# Hf MeffGate M5.13

Fix Interface Specifications (Public information)

12 May 2023

## Changes made in the latest revision

**Outlined below are the main changes made in the version M5.13 (since the public information of version M5.4 on 14 June 2018):**

– Trading mode. New value TradingSessionID [336] = 118 (xRolling on Stocks). Indication of Interest, Market Data Snapshot Full Refresh and Trading Session Status messages

– Trading mode. New value StipulationType [233] = 118 (xRolling on Stocks). Security List and Security List Update Report messages

– Security List and Security List Update Report: new EventType 167, 168, 169, 170, 171, 172 y 173 for xRolling stocks are added

– Modify description of MDOriginType [1024] value 0. Market Data Snapshot Full Refresh message

– Logon message sent by HF MEFFGate. New user field: Current business session date. BusinessSessionDate [21505].

**Outlined below are the main changes from the documentation published on 18 May 2021:**

– Adapt to new "Codification tables" document, unified for all BME Exchanges

– Security List and Security List Update Report: Correct mistake in LegPrice [566], this field belongs to group <InstrumentLeg>

– Market Data Snapshot Full Refresh message. The flag 865=212 covers both pre-transparency and post-transparency.

– Include the Indication of Interest operation in the RFQ section.

– New value for TradingSessionSubID [625]=204 in table 'Level2 - Trading Mode' in section 'X-Ref MMT-FIX'

– 'End of a FIX session started by the receiver' diagram should only have one arrow from Server to Client

– New table 'Level 3.9 - Algorithmic Indicator' in section 'X-Ref MMT-FIX'

– Market Data Snapshot Full Refresh message. New tag 2667 'AlgorithmicTradeIndicator'

– Market Data Snapshot Full Refresh message. Change the description of the E and F values of tag 269 [MDEntryType]

**Outlined below are the main changes from the documentation published on 12 January 2022:**

- Expand description of tag 864 in message Market Data Snapshot Full Refresh

- Market Data Snapshot Full Refresh message, tag273 must be present when tag269=6

- Security List Request message, tag263 must be present

**Outlined below are the main changes from the documentation published on 27 September 2022:**

- Adaptation of the document to the new corporate template

# Contents

# 1 Introduction

## 1.1 Scope of this manual

This document contains the definition of the MEFF trading system interface provided by MEFF for developing external applications. The interface is based on version 5.0 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 5.0 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 process market data using the HF MEFFGate server FIX interface.

## 1.2 Public information

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

| Public function | Related messages |
|---|---|
| Obtain session status | Trading Session Status Request |
| | Trading Session Status |
| Obtain information on securities | Security List Request |
| | Security List |
| | Security List Update Report |
| | Security Status Request |
| | Security Status |
| Obtain information on prices | Market Data Request |
| | Market Data Request Reject |
| | Market Data – Snapshot / Full Refresh |
| Obtain information about RFQ | Indication of Interest |
| Receive information from the Market Supervisor | News |

## 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 HF 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 5.0**. Details the annotations and adaptations that MEFF 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

## 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 5.0 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.<br>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) 5.0 Service Pack 2 (9 December 2013)<br>EP98-222 enhancing FIX 5.0 SP2 | FIX Committee |
| 2 | HF MEFFGate – FIX Interface Specifications T5.0 | MEFF |

# 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 this implementation are detailed here.

## 2.2 Fields ignored

In some cases, the content of certain fields of the entering messages may be ignored by HF 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 HF MEFFGate should not contain unsupported fields. Messages sent by HF 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 HF MEFFGate is 4096 bytes.

## 2.6 Encryption

HF 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 HF MEFFGate FIX protocol

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

It is important to distinguish between the version of the FIX protocol (in this case "5.0") and the version of the HF MEFFGate FIX protocol ("M5.13" in this edition).

More than one version of the HF MEFFGate FIX protocol may exist for the same version of FIX.

If the version requested by the client program is not available in the HF 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. HF MEFFGate implements the majority of the functions of the session level defined in the FIX 5.0 standard

## 3.2 FIX session and communication session

There are two types of session:

- **Communication session**. It begins when opening the socket (ip-address and port assigned to this service). It ends when the socket is closed.

- **FIX 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 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.

In addition to the two mentioned types of sessions, the trading session should also be considered. A trading session in an environment begins each day when the HF MEFFGate server loads the trading system data and accepts connections for said session.

The client program must begin a new FIX session in every communication session.

Given that HF 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, HF 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 trader. TargetCompID together with TargetSubID identify the environment.

