to AllocID in the original message

In the case of Give-up, this identifier is also present in the Confirmation messages received by the Executing Broker, which are generated as a result of a Give-up request.

This way, a user of the Executing Broker who wants to cancel a Give-up, must use this identifier in the field RefAllocID of the Allocation Instruction request message.

9.8 SecondaryAllocID Field

The SecondaryAllocID field is the unique identifier of each allocation performed at MEFF, be it a Give-up, assignment of the daily account or an actual allocation.

This field is present in the Confirmation messages which the client application receives informing of the status of the allocation.

In this way all messages relating to the same Give-up will contain the same value in this field, irrespective of the role of the receiver. This value allows the identification of the Give-up in an univocal manner by the CCP and the other participants.

9.9 Monitoring of trades by Trade Capture Report messages

In MEFF, the operation to transfer totally or partially a trade from one account to another is made by the making two new trades; the first one on the origin account with the opposite sign to the original trade and the second one on the destination account with the same sign as the original trade. The volume of both trades is the number of contracts that are transferred from the original trade.

Once made the transfer process, being an allocation on daily account, a transfer, a Give-up or an average trade price allocation, those users who are subscribed will receive the Trade Capture Report messages of corresponding the trade or trades.

In the case of the trade coming transfer, the Trade Capture Report message contains a series of relevant fields for the reconciliation of the information and monitoring of the trades:

- **SecondaryAllocID**. Contains the identifier assigned by the system to the corresponding allocation operation. This same field will be found in the related Confirmation messages, as explained in the above section.

- **PositionEffect**. Indicates if the trade opens or close the position. From the trades coming from a transfer, the one made on the origin account will contain "C" (Close), and the one made on the destination account "O" (Open)

- **SecondaryTradeReportRefID**. Contains the CCP register number of the origin trade.

- **SecondaryTradeReportID**. Contains the CCP register number of the new trade. This same field will be found in the related Confirmation messages that informs about the acceptance of the transfer.

- **TrdMatchID**. Contains the CCP trade number of the initial trade.

9.10 List of messages

Message	Description
Allocation Instruction (Msg Type = J)	Request for a transfer, an allocation from daily account, a Give-up or an average trade price allocation
Allocation Instruction Ack (Msg Type = P)	Acknowledgement of Allocation Instruction message. It contains the request status
Confirmation (Msg Type = AK)	Report of the state of transfer, allocation from daily account or a Give-up. Also used for requesting the acceptance or rejection of a Give-up
Confirmation Ack (Msg Type = AU)	Message of acceptation or rejection of a Give-up
Trade Capture Report (Msg Type = AE)	Report on execution of transfer. Sent to clients involved that are subscribed

9.11 Message flow

In the following diagrams the use of two arrows indicates that the message goes to all the users of the member

Allocation Request for a transfer or an allocation from daily account accepted

The client sends the request through an Allocation Instruction message. It first receives the Allocation Instruction Ack message with the field AllocStatus = 3, indicating that MEFFGate has received the message and sent it to the central systems. Subsequently an Allocation Instruction Ack message is received with the field AllocStatus = 0, indicating that the message has been accepted by the central system.

When the allocation has been made, a confirmation message is sent to all the client applications of the Member informing of the data associated with the request.

In addition, a Trade Capture Report is sent to the client applications subscribed to this type of message, for each trade derived from the allocation made.

FIX client   FIX server

Allocation Instruction ("J")

AllocID [70] = *X*

Allocation Instruction Ack("P")

AllocStatus [87] = 3 (Received)
AllocID = *pref+X*

Allocation Instruction Ack("P")

AllocStatus [87] = 0 (Accepted)
AllocID = *pref+X*

Confirmation("AK")

ConfirmTransType [666] = 0 (New)
ConfirmType [773] = 1 (Status)
SecondaryAllocID[793] = *idt*
SecondaryConfirmStatus [5683] = "A"
AllocID [70] = *pref+X*
SecondaryTradeReportID[818] = *NReg*

Trade Capture Report ("AE")

SecondaryTradeReportID[818] = *NReg*

Average Trade Price Allocation Request accepted

The client sends the request through an Allocation Instruction message. It first receives the Allocation Instruction Ack message with the field AllocStatus = 3, indicating that MEFFGate has received the message and sent it to the central systems. Subsequently an Allocation Instruction Ack message is received with the field AllocStatus = 0, indicating that the message has been accepted by the central system.

When the Average Trade Price allocation has been made Trade Capture Report messages are sent to the client applications subscribed to this type of message.

FIX client   FIX server

Allocation Instruction ("J")

AllocID [70] = *X*

Allocation Instruction Ack("P")

AllocStatus [87] = 3 (Received)
AllocID = *pref+X*

Allocation Instruction Ack("P")

AllocStatus [87] = 0 (Accepted)
AllocID = *pref+X*

Trade Capture Report ("AE")

SecondaryTradeReportID[818] = *NReg*
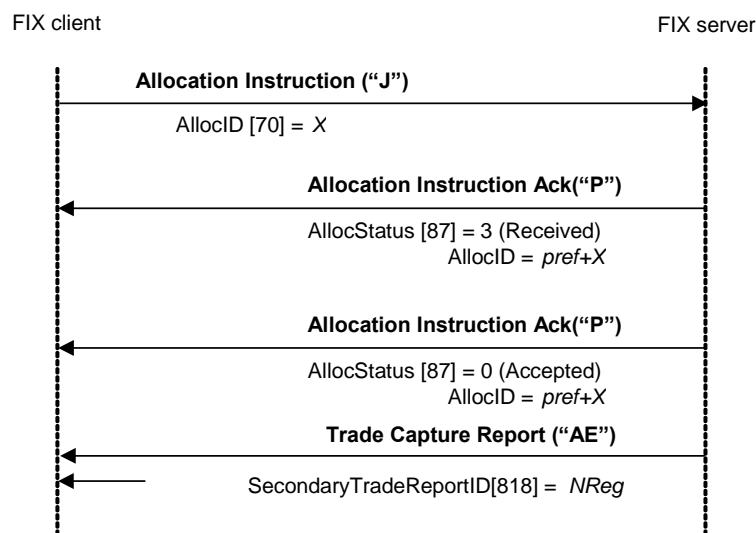
Allocation request rejected by central systems

The client sends the request through an Allocation Instruction message. It first receives the Allocation Instruction Ack message with the field AllocStatus = 3, indicating that MEFFGate has received the message and it has sent it to the central systems. Finally it receives an Allocation Instruction Ack message with the field AllocStatus = 1, indicating that the allocation has been rejected by the central systems.

FIX client   FIX server

Allocation Instruction ("J")

AllocID [70] = *X*

Allocation Instruction Ack("P")

AllocStatus [87] = 3 (Received)
AllocID = *pref+X*

Allocation Instruction Ack("P")

AllocStatus [87] = 1 (Rejected)
AllocID = *pref+X*

Allocation request rejected by the MEFFGate server

The client sends the request through an Allocation Instruction message. The message is rejected by MEFFGate which responds with an Allocation Instruction Ack message with the field AllocStatus = 1.

FIX client   FIX server

Allocation Instruction ("J")

AllocID [70] = *X*

Allocation Instruction Ack("P")

AllocStatus [87] = 1 (Rejected)
AllocID = *pref+X*

Request of an accepted Give-up which Clearing Broker is not the clearer of its own operative

Executing Broker

FIX Server

Clearing Broker

Allocation Instruction ("J")

AllocID [70] = X

Allocation Instruction Ack("P")

Clearing Member

AllocStatus [87] =3 (Received)
AllocID [70] = pref+X

Allocation Instruction Ack("P")

AllocStatus [87] =0 (Accepted)
AllocID = pref+X

Confirmation("AK")

ConfirmTransType [666] = 0 (New)
ConfirmID [664] = n
ConfirmType [773] = 1 (Status)
SecondaryAllocID[793] = idg
SecondaryConfirmStatus [5683] = "P"
AllocID [70] = pref+X

Confirmation("AK")

ConfirmTransType [666] = 0 (New)
ConfirmID [664] = m
ConfirmType [773] = 2 (Confirmation)
SecondaryAllocID[793] = idg
SecondaryConfirmStatus [5683] = "P"

Confirmation ack ("AU")

ConfirmID [664] = m
AffirmStatus [940] = 3 (Affirmed)
AllocAcount [79] = Destination account
SecondaryAllocID[793] = idg

Confirmation("AK")

ConfirmTransType [666] = 1 (Replace)
ConfirmID [664] = n'
ConfirmRefID [772] = n
ConfirmType [773] = 1 (Status)
SecondaryAllocID[793] = idg
SecondaryConfirmStatus [5683] = "N"
AllocID [70] = pref+X

Confirmation("AK")

ConfirmTransType [666] = 1 (Replace)
ConfirmID [664] = m'
ConfirmRefID [772] = m
ConfirmType [773] = 1 (Status)
SecondaryAllocID[793] = idg
SecondaryConfirmStatus [5683] = "N"

Confirmation("AK")

ConfirmTransType [666] = 0 (New)
ConfirmID [664] = p
ConfirmType [773] = 2 (Confirmation)
SecondaryAllocID[793] = idg
SecondaryConfirmStatus [5683] = "N"

(continues on the next page ...)

(… continues from previous page)

Clearing Member ⋮
Clearing Broker ⋮
Executing Broker   Servidor FIX

Confirmation ack ("AU")

ConfirmID [664] = *p*
AffirmStatus [940] = 3 (Affirmed)
SecondaryAllocID[793] = *idg*

Confirmation("AK")

ConfirmTransType [666] = 1 (Replace)
ConfirmID [664] = *n''*
ConfirmRefID [772] = *n'*
ConfirmType [773] = 1 (Status)
SecondaryAllocID[793] = *idg*
SecondaryConfirmStatus [5683] = "A"
AllocID [70] = *pref+X*
SecondaryTradeReportID [818] = *NReg*

Confirmation("AK")

ConfirmTransType [666] = 1 (Replace)
ConfirmID [664] = *m''*
ConfirmRefID [772] = *m'*
ConfirmType [773] = 1 (Status)
SecondaryAllocID[793] = *idg*
SecondaryConfirmStatus [5683] = "A"
SecondaryTradeReportID [818] = *NReg*

Confirmation("AK")

ConfirmTransType [666] = 1 (Replace)
ConfirmID [664] = *p'*
ConfirmRefID [772] = *p*
ConfirmType [773] = 1 (Status)
SecondaryAllocID[793] = *idg*
SecondaryConfirmStatus [5683] = "A"
SecondaryTradeReportID [818] = *NReg*

Trade Capture Report ("AE")

SecondaryTradeReportID [818] = *NReg*

Trade Capture Report ("AE")

SecondaryTradeReportID [818] = *NReg*

Trade Capture Report ("AE")

SecondaryTradeReportID [818] = *NReg*

Give-out cancelled by the Executing Broker

Give-up registration or cancellation request rejected

When a request to register or cancel a Give-up is not valid, it will be rejected via the corresponding Allocation instruction Ack message. In this case, an associated Confirmation message does not exist since the request is not processed.