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 HF MEFFGate, with respect to those sent by the client. Suppose that trader "001" of member "A001" has a session established with the Financial Contract Group at MEFF. The messages will be those shown below:

Client message to HF MEFFGate:

- o SenderCompID = "A001"
- o SenderSubID = "001"
- o TargetCompID = Operating MIC
- o TargetSubID = "M3" *

HF MEFFGate message to client:

- o SenderCompID = Operating MIC
- o SenderSubID = "M3"
- o TargetCompID = "A001"
- o TargetSubID = "001"

The list of values for TargerCompID/SenderCompID used by BME is located in codification table 2.

The list of values for TargetSubID/SenderSubID used by BME is located in codification table 1.

## 3.4 Client software and FIX sessions

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

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

A HF 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 contract group that is not available, his Logon message is answered with a Logout message with the appropriate explanation.

## 3.5 Start of the FIX session

On initiating a new communication session (opening a new socket), the client must initiate a new FIX session. The procedure to follow is described below.

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

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

## 3.6 Synchronisation at application level

When a client starts a FIX session (Logon message accepted), it receives a series of information related with the current Segment session.

To synchronise at the application level, the client may use the tags ApplID [1180] + ApplSeqNum [1181]. Value 0 in ApplID [1180] and ApplSeqNum [1181] means updates from the beginning of the business session. If this field is not specified, then the classical behaviour is assumed (snapshot of the current situation and updates from this time).

**It is recommended** the use of the user defined tag MoreSubscriptionsFollowing [21500], (**MoreSubscriptionsFollowing [21500] = "Y"**), in the subscription request. This way allows to group market information subscription requests (Trading Session Status Request, Security List Request, Security Status Request and Market Data Request) and to establish the moment in which the HF MEFFGate will begin to treat those requests. It can be combined with the reception of public information from a particular point in session to handle connections after a disconnection. Messages will be sent in the same order in which they were generated during the session; this implies that it is possible to receive Market Data Snapshot Full Refresh while receiving Security List Update Report.

When this tag is used, MoreSubscriptionsFollowing [21500] = "Y", HF MEFFGate will leave the different subscriptions requests pending and will not process them until a subscription request with MoreSubscriptionsFollowing [21500] = "N", is received. HF MEFFGate will assume that from this moment no subsequent subscription requests will be received and therefore any subsequent request will be rejected. Below is an example of how the tag MoreSubscriptionsFollowing [21500] is used.

If this tag is not used (or MoreSubscriptionsFollowing [21500] = "N" for all subscriptions), the current behaviour will be maintained, i.e. an immediate reply to each subscription request.

Below there is a message flow using the tag MoreSubscriptionsFollowing [21500]:
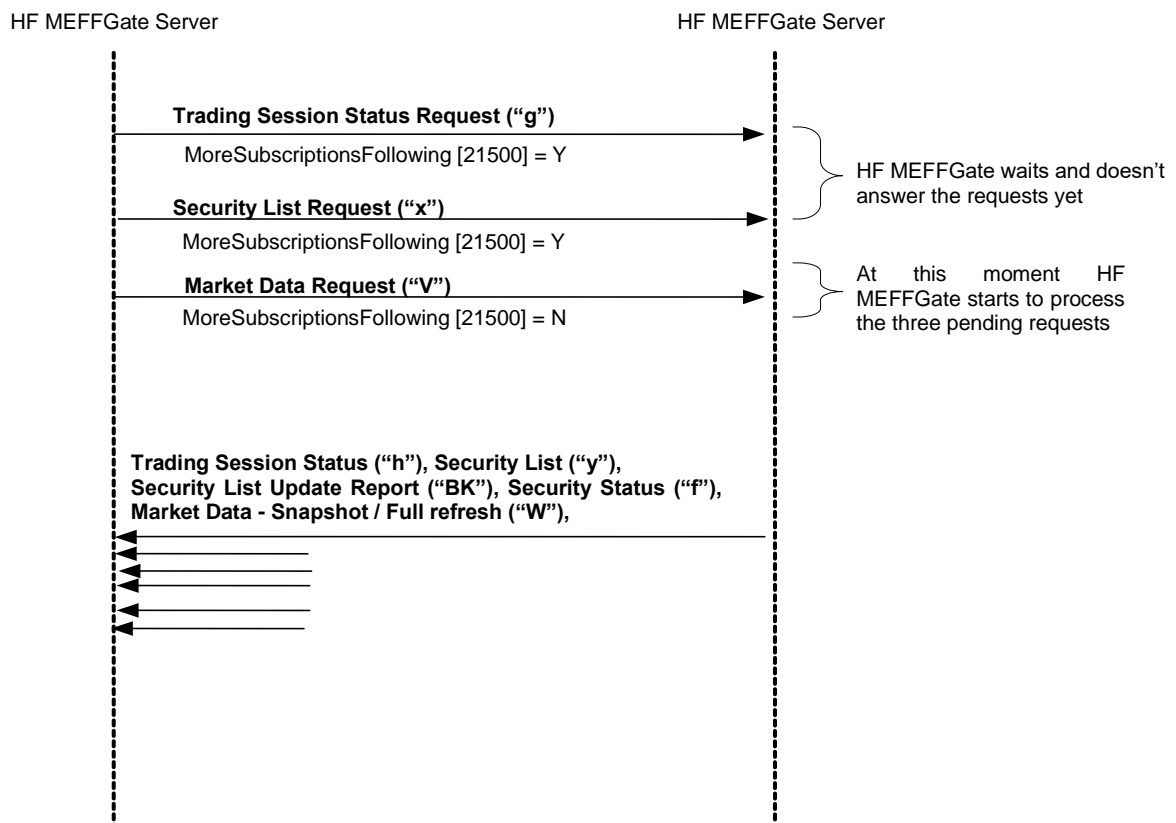
It should be taken into account that any subscription to information is cancelled when the FIX session ends. If this service is required when reconnecting to a new session, it must be requested again.

The series of messages not associated to subscriptions referred to in this section correspond to the following messages:
News

## 3.7 High availability

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

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

When a HF MEFFGate server fails, the client can continue working with another HF MEFFGate. The client must carry out the necessary processes to synchronise at the application level using the tags ApplID [1180] + ApplSeqNum [1181].

When a client application that has established a FIX session fails, the client application can synchronise at the application level from another equipment following the same procedure described in the previous paragraph.

## 3.8 List of messages

The functionality at the session level is implemented in FIX 5.0 using five administrative messages. All these are fully supported by the HF MEFFGate FIX protocol.

| Message | Description |
| --- | --- |
| Logon (Msg Type = A) | Request or confirmation of the start of a FIX session |
| Logout (Msg Type = 5) | Request or confirmation of the end of a FIX session |
| Heartbeat (Msg Type = 0) | Periodic notification that the connection is alive |
| Test Request (Msg Type = 1) | Request to send a Heartbeat message to confirm that the connection is alive |
| Reject (Msg Type = 3) | Reject a message at session level |

## 3.9 Message flow

### Start of FIX session

A request to start a FIX 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.

HF MEFFGate Client                                    HF MEFFGate Server

**Logon ("A")**
MsgSeqNum [34] = 1

**Logon ("A")**
HF MEFF Gate Client
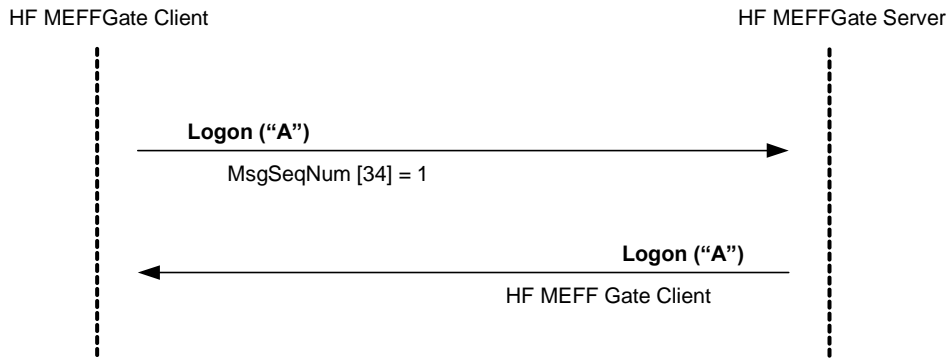
### Start of FIX session rejected

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

For more details on the behaviour of sequence numbers of both parties see section 0.