Miembro Origen   Servidor FIX

Allocation Instruction ("J")

AllocID [70] = *X*

Allocation Instruction Ack("P")

AllocStatus [87] =1 (Rejected)
AllocID [70] = *pref+X*
Text [58] = *rejection reason*
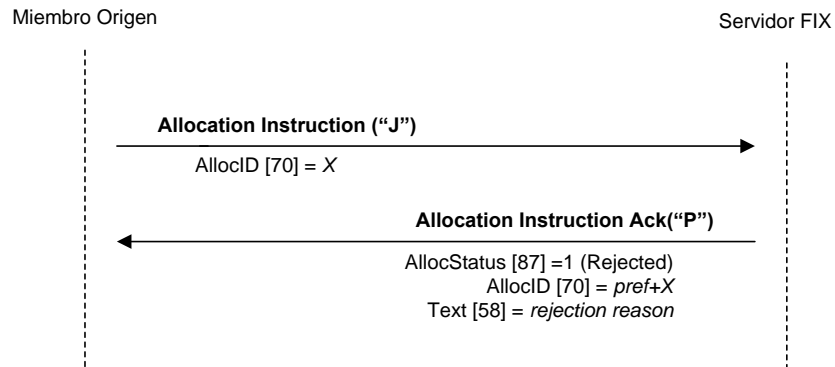
Send of message Confirmation Ack rejected

When a Confirmation Ack message is not valid, either due to the data or the status of the Give-up reference, it will be rejected by a Business Message Reject message.
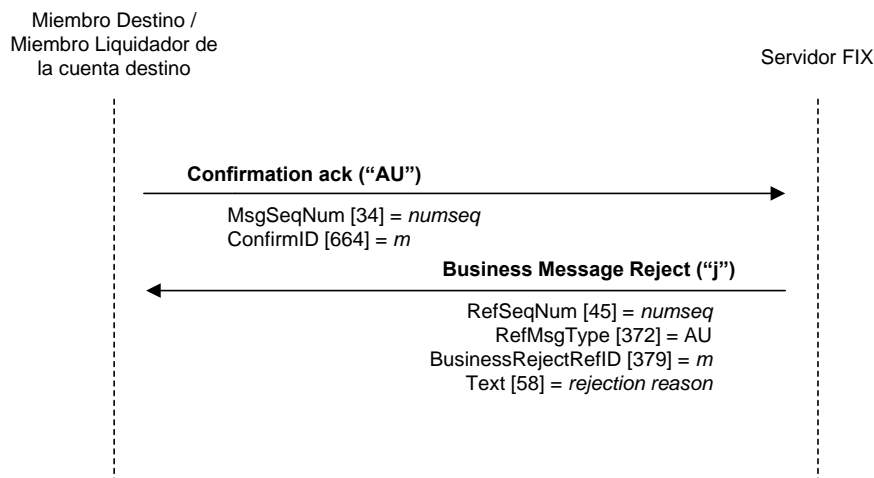
9.12 Annotations and adaptations of FIX 4.4

- Only some of the functionalities defined by FIX related to the Allocation Instruction message are supported. Transfers are only allowed post-trade, based on trades and not orders. Accordingly, certain fields related with orders that are required by the standard are ignored by MEFFGate

- FIX allows the assignment of various trades in a single message and allows allocation to various accounts. These functionalities are not supported by MEFFGate. An Allocation Instruction message is related with a single trade and destination account.

- The NoExecs (124), LastQty (32), AllocAccount (79) and AllocQty (80) fields are now required in the Allocation Instruction message

- The SecondaryTradeReportID (818) field has been added to the Allocation Instruction message.

- The SecondaryTradeReportID (818), NoExecs (124) and SecondaryTradeReportRefID (881) fields have been added to the Confirmation message

- The user field SecondaryConfirmStatus (5683) has been added to the Confirmation message

- The AllocAcount (79) and SecondaryAllocID (793) fields have been added to the Confirmation Ack message

9.13 Definition of messages

9.13.1 Allocation Instruction (Msg Type = J)

Message sent by the client to request a transfer, an allocation from the daily account, a Give-up request or an average trade price allocation.

The criteria used by MEFF to differentiate a Give-up of a transfer or an allocation is the presence of the 538 field in the message.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = J		
70	AllocID	Y		String(30)	Unique identifier for each Allocation Instruction message
71	AllocTransType	Y	0 = New 2 = Cancel	Char	It indicates if the message is a transfer or a cancellation request The cancellation can only be made on a transfer of Give-up that is pending of acceptance on the Clearing Broker
626	AllocType	Y	1 = Buyside Calculated	Int	
72	RefAllocID	N		String(30)	Identifier of the Give-up to cancel. Required if AllocTransType contains 2
796	AllocCancReplaceReason	N	1 = Original details incomplete/incorrect 99 = Other	Int	Required if AllocTransType contains 2. Ignored by MEFFGate.
857	AllocNoOrdersType	Y	0 = Not specified	Int	
124	NoExecs	Y*		NumInGroup	Indicates the number of trades that are going to be allocated. It must be 1 except average trade price allocation.
→32	LastQty	Y*	> 0, no decimals	Qty	Trade volume
→31	LastPx	N		Price	Trade price. Required except average trade price allocation.
→818*	SecondaryTradeReportID	Y*		String	CCP register number of the trade that is going to be transferred
54	Side	Y	1 = Buy 2 = Sell	Char	Indicates if the trade to be transferred is buyer side or seller side
	Start <Instrument>				
55	Symbol	Y	Contract code	String(22)	Contract code
	End <Instrument>				
53	Quantity	Y	> 0	Qty	Amount of contracts to be transferred
6	AvgPx	Y		Price	Resulting trade price. Must contain the same value as the LastPx field. Ignored by MEFFGate when it is an average trade price allocation.
75	TradeDate	Y		LocalMktDate	Ignored by MEFFGate
78	NoAllocs	Y*	1	NumInGroup	Number of destinations. MEFFGate only accepts a single destination

Tag	Name	Req	Valid values	Format	Description
→ 79	AllocAccount	Y*	Exact length	String(5)	Destination account. Ignored by MEFFGate when it is a Give-up request
→ 80	AllocQty	Y*		Qty	Amount to be transferred. Must be equal to the sum of all LastQty [32] sent in this message Allocation Instruction.
→	Start <NestedParties>				
→ 539	NoNestedPartyIDs	N		NumInGroup	Required if it is a Give-up request
→→ 524	NestedPartyID	N		String	Give-out internal reference, Clearing Broker member code, Give-up reference or Give-out mnemonic. For the Give-out internal reference (NestedPartyRole=3) this tag has a maximum lengh of 18 characters. For the Give-up Clearing Firm (NestedPartyRole=14) the length for this tag is 4 alphanumeric characters. For the Give-up reference (NestedPartyRole=24) this tag has a maximum lengh of 18 characters. For the Give-up mnemonic (NestedPartyRole=33) this tag has a maximum lengh of 10 characters. Required if NoNestedPartyIDs is specified
→→ 525	NestedPartyIDSource	N	D = Proprietary/ Custom code	Char	Required if NoNestedPartyIDs is specified
→→ 538	NestedPartyRole	N	3 = Give-out internal reference 14 = Give-up Clearing Firm 24 = Give-up Reference 33 = Give-up mnemonic	Int	Required if NoNestedPartyIDs is specified. Value 3 indicates the value specified in NestedPartyID is the reference assigned by the Executing Broker for internal purposes. It is associated to a Give-out mnemonic and it can be not unique. Need not be provided. Value 14 indicates the value specified in NestedPartyID is the Give-up Clearing Broker. Value 24 indicates the value specified in NestedPartyID is the Give-up reference. Value 33 indicates the value specified in NestedPartyID is the Give-out mnemonic.
→	End <NestedParties>				
	Standard Trailer	Y			

9.13.2 Allocation Instruction Ack (Msg Type = P)

Message used by MEFFGate to indicate the status of the request initiated with the Allocation Instruction message.

This message is only sent to the user who made the request.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = P		
70	AllocID	Y		String(30)	Identifier assigned by the client in the Allocation Instruction message
75	TradeDate	N		LocalMkt Date	
87	AllocStatus	Y	0 = Accepted 1 = Rejected 3 = Received not yet processed	Int	Request status The value 0 indicates the message has been accepted by the central system. The value 1 indicates that the transfer or allocation has been rejected. The value 3 indicates that the message has been received by the MEFFGate server
88	AllocRejCode	N	0 = Unknown account 1 = Incorrect quantity 7 = Other 8 = Incorrect allocated quantity	Int	Identifies the reason for rejection. Only present when AllocStatus = 1 If it contains the value 7, the Text field has an explanation of the reason for the rejection
58	Text	N		String	Explanation of reason for rejection
	Standard Trailer	Y			

9.13.3 Confirmation (Msg Type = AK)

Message used by MEFFGate to notify of the status of a daily account allocation, transfer or Give-up. Also used to request the acceptance or rejection of a Give-up.

In the description of fields of this message, the term "transfer" will be used to refer to a daily account allocation, the Give-up or the transfer itself.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = AK		
664	ConfirmID	Y		String	Unique identifier for each Confirmation message
772	ConfirmRefID	N		String	Identifier of Confirmation message being replace by this message. Included when ConfirmTransType = 1
666	ConfirmTransType	Y	0 = New 1 = Replace	Int	
773	ConfirmType	Y	1 = Status 2 = Confirmation	Int	Indicates if the message is a Give-up acceptance request (2=Confirmation) or a Give-up status notification (1=Status)
665	ConfirmStatus	Y	1 = Received	Int	The content of this field should not be considered. It is included to meet requirements of the standard
	Start <Parties>				
453	NoPartyIDs	N		NumInGroup	
→ 448	PartyID	N		String	Member, User code or Give-up reference
→ 447	PartyIDSource	N	D = Proprietary / custom code	String	Required if NoPartyIDs is specified
→ 452	PartyRole	N	1 = Executing Firm 3 = Give-out internal reference 4 = Clearing Firm 12 = Executing Trader 14 = Give-up Clearing Firm 24 = Give-up Reference 33 = Give-up mnemonic 36 = Clearing Broker Trader	Int	Required if NoPartyIDs is specified Value 1 is present in messages to the Executing and Clearing Broker of a Give-up informing who the Executing Broker is. Value 3, when present, informs the internal reference to the Executing Broker of a Give-up. Value 4 is present in messages to the Clearing Broker when the message contains an account in field 79, informing who the Clearing Member of the account is. This value is present too in the messages to the Clearing Member. Value 12 is present in messages to the users of Member who made the transfer request, as well as to the users of the Clearing Broker, informing which trader started the request. Value 14 is present in all messages related with Give-ups, informing who the Clearing Broker is. Value 24, when present, informs the Give-up reference Value 33, when present, informs the Give-out mnemonic to the Executing Broker and the Give-in mnemonic to the Clearing Broker. Value 36, when present, informs who the user of the Clearing Broker that accepted or rejected the Give-up is
	End <Parties>				

Tag	Name	Req	Valid values	Format	Description
70	AllocID	N		String(30)	Allocation Instruction message identifier related. Only present when the message goes to the Member of the trader who started the allocation
793	SecondaryAllocID	N		String(10)	Transfer unique identifier
60	TransactTime	Y		UTCTime stamp	Time in which the transaction is made, in UTC format
75	TradeDate	Y		LocalMkt Date	Date in which the transaction is made
	Start <Instrument>				
55	Symbol	Y		String(22)	Contract code
	End <Instrument>				
711	NoUnderlyings	Y	0	NumInGr oup	The content of this field should not be considered. It is included to meet requirements of the standard
555	NoLegs	Y	0	NumInGr oup	The content of this field should not be considered. It is included to meet requirements of the standard
80	AllocQty	Y	> 0, no decimal places	Qty	Give-up volume
54	Side	Y	1 = Buy 2 = Sell	Char	Indicates the sign of the trade to be transfer
862	NoCapacities	Y	0	Int	The content of this field should not be considered. It is included to meet requirements of the standard
79	AllocAccount	Y		String(5)	Destination account. When it is a Give-up this field is only informed when the message goes to a Clearing Broker or Clearing Member. It contains "[N/A]" otherwise.
6	AvgPx	Y		Price	Trade price
381	GrossTradeAmt	Y		Amt	Nominal/Effective amount traded
118	NetMoney	Y	0	Price	The content of this field should not be considered. It is included to meet requirements of the standard
818*	SecondaryTradeRep ortID	N		String	CCP register number of the new trade. Only present when the field SecondaryConfirmStatus = "A".
881*	SecondaryTradeRep ortRefID	N		String	CCP register number of trade to be transferred. Only included if the message goes to the Executing Broker
5683 *	SecondaryConfirmSt atus	N	"P" = Pending "N" = Accepted by the Clearing Broker ""R" = Rejected by the Clearing Broker "A" = Accepted by the Clearing Member "L" = Rejected by the Clearing Member "C" = Cancelled by the Executing Broker "S" = Cancelled by the system "H" = Not valid request	Char	Describes the Give-up state

Tag	Name	Req	Valid values	Format	Description
			"X" = Rejected request		
124*	NoExecs	N		NumInGroup	Indicates the number of trades that has been allocated
→881*	SecondaryTradeReportRefID	N		String	CCP register number of trade to be transferred. Only included if the message goes to the Executing Broker
	Standard Trailer	Y			

9.13.4 Confirmation Ack (Msg Type = AU)

Message used by the client to accept or reject a Give-up.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = AU		
664	ConfirmID	Y		String	Identifier of Confirmation request message
75	TradeDate	Y		LocalMkt Date	MEFFGate does not process this field and it is included to meet requirements of the standard
60	TransactTime	Y		UTCTime Stamp	MEFFGate does not process this field and it is included to meet requirements of the standard
940	AffirmStatus	Y	2 = Confirm rejected, i.e. not affirmed 3 = Affirmed	Int	
79*	AllocAccount	N	Exact length	String(5)	Give-in destination account. Required if it is an acceptance message of the Give-up by the Clearing Broker
793*	SecondaryAllocID	Y		String(10)	Give-up unique identifier which is being referred.
	Standard Trailer	Y			

9.13.5 Business Message Reject (MsgType = j)

This message is sent by MEFFGate to reject a Confirmation Ack invalid message. The description of this message can be found in section 5.8.5.