HF MEFFGate Client                                    HF MEFFGate Server

**Logon ("A")**
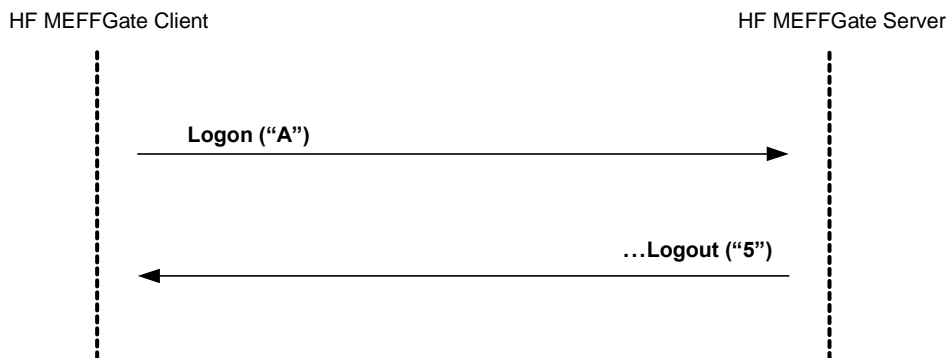
**…Logout ("5")**

### End of a FIX session started by the sender

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

HF MEFFGate Client                                    HF MEFFGate Server
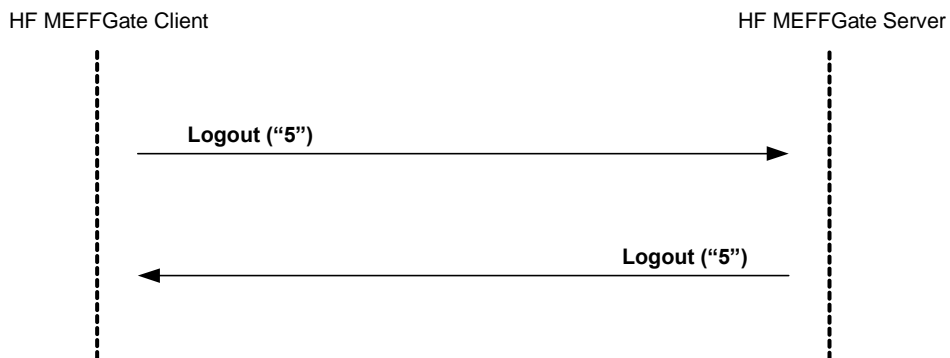
**Logout ("5")**

**Logout ("5")**

## End of a FIX session started by the receiver

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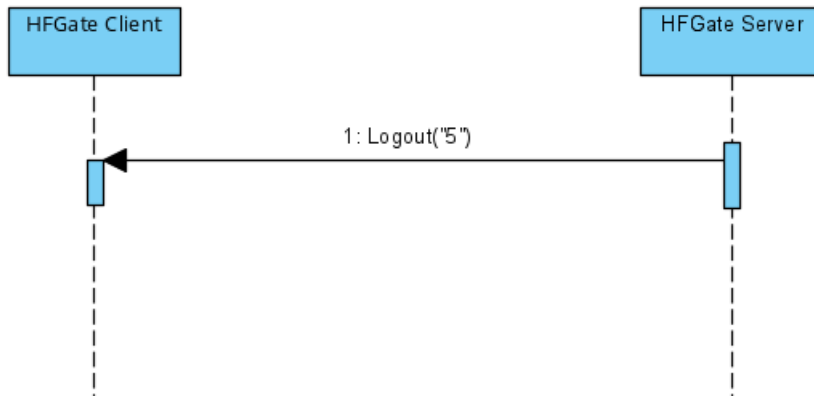


## Sending messages with identification fields of session (SenderCompID, SenderSubID, TargetCompID and TargetSubID) with different values from those associated to the current FIX session

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.
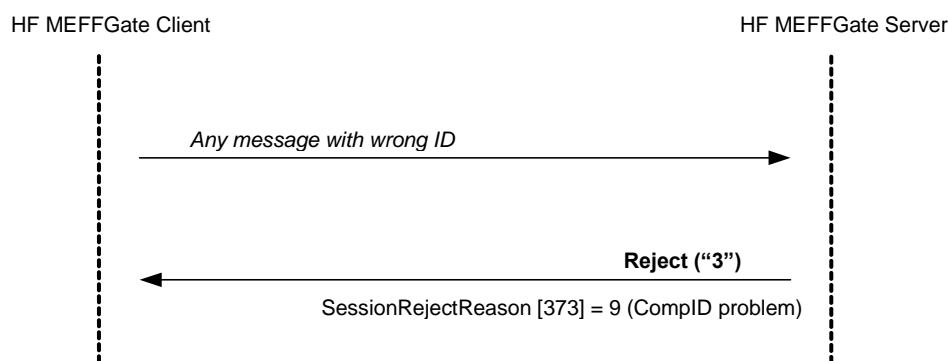


## 3.10 Annotations and adaptations of FIX 5.0

The user optional field LocalMktTimestamp [21501] has been added to the Logon message to Indicates for all tags in which a timestamp is included, the timestamp format (UTC format or local market time)

The optional fields ApplID [1180] and ApplSeqNum [1181] have been added to the Logon message to indicate that only updates from the point indicated are requested

The Text [58] and DefaultCstmApplVerID [1408] fields in the Logon message are now required

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

The SenderSubID [50] and TargetSubID [57] fields in the header of messages (Standard Message Header) are now required

The FIX   method of encryption is not supported

The Resend Request and Sequence Reset messages are not supported (and rejected by HF MEFFGate)

The valid values of the ResetSeqNumFlag [141] field in the Logon message are limited to the value "N"

The user field BusinessSessionDate [21505] has been added to the Logon message sent by HF MEFFGate to inform the current business session date

## 3.11 Definition of messages

### 3.11.1 Standard Message Header

Header is present in all FIX messages.

| Tag | Name | Req | Valid values | Format | Description |
|---|---|---|---|---|---|
| 8 | BeginString | Y | FIXT.1.1 | String | Indicates the start of a new message. 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 | See chapter "3.3 - Identification of the FIX session" | String | Identifier of the entity that sends the message.<br><br>It contains the operating MIC of the venue (see table 2 document "Codification tables") when the message is sent by HF MEFFGate.<br><br>It must contain the member code in the messages sent by the client application. |
| 56 | TargetCompID | Y | See chapter "3.3 - Identification of the FIX session" | String | Identifier of the entity that the message is sent to.<br><br>It should contain the operating MIC of the venue (see table 2 document "Codification tables") when the message is sent to HF MEFFGate, although HF MEFFGate ignores the content of this field.<br><br>It contains the member code in the messages sent by HF MEFFGate. |
| 34 | MsgSeqNum | Y | | SeqNum | Sequence number of the message within the current FIX session |
| 50 | SenderSubID | Y* | See chapter "3.3 - Identification of the FIX session" | String | The messages sent from HF MEFFGate to the client contain the code assigned to the contract group with which the connection was established (see table 1 document "Codification tables"). |

| Tag | Name | Req | Valid values | Format | Description |
|---|---|---|---|---|---|
| | | | | | Messages sent to HF MEFFGate must contain the trader code with which the FIX session was started |
| 57 | TargetSubID | Y* | See chapter "3.3 - Identification of the FIX session" | String | The messages sent from HF MEFFGate contain the code of the trader which it is to be sent to.<br><br>Messages sent to HF MEFFGate must contain the code of the contract group with which the connection was established (see table 1 document "Codification tables") |
| 52 | SendingTime | Y | | UTC Timestamp | Time message sent |

### 3.11.2 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.<br>It is always the last field of the message and its length is exactly 3 bytes |

### 3.11.3 Logon (Msg Type = A)

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

| Tag | Name | Req | Valid values | Format | Description |
|-----|------|-----|--------------|--------|-------------|
| | Standard Header | Y | MsgType = A | | |
| 98 | EncryptMethod | Y | 0 = None | Int | Ignored by HF MEFFGate |
| 108 | HeartBtInt | Y | >= 1 | 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 | | SeqNum | If informed only value 1 is allowed |
| 464 | TestMessageIndicator | N | Y = Test<br>N = Production | Boolean | Indicates whether it is a test or production session.<br>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 HF MEFFGate<br>If the client does not indicate anything, this parameter is not taken into account.<br>In any event HF MEFFGate always informs this field |
| 553 | Username | N | | String | Identifier of the user assigned by MEFF. Required when the message is sent by the client application.<br>It is currently comprised of the combination of the member code and the trader code assigned by MEFF |
| 554 | Password | N | | String | User Password. Required when the message is sent by the client application |
| 1137 | DefaultApplVerID | Y | 9 | String | Value 9 refers to FIX50SP2 |
| 1408 | DefaultCstmApplVerID | Y* | M5.13 | String | Exact identification of the version of the protocol used and expected by the client application |
| 58 | Text | Y* | | String | The client must include a descriptive string of the software name used by the FIX connection. This will be one that has passed the corresponding conformance test |

| Tag | Name | Req | Valid values | Format | Description |
|-----|------|-----|--------------|--------|-------------|
| 1180* | ApplID | N | | String | If provided, only updates from the point indicated will be sent. This value, used in conjunction with ApplSeqNum [1181], should match in the same field in any of the messages provided by the HF MEFFGate such as: Market Data Snapshot Full Refresh, Security List, Security List Update Report, Security Status, ... |
| 1181* | ApplSeqNum | N | | SeqNum | Required if ApplID [1180] is specified.<br><br>This value, used in conjunction with ApplID [1180], should match the same field in any of the messages provided by the HF MEFFGate such as: Market Data Snapshot Full Refresh, Security List, Security List Update Report, Security Status, ... |
| 21501* | LocalMktTimestamp | N | Y, N (default) | String | Indicates, for all tags in which a timestamp is included, the timestamp format:<br><br>Y – HF MEFFGate will send the local market time (all messages up to microseconds)<br><br>N – HF MEFFGate will send the the time in UTC format according to the FIX standard (all messages up to microseconds)<br><br>For more information see 4.5 |
| 21505* | BusinessSessionDate | N | | LocalMktDate | Current business session date.<br><br>This tag is only informed in the Logon response message sent by HF MEFFGate. The client application should not send this tag in the Logon message sent to HF MEFFGate. |
| | Standard Trailer | Y | | | |