10. Exercise Instructions

10.1 Introduction

The functionality of an exercise instruction allows a FIX client to request the exercise at the end of the day for a combination of options associated to a specific account and contract.

The client must specify the contract code of the option to exercise and the account in which the position is held. The contract code must refer to an option contract that accepts the exercise instruction; otherwise the instruction will be rejected.

An exercise instruction implicitly indicates that the other options are not to be exercised.

MEFFGate does not implement a method to indicate the number of options not to be exercised (where it is understood that the other options will be exercised). When a client does not want to exercise any option in a position (only allowed on the expiration day), he has to send an exercise instruction with volume 0.

The early exercise instruction (before the expiration date) can be rejected, whereby it will return to its original status, and consequently no option would be exercised.

Exercise instructions sent on the expiration date can also be rejected; whereby the default procedure will take place (only in-the-money options will be exercised based on the closing price of the underlying in Market Data message).

When more than one exercise instruction is sent for the same account and contract, the last instruction cancels any previous instruction, with the volume to exercise being the amount indicated in this last instruction.

There are two scenarios for exercise instructions depending on the expiration date, which are described in the following sections.

In both scenarios, options are exercised at the end of the trading session. During the day, a FIX client can consult through MEFFGate the pending exercise instructions for the end of the day. This consultation can be made using various selection criteria as detailed in the corresponding message description (Request for Positions).

The trades that are generated when options are exercised can be queried using the Trade Capture Report Request message, as described in chapter 7 Monitoring and Management of Positions.

10.2 Scenario 1: American options, before expiration date

An exercise instruction, in this scenario, becomes effective at the end of the session. During the session, the exercise instruction can be replaced by sending a new exercise instruction or cancelling it with the corresponding message.

If at the end of the session the account does not have the number of contracts indicated in the exercise instruction, only the available contracts will be exercised, with the rest of the instruction being cancelled.

10.3 Scenario 2: Expiration date

This scenario includes both American options and European option. On the expiration day there are three alternatives:

      a. **Send an exercise instruction as explained in scenario 1**. This instruction can be replaced or revoked. At the end of the session the total volume available will be exercised until the specified amount is met. The remaining position will not be

exercised and it will expire. If the instruction is revoked the default procedure will be followed as explained in point c

b. **Send an exercise instruction to exercise 0 contracts**. At the end of the session, no options will be exercised and all contracts will expire. This petition can be replaced by another or revoked, with a or c applying as appropriate

c. **Do nothing and opt for default procedure**. The default procedure is to exercise the position if it is in-the-money and not to exercise it if it is at-the-money or out-of-the-money. Note that the default procedure can only be applied to the full position once. This is the only alternative for contracts that do not accept exercise instructions

10.4 List of messages

Message	Description
Position Maintenance Request (Msg Type = AL)	Sent by the FIX client to request the exercise of a contract
Position Maintenance Report (Msg Type = AM)	Sent by MEFFGate to indicate the acceptance or rejection of an exercise instruction
Request For Positions (Msg Type = AN)	Used by the FIX client to request the exercise petitions that are pending on its accounts
Request For Positions Ack (Msg Type = AO)	Sent by MEFFGate as reply to Request For Positions message. It precedes the Position Report messages (if there are any)
Position Report (Msg Type = AP)	Sent by MEFFGate as reply to an accepted Request For Positions message. Notifies the active exercise instructions (but not realised) for a specific account and contract
Trade Capture Report (Msg Type = AE)	Sent by MEFFGate to notify that an option has been exercised. Only sent to related clients subscribed to this type of message

10.5 Message flow

Early exercise instruction

The client sends an instruction to exercise an American option before its expiration date. MEFFGate replies with a message accepting the instruction. The next exercise instruction sent on the same account and contract substitutes the previous order. At the end of the day, the volume indicated in the last instruction, or the volume available, if this is less, is exercised and the corresponding Trade Capture Report is received notifying this, if this type of message has been subscribed to. If no position is present in the account, no option is exercised and therefore no associated Trade Capture Report is generated.

MEFFGate Client   MEFFGate Server

Position Maintenance Request ("AL")

PosMaintAction [712] = 1 (New)
Account [1] = *X*
Symbol [55] = *Y*

Position Maintenance Report ("AM")

Position Maintenance Request ("AL")

PosMaintAction [712] = 1 (New)
Account [1] = *X*
Symbol [55] = *Y*

Position Maintenance Report ("AM")

.
.
.

Trade Capture Report ("AE")
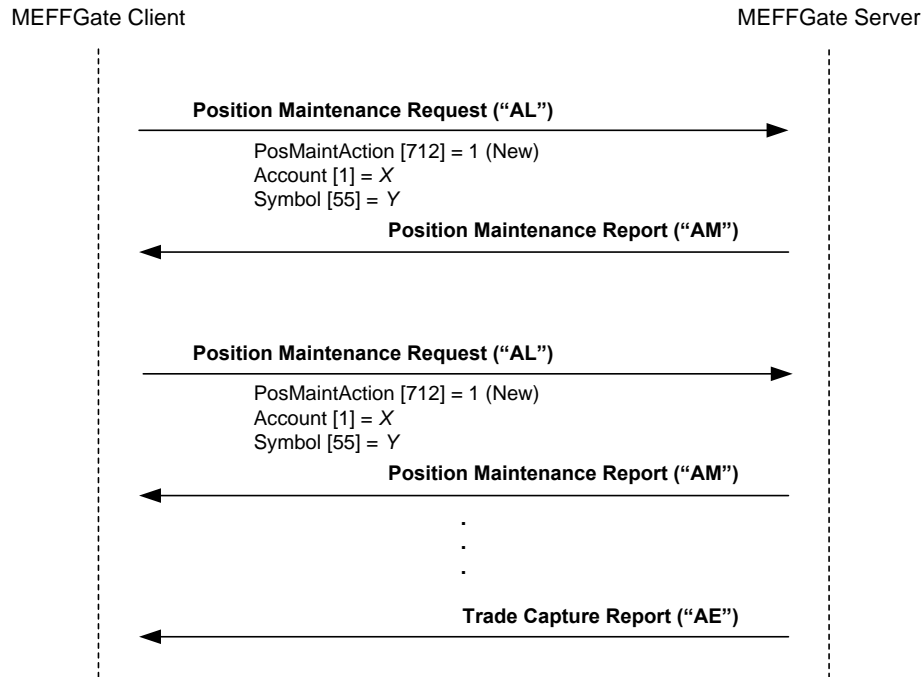
Exercise instruction and subsequent cancellation

Any exercise instruction can be cancelled with a Position Maintenance Request message with the field PosMaintAction = 3. Once it is cancelled the default procedure is followed, which means there is no early exercise and the option will be exercised on the expiration date if it is in-the-money.

In the example below, the options are considered to be in-the-money on the expiration day, so the options are exercised.

MEFFGate Client   MEFFGate Server

Position Maintenance Request ("AL")

PosMaintAction [712] = 1 (New)
Account [1] = *X*
Symbol [55] = *Y*

Position Maintenance Report ("AM")

Position Maintenance Request ("AL")

PosMaintAction [712] = 3 (Cancel)
Account [1] = *X*
Symbol [55] = *Y*

Position Maintenance Report ("AM")

.
.
.

Trade Capture Report ("AE")
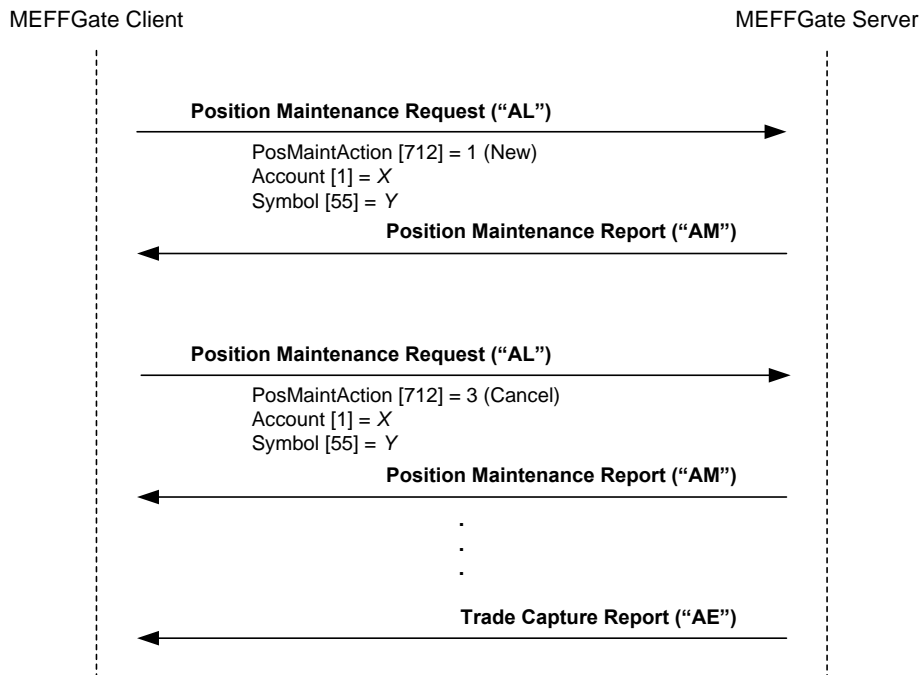
Exercise instruction rejected by the system

If the FIX client sends an invalid exercise instruction, on an European option prior to expiration, for example, it will be rejected with a Position Maintenance Report message with the field PosMaintStatus = 2. A rejected exercise instruction will not alter an exercise instruction sent previously.

MEFFGate Client   MEFFGate Server

Position Maintenance Request ("AL")

Position Maintenance Report ("AM")

PosMaintStatus [722] = 2 (Rejected)
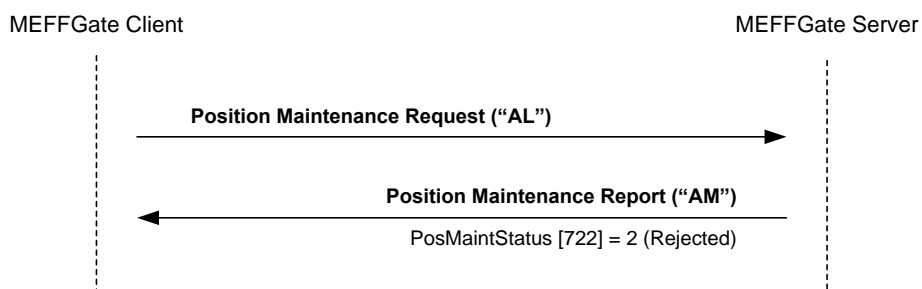
Request status of pending exercise instructions

The client can request the status of exercise instructions on a series of accounts and contracts. As a result he will obtain a Position Report message for each account-contract that has an exercise instruction.

This system implements the Snapshot and Update mechanism, and allows the client to receive updates as they occur, if required.

MEFFGate Client   MEFFGate Server

Request For Positions ("AN")

PosReqType [724] = 2 (Exercises)

Request For Positions Ack ("AO")

Positions Report ("AP")

...
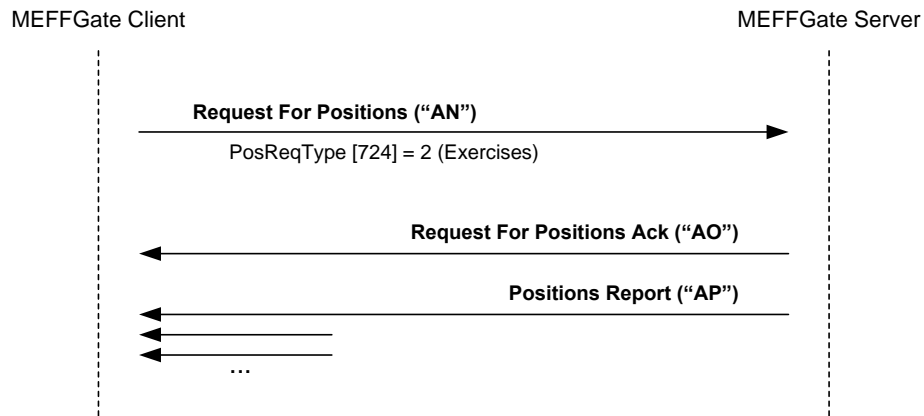
10.6 Annotations and adaptations of FIX 4.4

No annotations or adaptations have been made to the messages in this chapter

10.7 Definition of messages

The Position Maintenance Request and Position Maintenance Report messages mentioned in this chapter are also used for the position adjustment. The definition can be found in section 7.8.5 and 7.8.6 of chapter 7.