### 3.11.4  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 | | | |

### 3.11.5 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 | | | |

### 3.11.6  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 | | | |

### 3.11.7 Reject (Msg Type = 3)

The Reject message is used by HF 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 | | SeqNum | Sequence number of the rejected message |
| 373 | SessionRejectReason | N | 0 = Invalid tag number<br>1 = Required tag missing<br>2 = Tag not defined for this message type<br>3 = Undefined Tag<br>4 = Tag specified without a value<br>5 = Value is incorrect (out of range) for this tag<br>6 = Incorrect data format for value<br>9 = CompID problem<br>11 = Invalid MsgType<br>13 = Tag appears more than once<br>14 = Tag specified out of required order<br>15 = Repeating group fields out of order<br>16 = Incorrect NumInGroup count for repeating group<br>17 = Non "data" value includes field delimiter (SOH character)<br>99 = Other | Int | Code indicating the rejection motive |

| Tag | Name | Req | Valid values | Format | Description |
|-----|------|-----|--------------|--------|-------------|
| 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 Instrument block

In some requests, the FIX client may specify selection criteria for the securities. In these cases, it will only receive information on the securities 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 |
|---|---|
| SecurityType [167] | Product type |
| SecurityID [48] | MEFF Underlying asset |
| MaturityMonthYear [200] | Contract expiration |

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

### 4.1.1 SecurityType [167]

This code identifies the product type (see table 6 in document "Codification Tables"). Only messages sent by HF MEFFGate. Not allowed in messages sent by FIX client.

### 4.1.2 Underlying asset (SecurityID [48] field)

This code identifies the underlying asset of a contract (see table 7 in document "Codification Tables")

### 4.1.3 Expiration (MaturityMonthYear [200] field)

For contracts with standard maturities, indicates the month and year when the contract expires. In this case, the format for this field is YYYYMM (e.g. 201312)

For contracts with non-standard maturities, indicates the date when the contract expires. In this case, the format for this field is YYYYMMDD (e.g. 20131219)

For contracts with week standard maturities, the format for this field is YYYYMMwW (e.g. 201312w2).

### 4.1.4 Combination of selection criteria

When various selection criteria are combined, only those securities that meet all the requirements are selected. When a selection criteria is not specified it is understood that this criteria is to be ignored and no security will be discarded for this reason.

The following table shows some examples:

| SecurityType [167] | SecurityID [48] | MaturityMonthYear [200] | Meaning |
|---|---|---|---|
| F | FIE | (omitted) | All futures on IBEX index |
| F | BBVA | (omitted) | All the BBVA futures contracts with physical delivery |
| (omitted) | FIE | 201203 | All the contracts with IBEX index as underlying, with March 2012 expiration |
| O | (omitted) | 201206 | All options with June 2012 expiration |
| R | TEF | (omitted) | All time-spread contracts where Telefonica stocks is underlying of at least one leg |
| (omitted) | (omitted) | (omitted) | All contracts |
| X | (any) | (any) | **Wrong selection criteria** |

## 4.2 Limitation on the maximum permitted number of subscriptions

Only one subscription per subscription type during the session is allowed, except for Market Data where up to 5 subscriptions are supported.

If, once reached that limit, the client application tries to establish new subscriptions, they will be rejected with an error message indicating that the maximum permitted number of subscriptions has been reached.

## 4.3 Reception of public information from a particular point in the session

It is possible to receive only the updates from a particular point in the business session indicated by the client application. For this purpose tags ApplID [1180] and ApplSeqNum [1181] are used in the Logon message. If this field is not specified then the classical behaviour is assumed (snapshot of the current situation and updates from this time for Market Data Snapshot Full Refresh and updates from the beginning of the business session for Trading Session Status, Security List and Security Status).

Value 0 in ApplSeqNum [1181] means updates from the beginning of the business session.

## 4.4 Fields not informed on receipt of information

If no information is received from HF MEFFGate for an specific field, then it should be considered that this has not changed since last update.

This applies, for example, to Market Data Snapshot Full Refresh and Security Status messages.

To see it better, let's consider two examples:

**Example 1**: Market Data request.

Initial market. This is the bid-offer situation for an specific contract:

| Bid | Offer |
|---|---|
| | 10 @ 9015 |
| 2 @ 9014 | |
| 6 @ 9012 | |

The following Market Data Snapshot Full Refresh message is sent:

...

NoMDEntries [268] = 3

MDEntryType [269] = 0 (Bid)

    MDEntryPx [270] = 9014

    MDEntrySize [271] = 2, ...

MDEntryType [269] = 0 (Bid)

    MDEntryPx [270] = 9012

    MDEntrySize [271] = 6, ...

MDEntryType [269] = 1 (Offer)

    MDEntryPx [270] = 9015

    MDEntrySize [271] = 10, ...

...