The Request For Positions, Request For Positions Ack, Position Report and Trade Capture Report messages mentioned in this chapter are also used for the position monitoring. The definition can be found in sections 7.8.1, 7.8.2, 7.8.3 and 7.8.4.

11. Communication of Events

11.1 Introduction

This chapter describes two functionalities based on the News message:

- Relay information from the CCP supervisor

- Send messages from a client application to the CCP supervisor

In both cases the information transferred has a free text format.

A client program does not need to subscribe to receive these messages. Every client is implicitly subscribed from the start of the session.

There is no way of checking that the message has reached its destination.

On establishing a communications connection, if the client continues the FIX session he will receive all the pending News messages from the time of disconnection. When the client opts to begin a new FIX session, he receives all the News messages addressed to him that have been generated from the start of the session.

11.2 List of messages

Message	Description
News (Msg Type = B)	Used to receive text messages from the CCP supervisor. Also used to send text messages to the CCP supervisor

11.3 Message flow

Message reception

FIX client   FIX server

News ("B")

Sending message

FIX client   FIX server

News ("B")
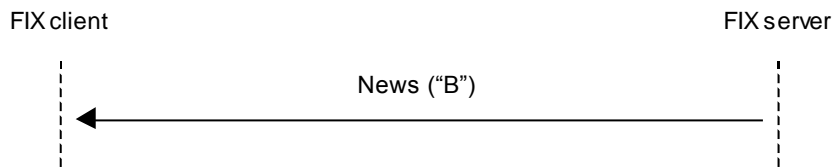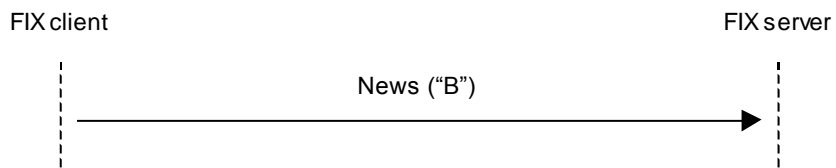
11.4 Annotations and adaptations of FIX 4.4

- Only one line of up to 78 characters per message is allowed

11.5 Definition of messages

11.5.1 News (Msg Type = B)

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = B		
61	Urgency	N	0 = Normal 1 = Flash 2 = Background	Char	The default value is 0
148	Headline	Y		String	Message header. Ignored by MEFFGate
33	LinesOfText	Y	1	NumInGroup	Number of lines of text. Only one line allowed
→58	Text	Y		String(78)	One line of text
	Standard Trailer	Y			

12. Give-up References and Filters Management

12.1 Introduction

Give-up references and filters management covers various functions. From the perspective of a FIX client these are:

- Give-out references management by the Executing Broker

- Give-in references management by the Clearing Broker

- Give-in filters management by the Clearing Broker

- Give-in filters management by the Clearing Member

- Automatic Give-out requests management

- Give-up references and filters report

Each of these functions is explained in a separate section, including a description of the method of use, the list of related messages, the message flow, the additions or annotations incorporated in this implementation for each function and a detailed description of the messages.

12.2 RegistID

The field RegistID, present in a request initiated by a Registration Instructions message, is the identifier that relates to the request with Registration Instructions Response messages.

The field RegistID assigned by the client should be ten characters length. If length is inferior, MEFFGate complete with spaces to achieve that length. MEFFGate also expects that messages sent by the client use an RegistID of 30 length, in this case only the last ten positions can be fixed free, since the 20 first should coincide with the format explained below.

Message sending with RegistID unprefixed

Message sending with RegistID prefixed
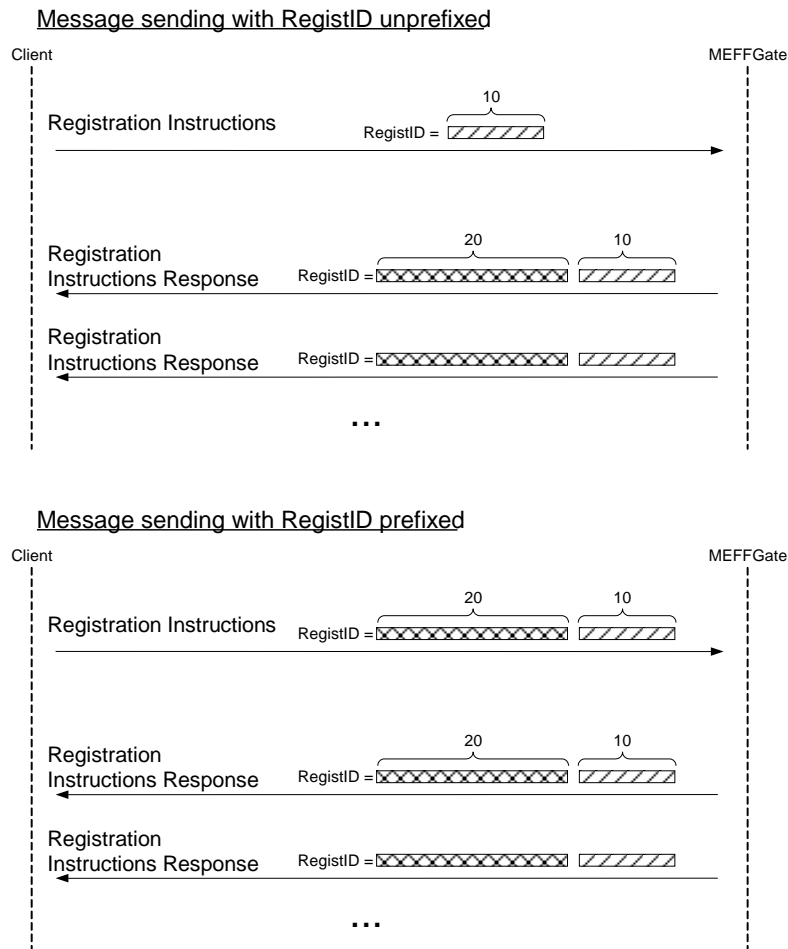
- Assigning of the prefix to the RegistID identifier -

A process in the MEFFGate of assigning a prefix to the RegistID field is performed to avoid duplicates in this identifier.

The RegistID assigned by MEFFGate in the reply message has the format AAMMDDMmmmTttMmmmTttNnnnnnnnnn, made by the following codes:

- **AAMMDD**. It is the date of the Clearing Contract Group session

- **MmmmTtt**. Contains the member and user code of connection from which the request was made

- **Nnnnnnnnnn**. It is the value assigned by the client application to AllocID in the original message

A user who wants to modify or cancel a reference or a Give-up filter, must use this identifier in the field RegistRefID of the Registration Instructions request message.

12.3 Give-out References management by the Executing Broker

12.3.1 Description

The FIX client uses this function for the maintenance of the references used by the Executing Broker when a Give-out request is sent.

These references will be common for all the Member's traders and can be modified in realtime

In the Give-out request, the Executing Broker must indicate a reference that enables the Clearing Broker to uniquely identify the source of the Give-in (together with the code of the Executing Broker). This is the "Give-up Reference".

The Executing and Clearing Brokers should make an agreement to establish this common reference.

To facilitate the sending of a Give-out and internal management, Mnemonic Reference and Internal Reference can be created. These codes will be defined by the Executing Broker and they do not require the agreement of the Clearing Broker

12.3.2 List of messages

Message	Description
Registration Instructions (Msg Type = o)	Used by the client to request the Give-out references management by the Executing Broker
Registration Instructions Response (Msg Type = p)	Sent by MEFFGate to notify or reject the Give-out references management by the Executing Broker

12.3.3 Message flow

Give-out references management
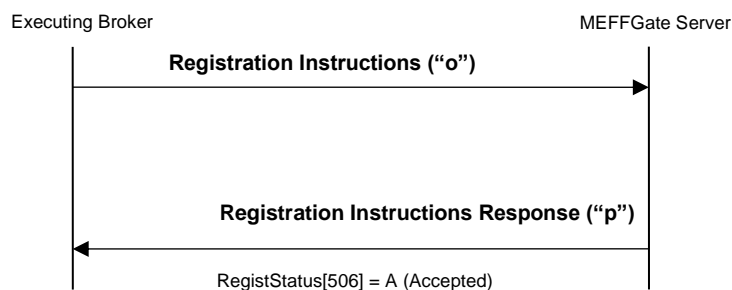
Incorrect Give-out references management
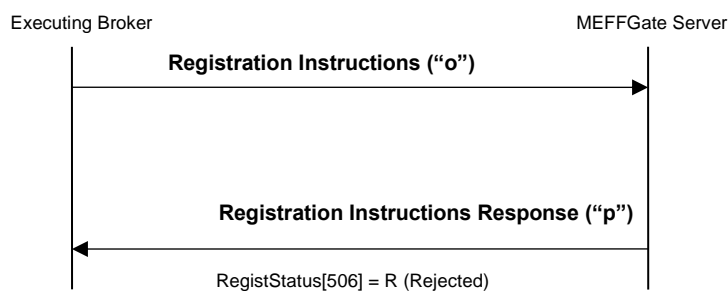
12.3.4 Annotations and adaptations of FIX 4.4

- In the Registration Instructions message, the fields NoPartyIDs (453) and NoPartySubIDs (802) are now required

- The field Text (58) has been added to the Registration Instructions Response message

12.3.5 Definition of messages

12.3.5.1 Registration Instructions (Msg Type = o)

Message sent by the client to manage the Give-out references.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = o		
513	RegistID	Y		String (30)	Unique identifier for each Registration Instructions message
514	RegistTransType	Y	0 = New 1 = Replace 2 = Cancel	Char	
508	RegistRefID	N		String (30)	Reference identifier for the RegistID (513) with Cancel and Replace RegistTransType (514) transaction types. Required if RegistTransType = 1 or 2
	Start \<Parties\>				
453	NoPartyIDs	Y*		NumInGroup	
→ 448	PartyID	Y		String	Member code or reference. For the Give-out internal Referencia (PartyRole=3) the maximum length for this field is 18 alphanumeric characters. For the Clearing Broker (PartyRole = 14) the exact length for this field is 4 alphanumeric characters. For the Give-up reference (PartyRole = 24) the maximum length for this field is 18 alphanumeric characters. For the Give-out mnemonic (PartyRole = 33) the maximum length for this field is 10 alphanumeric characters.
→ 447	PartyIDSource	Y	D = Proprietary / Custom code	String	
→ 452	PartyRole	Y	3 = Give-out internal reference 14 = Give-up Clearing Firm 24 = Give-up Reference 33 = Give-up Mnemonic	Int	Value 3 is for Give-out internal reference Value 14 is the Clearing Broker. Value 24 is for Give-up reference. Value 33 is for Give-out mnemonic.
→ 802	NoPartySubIDs	Y*	1	NumInGroup	
→→ 523	PartySubID	Y	GOR = Give-out references	String	
→→ 803	PartySubIDType	Y		Int	This field is required by the standard. MEFFGate does not require this field to be present
	End \<Parties\>				
	Standard Trailer	Y			

12.3.5.2 Registration Instructions Response (Msg Type = p)

Message used by MEFFGate to indicate the status of the request initiated with the Registration Instructions message.