A new change on the bid side (price 9012 is deleted):

| Bid | Offer |
|---|---|
| | 10 @ 9015 |
| 2 @ 9014 | |

HF MEFFGate communicates this update **without necessarily informing** the offer side:

...

NoMDEntries [268] = **1**

MDEntryType [269] = 0 (Bid)

    MDEntryPx [270] = 9014

    MDEntrySize [271] = 2, ...

...

Another new change on the bid side (last price 9014 is deleted):

| Bid | Offer |
|---|---|
| - | 10 @ 9015 |

HF MEFFGate communicates this update **without necessarily informing** the offer side:

…

NoMDEntries [268] = **1**

MDEntryType [269] = 0 (Bid)

MDEntryPx [270] = …

MDEntrySize [271] = 0, …

…

**Example 2**: Security Status request.

Let's assume a contract, ready to trade, with an specific price range. This is the Security Status message sent:

…

SecurityTradingStatus [326] = 17 (Ready to trade)

HighPx [332] = 9075

LowPx [333] = 8975

…

The security status changes to auction:

HF MEFFGate communicates this update **without necessarily informing** the price range (HighPx, LowPx):

…

SecurityTradingStatus [326] = 21 (Pre-Open)

…

## 4.5 Timestamp format

The system permits the user to define, for all tags in which a timestamp is included, whether the format is UTC (according to the FIX standard), or the local market time.

For this functionality the user defined tag LocalMktTimestamp [21501] is used in the Logon message.

When this tag is used, with LocalMktTimestamp [21501] = "Y", HF MEFFGate will send the local market time (all messages up to microseconds).

If this tag is not used (or LocalMktTimestamp [21501] = "N"), HF MEFFGate will send the time in UTC format (all messages up to microseconds).

## 4.6 SeqNum data types

According to the FIX standard, the SeqNum data type is an int field and value must be positive. The client application must be ready to receive values greater than $2^{31}$.

## 4.7 X-Ref MMT-FIX

**Level 1 - Market Mechanism**

**MARKET MODEL TIPOLOGY**          **FIX (Market Data Snapshot Full Refresh message)**

| Full Name | Code (Efficient Mode) | MdOriginType [1024] |
|---|---|---|
| Central Limit Order Book | 1 | 0 |
| Quote Driven Market | 2 | 3 |
| Dark Order Book | 3 | 4 |
| Off Book (including Voice or Messaging Trading) | 4 | 1 |
| Periodic Auction | 5 | 5 |
| Request for quotes | 6 | 6 |

**Level 2 - Trading Mode**

**MARKET MODEL TIPOLOGY**          **FIX (Trading Session Status message)**

| Full Name | Code (Efficient Mode) | TradingSessionSubId [625] |
|---|---|---|
| Undefined Auction | 1 | 8 |
| Scheduled Opening Auction | O | 2 |
| Scheduled Closing Auction | K | 4 |
| Scheduled Intraday Auction | I | 6 |
| Unscheduled Auction | U | 9 |
| Continuous Trading | 2 | 3 |
| At Market Close Trading | 3 | 5 |
| Out of Main Session Trading | 4 | 10 |
| Trade Reporting (On Exchange) | 5 | 201 |
| Trade Reporting (On Exchange) | 5 | 203 |
| Trade Reporting (On Exchange) | 5 | 204 |

## Level 3.1 - Transaction category

MARKET MODEL TIPOLOGY                    FIX (Market Data Snapshot Full Refresh message)

| Full Name | Code (Efficient Mode) | EventText [868] con EventType [865] = 211 |
|---|---|---|
| Package Trade (excluding Exchange for Physicals) "TPAC" | Z | Z |
| Exchange for Physicals Trade "XFPH" | Y | Y |

## Level 3.5 - Benchmark or Reference Price Indicator

MARKET MODEL TIPOLOGY            FIX (Market Data Snapshot Full Refresh message)

| Full Name | Code (Efficient Mode) | TradeCondition [277] | TrdRegPublicationType [2669] | TrdRegPublicationReason [2670] |
|---|---|---|---|---|
| Benchmark Trade "BENC" | B | 6 | - | - |

## Level 3.8 - Ordinary/Standard Trades or Trades Outside Price Formation / Discovery Process

MARKET MODEL TIPOLOGY                    FIX (Market Data Snapshot Full Refresh message)

| Full Name | Code (Efficient Mode) | TradePriceCondition [1839] |
|---|---|---|
| Plain-Vanilla Trade | P | - |
| Non-Price Forming Trade (formerly defined as a Technical Trade) "NPFT" | T | 15 |