This message is only sent to the user who made the request.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = p		
513	RegistID	Y		String (30)	Identifier assigned by the client in the Registration Instructions message
514	RegistTransType	Y	0 = New 1 = Replace 2 = Cancel	Char	
508	RegistRefID	N		String (30)	Identifier of Registration Instructions message which is replaced or cancelled by this message. Included when RegistTransType = 1 or 2
	Start <Parties>				
453	NoPartyIDs	N		NumInGroup	
→ 448	PartyID	N		String	Member code or reference.
→ 447	PartyIDSource	N	D = Proprietary / Custom code	String	
→ 452	PartyRole	N	3 = Give-out internal reference 14 = Give-up Clearing Firm 24 = Give-up Reference 33 = Give-up Mnemonic	Int	Value 3 for Give-out internal reference Value 14 is the Clearing Broker. Value 24 is for Give-up reference. Value 33 is for Give-out mnemonic.
→ 802	NoPartySubIDs	N	1	NumInGroup	
→→ 523	PartySubID	Y	GOR = Give-out references	String	
→→ 803	PartySubIDType	Y		Int	The content of this field should not be considered. It is included to meet requirements of the standard
	End <Parties>				
506	RegistStatus	Y	A = Accepted R = Rejected	Char	Status of the Registration Instructions request message. If it contains the value "R", there is an explanation for the rejection in the Text field
58*	Text	N		String	If RegistStatus = "R" there is an explanation of the rejection
	Standard Trailer	Y			

12.4 Give-in References Management by the Clearing Broker

12.4.1 Description

The FIX client uses this function for the maintenance of the references used by the Clearing Broker when a Give-in acceptance is requested.

These references will be common for all the Member's traders and can be modified in realtime

In the Give-out request, the Executing Broker must indicate a reference that enables the Clearing Broker to uniquely identify the source of the Give-in (together with the code of the Executing Broker). This is the "Give-up Reference".

The Executing and Clearing Brokers should make an agreement to establish this common reference.

To facilitate the acceptance of a Give-in, a mnemonic reference can be created. This code will be defined by the Clearing Broker and its does not require the agreement of the Executing Broker.

12.4.2 List of messages

Message	Description
Registration Instructions (Msg Type = o)	Used by the client to request the Give-in references management by the Clearing Broker
Registration Instructions Response (Msg Type = p)	Sent by MEFFGate to notify or reject the Give-in references management by the Clearing Broker

12.4.3 Message flow

Give-in references management

Clearing Broker   MEFFGate Server

Registration Instructions ("o")

Registration Instructions Response ("p")

RegistStatus[506] = A (Accepted)
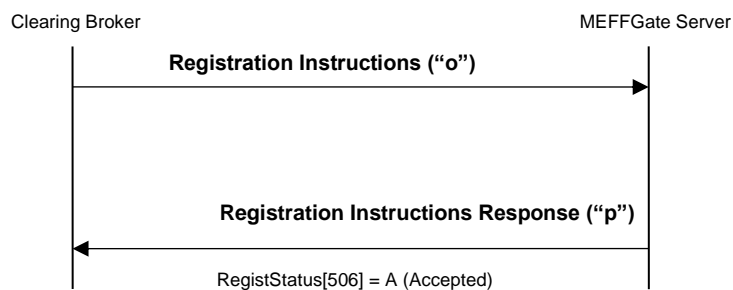
Incorrect Give-in references management

Clearing Broker   MEFFGate Server

Registration Instructions ("o")

Registration Instructions Response ("p")

RegistStatus[506] = R (Rejected)
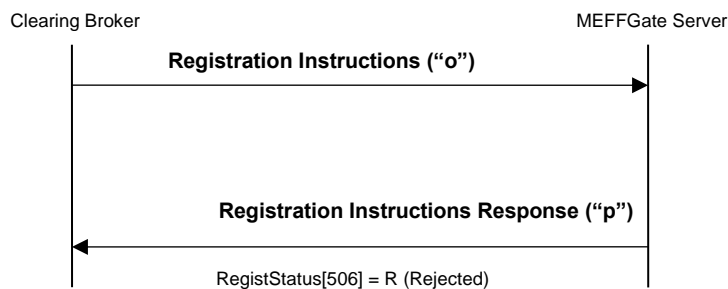
12.4.4 Annotations and adaptations of FIX 4.4

- In the Registration Instructions message, the fields NoPartyIDs (453), NoPartySubIDs (802) and Account (1) are now required

- The field Text (58) has been added to the Registration Instructions Response message

12.4.5 Definition of messages

12.4.5.1 Registration Instructions (Msg Type = o)

Message sent by the client to manage the Give-in references.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = o		
513	RegistID	Y		String (30)	Unique identifier for each Registration Instructions message
514	RegistTransType	Y	0 = New 1 = Replace 2 = Cancel	Char	
508	RegistRefID	N		String (30)	Reference identifier for the RegistID (513) with Cancel and Replace RegistTransType (514) transaction types. Required if RegistTransType = 1 or 2
	Start <Parties>				
453	NoPartyIDs	Y*	3	NumInGroup	
→ 448	PartyID	Y		String	Member code or reference. For the Executing Broker (PartyRole = 1) the maximum length for this field is 4 alphanumeric characters. For the Give-up reference (PartyRole = 24) the maximum length for this field is 18 alphanumeric characters. For the Give-in mnemonic (PartyRole = 33) the maximum length for this field is 10 alphanumeric characters.
→ 447	PartyIDSource	Y	D = Proprietary / Custom code	String	
→ 452	PartyRole	Y	1 = Executing Firm 24 = Give-up Reference 33 = Give-up Mnemonic	Int	Value 1 is the Executing Broker. Value 24 is for Give-up reference. Value 33 is for Give-in mnemonic.
→ 802	NoPartySubIDs	Y*	1	NumInGroup	
→→ 523	PartySubID	Y	GIR = Give-in references	String	
→→ 803	PartySubIDType	Y		Int	This field is required by the standard. MEFFGate does not require this field to be present
	End <Parties>				
1	Account	Y*		String (5)	Give-in destination account
	Standard Trailer	Y			

12.4.5.2 Registration Instructions Response (Msg Type = p)

Message used by MEFFGate to indicate the status of the request initiated with the Registration Instructions message.

This message is only sent to the user who made the request.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = p		
513	RegistID	Y		String	Identifier assigned by the client in the Registration Instructions message
514	RegistTransType	Y	0 = New 1 = Replace 2 = Cancel	Char	
508	RegistRefID	N		String	Identifier of Registration Instructions message which is replaced or cancelled by this message. Included when RegistTransType = 1 or 2
	Start <Parties>				
453	NoPartyIDs	N		NumInGroup	
→ 448	PartyID	N		String	Member code or reference.
→ 447	PartyIDSource	N	D = Proprietary / Custom code	String	
→ 452	PartyRole	N	1 = Executing Firm 24 = Give-up Reference 33 = Give-up Mnemonic	Int	Value 1 is the Executing Broker. Value 24 is for Give-up reference. Value 33 is for Give-in mnemonic.
→ 802	NoPartySubIDs	N	1	NumInGroup	
→→ 523	PartySubID	Y	GIR = Give-in references	String	
→→ 803	PartySubIDType	Y		Int	The content of this field should not be considered. It is included to meet requirements of the standard
	End <Parties>				
1	Account	N		String	Give-in destination account
506	RegistStatus	Y	A = Accepted R = Rejected	Char	Status of the Registration Instructions request message. If it contains the value "R", there is an explanation for the rejection in the Text field
58*	Text	N		String	If RegistStatus = "R" there is an explanation of the rejection
	Standard Trailer	Y			

12.5 Give-in Filters Management by the Clearing Broker

12.5.1 Description

The FIX client uses this function to configure the filters that allow for the automatic acceptance of trades.

The acceptance of a Give-in can be automated using certain filters defined by the Clearing Broker and/or Clearing Member, where any Give-in that does not pass those filters is left pending to be manually accepted or rejected.

12.5.2 List of messages

Message	Description
Registration Instructions (Msg Type = o)	Used by the client to request the Give-in filters management by the Clearing Broker
Registration Instructions Response (Msg Type = p)	Sent by MEFFGate to notify or reject the Give-in filters management by the Clearing Broker

12.5.3 Message flow

Give-in filters management by Clearing Broker

Clearing Broker   MEFFGate Server

Registration Instructions ("o")

Registration Instructions Response ("p")

RegistStatus[506] = A (Accepted)

Incorrect Give-in filters management by Clearing Broker

Clearing Broker   MEFFGate Server

Registration Instructions ("o")

Registration Instructions Response ("p")

RegistStatus[506] = R (Rejected)
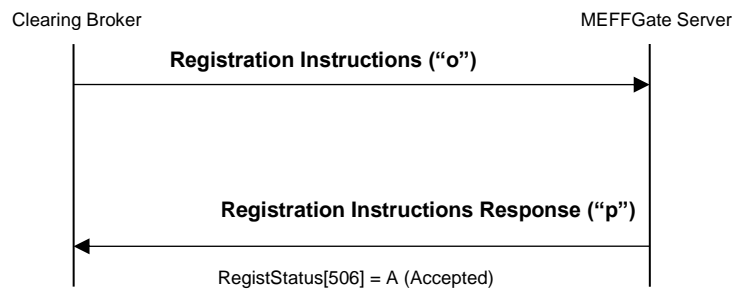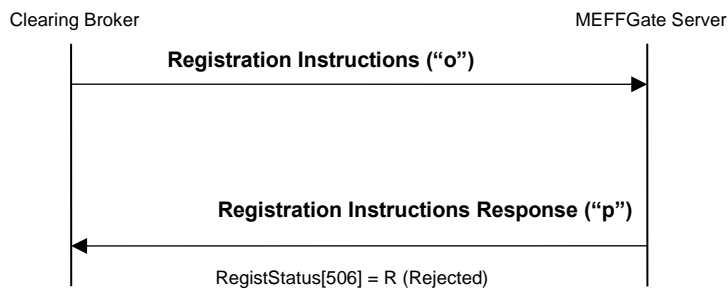
12.5.4 Annotations and adaptations of FIX 4.4

- In the Registration Instructions message, the fields NoPartyIDs (453) and NoPartySubIDs (802) are now required

- The field Text (58) has been added to the Registration Instructions Response message

12.5.5 Definition of messages

12.5.5.1 Registration Instructions (Msg Type = o)

Message sent by the client to manage the Give-in filters by the Clearing Broker.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = o		
513	RegistID	Y		String (30)	Unique identifier for each Registration Instructions message
514	RegistTransType	Y	0 = New 1 = Replace 2 = Cancel	Char	
508	RegistRefID	N		String (30)	Reference identifier for the RegistID (513) with Cancel and Replace RegistTransType (514) transaction types. Required if RegistTransType = 1 or 2
	Start <Parties>				
453	NoPartyIDs	Y*	2	NumInGroup	
→ 448	PartyID	Y		String	Member code or reference. For the Executing Broker (PartyRole = 1) the maximum length for this field is 4 alphanumeric characters. For the Give-up reference (PartyRole = 24) the maximum length for this field is 18 alphanumeric characters.
→ 447	PartyIDSource	Y	D = Proprietary / Custom code	String	
→ 452	PartyRole	Y	1 = Executing Firm 24 = Give-up Reference	Int	Value 1 is the Executing Broker. Value 24 is for Give-up reference. The use of the wildcard "?" is only allowed in all the Executing Broker positions or in all the Give-up reference positions (but not in both fields simultaneously).
→ 802	NoPartySubIDs	Y*	1	NumInGroup	
→→ 523	PartySubID	Y	GIF = Give-in filters	String	
→→ 803	PartySubIDType	Y		Int	This field is required by the standard. MEFFGate does not require this field to be present
	End <Parties>				
	Start <Stipulations>				
232*	NoStipulations	Y	2	NumInGroup	
→ 233*	StipulationType	Y	TAL = Transaction Amount Limit SAL = Session Amount Limit	String	
→ 234*	StipulationValue	Y	[N/A] or an numeric value >=0 and <= 999999999	String	When StipulationType = TAL, contains the maximum amount for a Give-in that will be accepted automatically for this Executing Broker and reference. When StipulationType = SAL, contains the maximum amount per session of Give-ins that will be accepted automatically for this Executing Broker and reference. Must contain [N/A] when there is no a maximum amount to validate.

Tag	Name	Req	Valid values	Format	Description
	End <Stipulations>				
	Standard Trailer	Y			

12.5.5.2 Registration Instructions Response (Msg Type = p)

Message used by MEFFGate to indicate the status of the request initiated with the Registration Instructions message.

This message is only sent to the user who made the request.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = p		
513	RegistID	Y		String	Identifier assigned by the client in the Registration Instructions message
514	RegistTransType	Y	0 = New 1 = Replace 2 = Cancel	Char	
508	RegistRefID	N		String	Identifier of Registration Instructions message which is replaced or cancelled by this message. Included when RegistTransType = 1 or 2
	Start <Parties>				
453	NoPartyIDs	N		NumInGroup	
→ 448	PartyID	N		String	Member code or reference. It is possible to use the wildcard "?" character.
→ 447	PartyIDSource	N	D = Proprietary / Custom code	String	
→ 452	PartyRole	N	1 = Executing Firm 24 = Give-up Reference	Int	Value 1 is the Executing Broker. Value 24 is for Give-up reference.
→ 802	NoPartySubIDs	N		NumInGroup	
→→ 523	PartySubID	Y	GIF = Give-in filters	String	
→→ 803	PartySubIDType	Y		Int	The content of this field should not be considered. It is included to meet requirements of the standard
	End <Parties>				
	Start <Stipulations>				
232*	NoStipulations	N		NumInGroup	
→ 233*	StipulationType	N	TAL=Transaction Amount Limit SAL = Session Amount Limit	String	
→ 234*	StipulationValue	N		String	When StipulationType = TAL, contains the maximum amount for a Give-in that will be accepted automatically for this Executing Broker and reference. When StipulationType = SAL, contains the maximum amount per session of Give-ins that will be accepted automatically for this Executing Broker and reference. This field contains [N/A] when there is no a maximum amount to validate.
	End <Stipulations>				
506	RegistStatus	Y	A = Accepted R = Rejected	Char	Status of the Registration Instructions request message. If it contains the value "R", there is an explanation for the rejection in the Text field
58*	Text	N		String	If RegistStatus = "R" there is an explanation of the rejection

Tag	Name	Req	Valid values	Format	Description
	Standard Trailer	Y			

12.6 Give-in Filters Management by the Clearing Member

12.6.1 Description

The FIX client uses this function to configure the filters that allow for the automatic acceptance of trades.

The acceptance of a Give-in can be automated using certain filters defined by the Clearing Broker and/or Clearing Member, where any Give-in that does not pass those filters is left pending to be manually accepted or rejected.

12.6.2 List of messages

Message	Description
Registration Instructions (Msg Type = o)	Used by the client to request the Give-in filters management by the Clearing Member
Registration Instructions Response (Msg Type = p)	Sent by MEFFGate to notify or reject the Give-in filters management by the Clearing Member

12.6.3 Message flow

Give-in filters management by Clearing Member

Clearing Member   MEFFGate Server

Registration Instructions ("o")

Registration Instructions Response ("p")

RegistStatus[506] = A (Accepted)

Incorrect Give-in filters management by Clearing Member

Clearing Member   MEFFGate Server

Registration Instructions ("o")

Registration Instructions Response ("p")

RegistStatus[506] = R (Rejected)
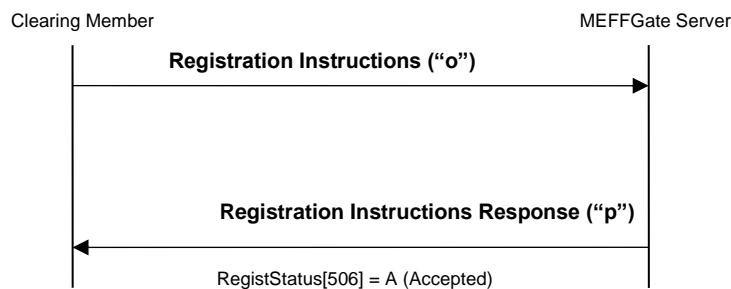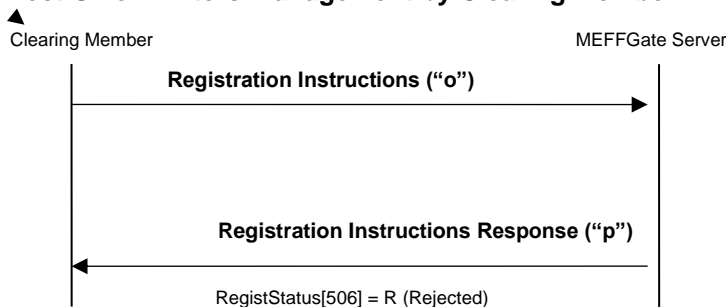
12.6.4 Annotations and adaptations of FIX 4.4

- In the Registration Instructions message, the fields NoPartyIDs (453) and NoPartySubIDs (802) are now required

- The field Text (58) has been added to the Registration Instructions Response message

- The block Stipulations has been added to the Registration Instructions Response message

12.6.5 Definition of messages

12.6.5.1 Registration Instructions (Msg Type = o)

Message sent by the client to manage the Give-in filters by the Clearing Member.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = o		
513	RegistID	Y		String (30)	Unique identifier for each Registration Instructions message
514	RegistTransType	Y	0 = New 1 = Replace 2 = Cancel	Char	
508	RegistRefID	N		String (30)	Reference identifier for the RegistID (513) with Cancel and Replace RegistTransType (514) transaction types. Required if RegistTransType = 1 or 2
	Start <Parties>				
453	NoPartyIDs	Y*	1	NumInGroup	
→ 448	PartyID	Y		String (4)	Clearing Broker code
→ 447	PartyIDSource	Y	D = Proprietary / Custom code	String	
→ 452	PartyRole	Y	14 = Give-up Clearing Firm	Int	
→ 802	NoPartySubIDs	Y*	1	NumInGroup	
→→ 523	PartySubID	Y	GIFCM = Give-in filters of clearing member	String	
→→ 803	PartySubIDType	Y		Int	This field is required by the standard. MEFFGate does not require this field to be present
	End <Parties>				
1	Account	Y*		String (5)	Give-in destination account. The use of the wildcard "?" is only allowed in all the five positions.
	Start <Stipulations>				
232*	NoStipulations	Y	2	NumInGroup	
→ 233*	StipulationType	Y	TAL=Transaction Amount Limit SAL = Session Amount Limit	String	
→ 234*	StipulationValue	Y	[N/A] or an numeric value >=0 and <= 999999999	String	When StipulationType = TAL, contains the maximum amount for a Give-in that will be accepted automatically for this Clearing Broker and account. When StipulationType = SAL, contains the maximum amount per session of Give-ins that will be accepted automatically for this Clearing Broker and account. Must contain [N/A] when there is no a maximum amount to validate.
	End <Stipulations>				
	Standard Trailer	Y			

12.6.5.2 Registration Instructions Response (Msg Type = p)

Message used by MEFFGate to indicate the status of the request initiated with the Registration Instructions message.

This message is only sent to the user who made the request.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = p		
513	RegistID	Y		String	Identifier assigned by the client in the Registration Instructions message
514	RegistTransType	Y	0 = New 1 = Replace 2 = Cancel	Char	
508	RegistRefID	N		String	Identifier of Registration Instructions message which is replaced or cancelled by this message. Included when RegistTransType = 1 or 2
	Start <Parties>				
453	NoPartyIDs	N		NumInGroup	
→ 448	PartyID	N		String	Clearing Broker code
→ 447	PartyIDSource	N	D = Proprietary / Custom code	String	
→ 452	PartyRole	N	14 = Give-up Clearing Firm	Int	
→ 802	NoPartySubIDs	N	1	NumInGroup	
→→ 523	PartySubID	Y	GIFCM = Give-in filters of clearing member	String	
→→ 803	PartySubIDType	Y		Int	The content of this field should not be considered. It is included to meet requirements of the standard
	End <Parties>				
1	Account	N		String	Give-in destination account. It is possible to use the wildcard "?" character.
	Start <Stipulations>				
232*	NoStipulations	N		NumInGroup	
→ 233*	StipulationType	N	TAL=Transaction Amount Limit SAL = Session Amount Limit	String	
→ 234*	StipulationValue	N		String	When StipulationType = TAL, contains the maximum amount for a Give-in that will be accepted automatically for this Clearing Broker and account. When StipulationType = SAL, contains the maximum amount per session of Give-ins that will be accepted automatically for this Clearing Broker and account. This field contains [N/A] when there is no a maximum amount to validate.
	End <Stipulations>				
506	RegistStatus	Y	A = Accepted R = Rejected	Char	Status of the Registration Instructions request message. If it contains the value "R", there is an explanation for the rejection in the Text field
58*	Text	N		String	If RegistStatus = "R" there is an

Tag	Name	Req	Valid values	Format	Description
					explanation of the rejection
	Standard Trailer	Y			

12.7 Automatic Give-out Requests Management

12.7.1 Description

The FIX client uses this function to configure the automatic give-out initiation, based on the account code, for trades originated in the trading engine (order matching and cross orders).

12.7.2 List of messages

Message	Description
Registration Instructions (Msg Type = o)	Used by the client to request the automatic Give-out management
Registration Instructions Response (Msg Type = p)	Sent by MEFFGate to notify or reject the automatic Give-out request management

12.7.3 Message flow

Automatic Give-out request management
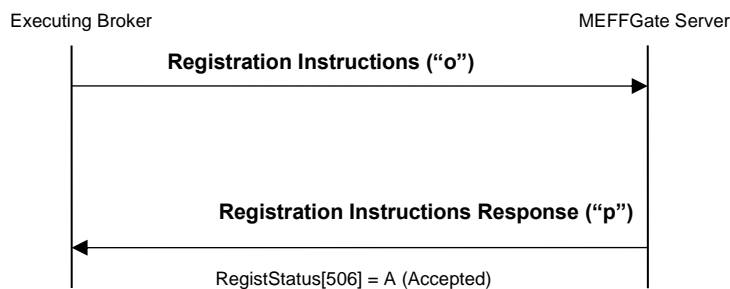
Incorrect automatic Give-out request management
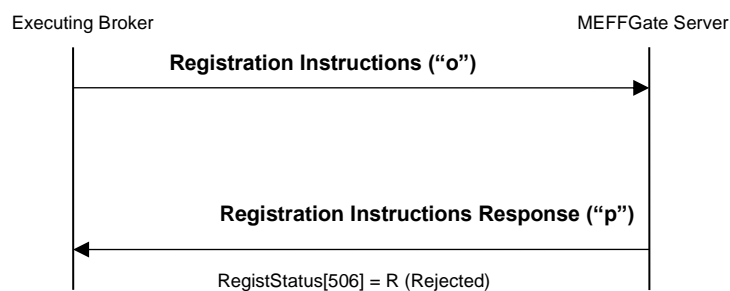
12.7.4 Annotations and adaptations of FIX 4.4

- In the Registration Instructions message, the fields NoPartyIDs (453) and NoPartySubIDs (802) are now required

- The field Text (58) has been added to the Registration Instructions Response message

12.7.5 Definition of messages

12.7.5.1 Registration Instructions (Msg Type = o)

Message sent by the client to manage the automatic Give-out requests.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = o		
513	RegistID	Y		String (30)	Unique identifier for each Registration Instructions message
514	RegistTransType	Y	0 = New 1 = Replace 2 = Cancel	Char	
508	RegistRefID	N		String (30)	Reference identifier for the RegistID (513) with Cancel and Replace RegistTransType (514) transaction types. Required if RegistTransType = 1 or 2
	Start <Parties>				
453	NoPartyIDs	Y*	1	NumInGroup	
→ 448	PartyID	Y		String(10)	Give-out mnemonic
→ 447	PartyIDSource	Y	D = Proprietary / Custom code	String	
→ 452	PartyRole	Y	33 = Give-up Mnemonic	Int	
→ 802	NoPartySubIDs	Y*	1	NumInGroup	
→→ 523	PartySubID	Y	AGR = Automatic Give-out request	String	
→→ 803	PartySubIDType	Y		Int	This field is required by the standard. MEFFGate does not require this field to be present
	End <Parties>				
1	Account	Y*		String (5)	Give-out account
	Standard Trailer	Y			

12.7.5.2 Registration Instructions Response (Msg Type = p)

Message used by MEFFGate to indicate the status of the request initiated with the Registration Instructions message.

This message is only sent to the user who made the request.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = p		
513	RegistID	Y		String	Identifier assigned by the client in the Registration Instructions message
514	RegistTransType	Y	0 = New 1 = Replace 2 = Cancel	Char	
508	RegistRefID	N		String	Identifier of Registration Instructions message which is replaced or cancelled by this message. Included when RegistTransType = 1 or 2
	Start <Parties>				
453	NoPartyIDs	N		NumInGroup	
→ 448	PartyID	N		String	Give-out mnemonic
→ 447	PartyIDSource	N	D = Proprietary / Custom code	String	
→ 452	PartyRole	N	33 = Give-up Mnemonic	Int	
→ 802	NoPartySubIDs	N	1	NumInGroup	
→→ 523	PartySubID	Y	AGR = Automatic Give-out request	String	
→→ 803	PartySubIDType	Y		Int	The content of this field should not be considered. It is included to meet requirements of the standard
	End <Parties>				
1	Account	N		String	Give-out account
506	RegistStatus	Y	A = Accepted R = Rejected	Char	Status of the Registration Instructions request message. If it contains the value "R", there is an explanation for the rejection in the Text field
58*	Text	N		String	If RegistStatus = "R" there is an explanation of the rejection
	Standard Trailer	Y			

12.8 Give-up References and Filters Report

12.8.1 Description

The client application can request a query of the Give-up references and filters with a Registration Instructions message.

In reply to this request a Registration Instructions Response message is received for each Give-up reference and filter available at this moment.

12.8.2 List of messages

Message	Description
Registration Instructions (Msg Type = o)	Used by the client to request the Give-up references and filters report
Registration Instructions Response (Msg Type = p)	Sent by MEFFGate to notify or reject the Give-up references and filters report

12.8.3 Message flow

Give-up references and filters information request

MEFFGate Client   MEFFGate Server

Registration Instructions ("o")

SubscriptionRequestType = 0 (Snaphot)

Registration Instructions Response ("p")

RegistStatus[506] = A (Accepted)

…
One mensaje per every reference
and/or Give-up filter
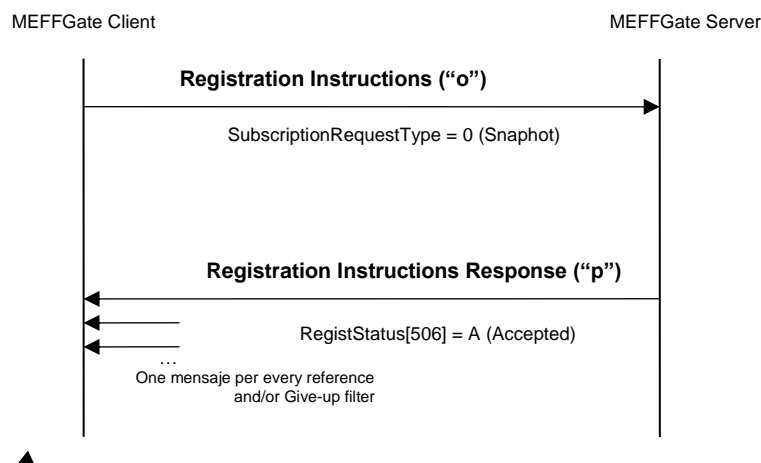
Incorrect Give-up references and filters information request

MEFFGate Client   MEFFGate Server

Registration Instructions ("o")

SubscriptionRequestType = 0 (Snaphot)

Registration Instructions Response ("p")

RegistStatus[506] = R (Rejected)
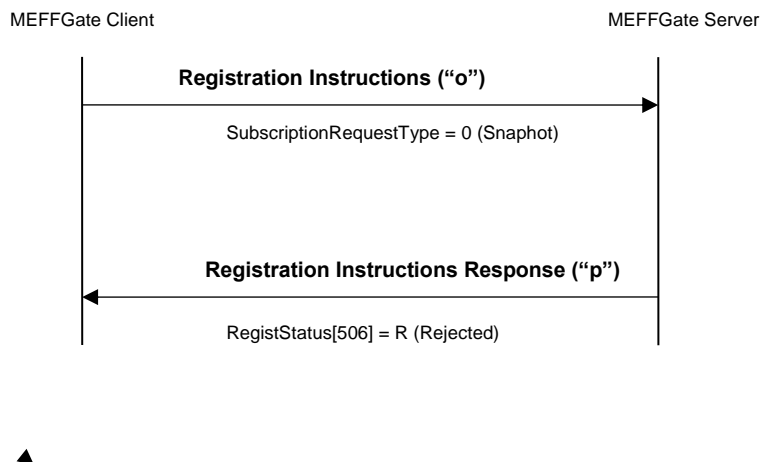
12.8.4 Annotations and adaptations of FIX 4.4

- The field SubscriptionRequestType (263) has been added to the Registration Instructions message

- The fields LastRptRequested (912) and Text (58) have been added to the Registration Instructions Response message

12.8.5 Definition of messages

12.8.5.1 Registration Instructions (Msg Type = o)

Message sent by the client to request the Give-up references and filters report.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = o		
513	RegistID	Y		String (10)	Unique identifier for each Registration Instructions message
514	RegistTransType	Y		Char	Ignored by MEFFGate when SubscriptionRequestType[263] is specified
263*	SubscriptionRequestType	N	0 = Snasphot	Char	If this tag is specified indicates this is a request for all the existing references and filters. In this case MEFFGate ignores the rest of the fields.
	Standard Trailer	Y			

12.8.5.2 Registration Instructions Response (Msg Type = p)

Message used by MEFFGate to indicate the status of the request initiated with the Registration Instructions message.

This message is only sent to the user who made the request.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = p		
513	RegistID	Y		String (10)	Identifier assigned by the client in the Registration Instructions message
514	RegistTransType	Y	0 = New	Char	
912*	LastRptRequested	N	Y = Last message N = Not last message	Boolean	
	… <Parties> Account <Stipulations> …				Spefific infomation depending on the register type.
506	RegistStatus	Y	A = Accepted R = Rejected	Char	Status of the Registration Instructions request message. If it contains the value "R", there is an explanation for the rejection in the Text field
58*	Text	N		String	If RegistStatus = "R" there is an explanation of the rejection
	Standard Trailer	Y			

13.   Deliveries of the Futures Contract on the Government Bond

13.1 Introduction

The deliveries functionality allows a FIX client to:

a) Obtain the published deliverables, via the Security List message, in the block InstrumentLeg

b) Inform, during the period of notification of deliverables, what references, from the list of deliverables, will be delivered for each sold position. This is communicated through the Position Maintenance Request messages

c) Be informed of operations to be made in IBERCLEAR. These operations are communicated via Trade Capture Report messages received via a specific subscription using a Trade Capture Report Request message with TrdType [828] = 15.

Below each of these phases is described in more detail.

13.2 List of deliverable securities

The list of deliverable securities and conversion factors are informed in the InstrumentLeg of the Security List message. The most relevant information would be:

- Derivatives contract: Symbol [55]

- Delivery security code: LegSymbol [600]

- ISIN delivery security code: LegSecurityAltID [605]

- Conversion factor: LegFactor [253]

- Accrued interest: AccruedInterestAmt [159]

13.3 Delivery notice

Delivery commences for Members with short positions, for which MEFF provides a delivery notification period during which each Member will inform MEFF of which references from the delivery securities list (see 13.2 - List of deliverable securities) it will deliver for each short position. When this period has finished, if there are any short positions for which MEFF has not been notified of the reference to be delivered, the Clearing Member must perform notificattion to MEFF. Finally, if there are still any positions for which the deliverable reference has not been notified, MEFF will assign the reference itself, selecting the cheapest to deliver in preference.

This delivery notification is used through the Position Maintenance Request message. The most relevant information would be:

- PosTransType [709] = 4 (Delivery)

- Symbol [55] = Derivatives contract

- Account [1] = Holder

- PosType [703] = DN (Delivery Notice Qty)

- ShortQty [705]: Number of contracts to deliver

- NestedPartyRole [538] = 13 (when the notification is made by the Trading Member or 4 (when the notification is made by the Clearing Member)

- LegSymbol [600]: Reference to deliver

Please note that whenever Position Maintenance Request modification is submitted, the last one completely replaces the old one. So, every time it is necessary to send all delivery notification information.

For more information, please see "13.3 – Delivery notice"

13.4 Purchase/sale trades to carry out

The operations to be made in IBERCLEAR are communicated via Trade Capture Report messages received via a specific subscription using a Trade Capture Report Request message with TrdType [828] = 15. There are 2 different levels of information:

- Spot trades broken down by holder. The Account [1] field is 3 characters long (Holder associated with the trade)

- Spot trades by member. The Account [1] field is not informed

13.5 List of messages

Message	Description
Security List (Msg Type = y)	Sent by MEFFGate to inform the list of deliverable securities and conversion factors
Position Maintenance Request (Msg Type = AL)	Sent by the FIX client to inform, during the period of notification of deliverables, what references, from the list of deliverables, will be delivered for each sold position
Position Maintenance Report (Msg Type = AM)	Sent by MEFFGate to indicate the acceptance or rejection of a delivery notice
Trade Capture Report Request (Msg Type = AD)	Specific request for delivery trades via the specific option TrdType [828] = 15
Trade Capture Report Request Ack (Msg Type = AQ)	Acknowledgement of Trade Capture Report Request message
Trade Capture Report (Msg Type = AE)	Sent by MEFFGate to inform the operations to be made in IBERCLEAR as the delivery process result. Only sent to related clients subscribed to this type of message via the specific option TrdType [828] = 15

13.6 Message flow

List of deliverable securities request

MEFFGate Client   MEFFGate Server

Symbol [55] = ABC, …, NoLegs [555] = 2,
$LegSymbol_1$ [600] = A1, $LegFactor_1$[253] = 0.96680,…,
$LegSymbol_2$ [600] = A2, $LegFactor_2$[253] = 0.989058,…,
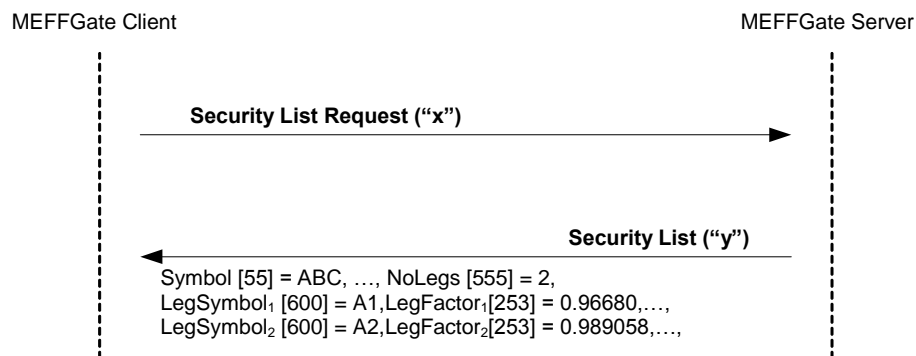
Delivery notice

Initially, MEFFGate sends an unsolicited Position Maintenance Report message with PosTransType [709] = 4 (Delivery), PosMaintAction [712] = 1 (New), PosReqID [710] informed, PosType [703] = DLV (Total Delivery Qty) and ShortQty [705] informed with the number of contracts to be delivered.

Then the client sends a delivery notice using the Position Maintenance Request message with PosTransType [709] = 4 (Delivery), PosMaintAction [712] = 2 (Replace), PosType [703] = DN (Delivery Notice Qty) and ShortQty [705] informed with the number of contracts that are delivered.

MEFFGate replies with a message accepting the instruction. The next delivery notice sent completely substitutes the previous order.

MEFFGate Client
Clearing
Member
LLLL Trading
Member
NNNN   MEFFGate Server

Position Maintenance Report ("AM")

PosTransType [709] = 4 (Delivery), PosMaintAction [712] = 1 (New), PosReqID [710] = X1,
Symbol [55] = FB10U3, Account [1] = 00P

NoPositions [702] = 1,
PosType [703] = DLV (Total Delivery Qty), ShortQty [705] = 175, NestedPartyRole = 13, NestedPartyID = NNNN

Position Maintenance Request ("AL")

PosTransType [709] = 4 (Delivery), PosMaintAction [712] = 2 (Replace), PosReqID [710] = X2,
OrigPosReqRefID [713] = X1, Symbol [55] = FB10U3, Account [1] = 00P

NoPositions [702] = 3
PosType [703] = DN (Delivery Notice Qty), ShortQty [705] = 80, LegSymbol [600] = 5,50/10
NestedPartyRole = 13, NestedPartyID = NNNN

PosType [703] = DN (Delivery Notice Qty),ShortQty [705] = 75, LegSymbol [600] = 5,85/11
NestedPartyRole = 13, NestedPartyID = NNNN

PosType [703] = DLV (Total Delivery Qty), ShortQty [705] = 175, NestedPartyRole = 13, NestedPartyID = NNNN

Position Maintenance Report ("AM")

PosTransType [709] = 4 (Delivery), PosMaintAction [712] = 2 (Replace), PosReqID [710] = X2,
OrigPosReqRefID [713] = X1, Symbol [55] = FB10U3, Account [1] = 00P

NoPositions [702] = 3
PosType [703] = DLV (Total Delivery Qty), ShortQty [705] = 175, NestedPartyRole = 13, NestedPartyID = NNNN

PosType [703] = DN (Delivery Notice Qty), ShortQty [705] = 80, LegSymbol [600] = 5,50/10
NestedPartyRole = 13, NestedPartyID = NNNN

PosType [703] = DN (Delivery Notice Qty), ShortQty [705] = 75, LegSymbol [600] = 5,85/11
NestedPartyRole = 13, NestedPartyID = NNNN

Position Maintenance Request ("AL")

PosTransType [709] = 4 (Delivery), PosMaintAction [712] = 2 (Replace),, PosReqID [710] = Y1,
OrigPosReqRefID [713] = X2, Symbol [55] = FB10U3, Account [1] = 00P

NoPositions [702] = 4
PosType [703] = DN (Delivery Notice Qty), ShortQty [705] = 20, LegSymbol [600] = 5,50/10
NestedPartyRole = 4, NestedPartyID = LLLL

PosType [703] = DN (Delivery Notice Qty), ShortQty [705] = 80, LegSymbol [600] = 5,50/10
NestedPartyRole = 13, NestedPartyID = NNNN

PosType [703] = DN (Delivery Notice Qty),ShortQty [705] = 75, LegSymbol [600] = 5,85/11
NestedPartyRole = 13, NestedPartyID = NNNN

PosType [703] = DLV (Total Delivery Qty), ShortQty [705] = 175, NestedPartyRole = 13, NestedPartyID = NNNN

Position Maintenance Report ("AM")

PosTransType [709] = 4 (Delivery), PosMaintAction [712] = 2 (Replace),, PosReqID [710] = Y1,
OrigPosReqRefID [713] = X2, Symbol [55] = FB10U3, Account [1] = 00P

NoPositions [702] = 4
PosType [703] = DN (Delivery Notice Qty), ShortQty [705] = 20, LegSymbol [600] = 5,50/10
NestedPartyRole = 4, NestedPartyID = LLLL
PosType [703] = DN (Delivery Notice Qty), ShortQty [705] = 80, LegSymbol [600] = 5,50/10
NestedPartyRole = 13, NestedPartyID = NNNN
PosType [703] = DN (Delivery Notice Qty),ShortQty [705] = 75, LegSymbol [600] = 5,85/11
NestedPartyRole = 13, NestedPartyID = NNNN
PosType [703] = DLV (Total Delivery Qty), ShortQty [705] = 175, NestedPartyRole = 13, NestedPartyID = NNNN

Delivery notice rejected by the system

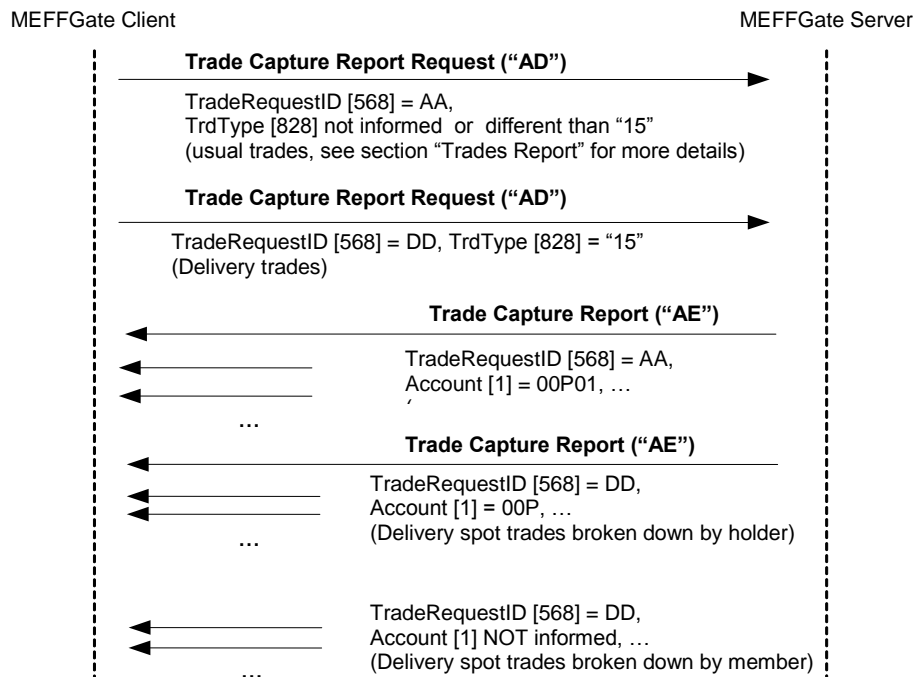If the FIX client sends a selivery notice instruction, it will be rejected with a Position Maintenance Report message with the field PosMaintStatus = 2. A rejected delivery notice will not alter a delivery notice sent previously.

MEFFGate Client   MEFFGate Server

Position Maintenance Request ("AL")

Position Maintenance Report ("AM")
PosMaintStatus [722] = 2 (Rejected)

Delivery spot trades

The delivery sport trades are communicated via Trade Capture Report messages received via a specific subscription using a Trade Capture Report Request message with TrdType [828] = 15. There are 2 different levels of information:

- Spot trades broken down by holder. The Account [1] field is 3 characters long (Holder associated with the trade)

- Spot trades by member. The Account [1] field is not informed

MEFFGate Client   MEFFGate Server

Trade Capture Report Request ("AD")

TradeRequestID [568] = AA,
TrdType [828] not informed or different than "15"
(usual trades, see section "Trades Report" for more details)

Trade Capture Report Request ("AD")

TradeRequestID [568] = DD, TrdType [828] = "15"
(Delivery trades)

Trade Capture Report ("AE")

TradeRequestID [568] = AA,
Account [1] = 00P01, …
…

Trade Capture Report ("AE")

TradeRequestID [568] = DD,
Account [1] = 00P, …
(Delivery spot trades broken down by holder)
…

TradeRequestID [568] = DD,
Account [1] NOT informed, …
(Delivery spot trades broken down by member)
…

13.7 Annotations and adaptations of FIX 4.4

- The LegSymbol [600] field has been added to the Position Report, Position Maintenance Request and Position Maintenance Report messages

13.8 Definition of messages

The Security List mentioned in this chapter is also used for the static contract information. The definition can be found in section 6.4.2 of chapter 6.

The Position Maintenance Request, Position Maintenance Report and Trade Capture Report messages mentioned in this chapter are also used for the monitoring and management of positions. The definition can be found in section 7.8.5, 7.8.6 and 7.8.4 of chapter 7.

The Trade Capture Report and Request Trade Capture Report Request Ack messages mentioned in this chapter are also used for the trades report. The definition can be found in sections 8.6.1 and 8.6.2 of chapter 8.

14.   Margins calculation portfolio

14.1 Introduction

This functionality allows Information concerning Margins during the session from a portfolio sent by the FIX client.

For this purpose, Margin Requirement Inquiry and Margin Requirement Report messages are used.

The Margin Requirement Inquiry message is used by the client application to send a portfolio with the position in each security; MEFFGate replies, using a Margin Requirement Inquiry message, with the estimated information concerning Margins at this point of the session.

The field MarginReqmtInqID [1635], present in a request initiated by a Margin Requirement Inquiry, is the identifier that relates to the request with the Margin Requirement Report message.
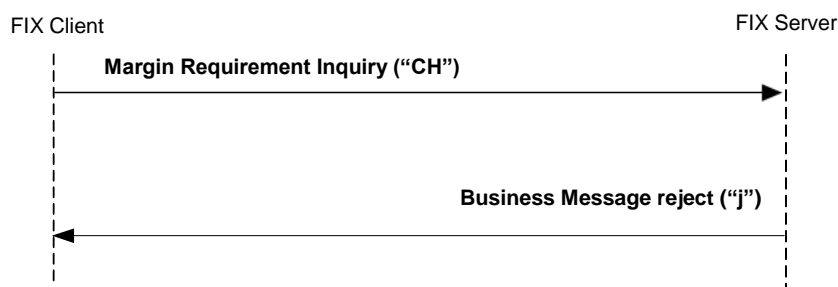
14.2 List of messages

Message	Description
Margin Requirement Inquiry (Msg Type = CH)	Used by the client to request information concerning Margins through a portfolio with the position in each security
Margin Requirement Report (Msg Type = CJ)	Sent by MEFFGate as a reply to a Margin Requirement Inquiry message with the estimated information concerning Margins at this point of the session

14.3 Message flow

Information concerning Margins sent in response to a Margin Requirement Inquiry message

FIX Client   FIX Server

Margin Requirement Inquiry ("CH")

Margin Requirement Report ("CJ")
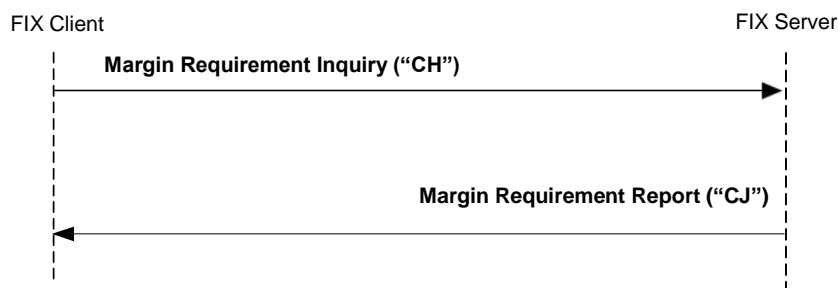
Reject message sent in response to a Margin Requirement Inquiry message

FIX Client   FIX Server

Margin Requirement Inquiry ("CH")

Business Message reject ("j")

14.4 Annotations and adaptations of FIX 4.4

- Stipulations block has been added to the Margin Requirement Inquiry and Margin Requirement Report messages. This block is required in the Margin Requirement Inquiry message.

14.5 Definition of messages

14.5.1 Margin Requirement Inquiry (Msg Type = CH)

Used by the client to request information concerning Margins through a portfolio with the position in each security

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = CH		
1635	MarginReqmtInqID	Y		String(30)	Unique identifier for each Margin Requirement Inquiry message
	Start <MarginReqmtInqQualGrp>				
1636	NoMarginReqmtInqQualifier	Y	1	NumInGroup	Always 1
→1637	MarginReqmtInqQualifier	Y	0 = Summary	Int	Always 0
	End <MarginReqmtInqQualGrp>				
	Start <Stipulations>				
232*	NoStipulations	Y*		NumInGroup	
→ 233*	StipulationType	Y*	SYMBOL = Contract code POS = Position	String	
→ 234*	StipulationValue	Y*		String	When StipulationType [233] = SYMBOL, contains the contract code. When StipulationType [233] = POS, contains the position in this contract (with sign).
	End <Stipulations>				
	Standard Trailer	Y			

14.5.2 Margin Requirement Report (Msg Type = CJ)

Sent by MEFFGate as a reply to a Margin Requirement Inquiry message with the estimated information concerning Margins at this point of the session

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = CJ		
1642	MarginReqmtRptID	Y		String	Unique identifier for each Margin Requirement Report message
1635	MarginReqmtInqID	Y		String	Identifier assigned by the client in the Margin Requirement Inquiry message
	Start \<MarginAmount\>				
1643	NoMarginAmt	Y	1	NumInGroup	Always 1
→1645	MarginAmt	Y		Amt	Estimated Margins at this point of the session
1638	MarginReqmtRptType	Y	0 = Summary	Int	
	End \<MarginAmount\>				
	Start \<Stipulations\>				
232*	NoStipulations	N		NumInGroup	
→ 233*	StipulationType	N	SYMBOL = Contract code POS = Position	String	
→ 234*	StipulationValue	N		String	When StipulationType [233] = SYMBOL, contains the contract code, as specified in the related Margin Requirement Report message. When StipulationType [233] = POS, contains the position in this contract (with sign), as specified in the related Margin Requirement Report message.
	End \<Stipulations\>				
	Standard Trailer	Y			

Appendix A User Fields

The following table shows the user fields that are found in the messages of this manual

Tag	Name	Format	Description
5680	ProprietaryFixProtocolVersion	String	Exact identification of the version of the protocol used and expected by the initiator
5679	FixEngineName	String	Contains a descriptive chain of software used by the client for the FIX connection. Only used for informative purposes
5681	ExchangeTradeType	String	Exchange trade type (see table 19 in document "Codification Tables")
5682	NewSecuritySubscription	Char	Field to request subscription to the definition of new contracts
5683	SecondaryConfirmStatus	Char	Describes the Give-up state

Appendix B Codification Tables

B.1 Security type in tag SecurityType [167]

Security type	Description
F	Future
O	Option
W	Swap
X	Other