## Level 3.9 - Algorithmic Indicator

**MARKET MODEL TIPOLOGY**              **FIX (Market Data Snapshot Full Refresh message)**

| Full Name | Code (Efficient Mode) | AlgorithmicTradeIndicator [2667] |
|---|---|---|
| Algorithmic trade "ALGO" | H | 1 |
| No Algorithmic trade | - | 0 |

## Level 4.1 - Publication Mode / Post-Trade Deferral Reason

**MARKET MODEL TIPOLOGY**        **FIX (Market Data Snapshot Full Refresh message)**

| Full Name | Code (Efficient Mode) | TradePublishIndicator [1390] | TrdRegPublicationType [2669] | TrdRegPublicationReason [2670] |
|---|---|---|---|---|
| Immediate Publication | - | 1 | - | - |
| Non-Immediate Publication | 1 | 2 | - | - |
| Non-Immediate Publication: Deferral for "Large in Scale" "LRGS" | 2 | 2 | 1 | 6 |
| Non-Immediate Publication: Deferral for "Illiquid Instrument" (RTS 2 only) "ILQD" | 3 | 2 | 1 | 7 |
| Non-Immediate Publication: Deferral for "Size Specific" (RTS 2 only) "SIZE" | 4 | 2 | 1 | 8 |

## Level 4.2 - Post-Trade deferral or Enrichment

**MARKET MODEL TIPOLOGY**                    **FIX (Market Data Snapshot Full Refresh message)**

| Full Name | Code (Efficient Mode) | RegulatoryReportType [1934] |
|---|---|---|
| Limited details trade "LMTF" | 1 | 11 |
| Daily aggregated trade "DATF" | 2 | 12 |
| Volume omission trade "VOLO" | 3 | 13 |
| Four weeks aggregation trade "FWAF" | 4 | 14 |
| Indefinite aggregation trade "IDAF" | 5 | 15 |
| Volume omission trade. Eligible for subsequent enrichment in aggregated form "VOLW" | 6 | 16 |
| Full details of previous LMTF "FULF" | 7 | 17 |
| Full details of previous DATF "FULA" | 8 | 18 |
| Full details of previous VOLO "FULV" | 9 | 19 |
| Full details of previous FWAF "FULJ" | V | 20 |
| Full details of previous VOLW "COAJ" | W | 21 |

# 5 Common Application Messages

## 5.1 Introduction

This chapter presents some common messages at the application level that cover three functions: the control of the communication status, the individual user password change and the rejection of messages by HF MEFFGate.

## 5.2 Network communication status

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

HF MEFFGate will always send Network Counterparty System Response messages reporting on status of connection between HF MEFFGate and the central systems (whether or not the client subscribed to it).

The information supplied with these messages only refers to the connection between the equipment and should not be confused with the status of the trading session, which is covered in 6.2.

## 5.3 Password change

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

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

## 5.4 Rejection of application messages

When HF 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 Network Counterparty System Status Request message.

## 5.5 List of messages

| Message | Description |
|---------|-------------|
| Network Counterparty System Status Request (Msg Type = BC) | Request of connection status between HF MEFFGate and the central systems |
| Network Counterparty System Status Response (Msg Type = BD) | Report on status of connection between HF 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

**Subscription to connection status**

| HF MEFFGate Client | | HF MEFFGate Server |
|---|---|---|

**Network CounterParty System Status Request ("BC")** →

NetworkRequestType [935] = 2 (Subscribe)

← **Network CounterParty System Status Response ("BD")**

←

←

…
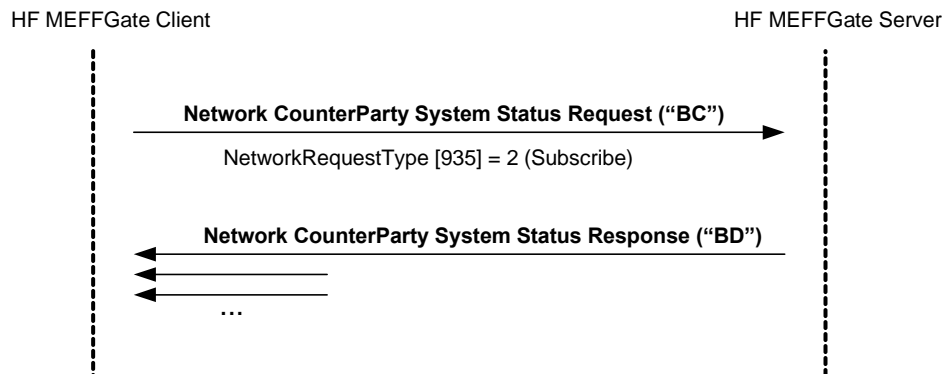
**Report on connection status without any subscription**

| HF MEFFGate Client | | HF MEFFGate Server |
|---|---|---|

← **Network CounterParty System Status Response ("BD")**

←

←

…
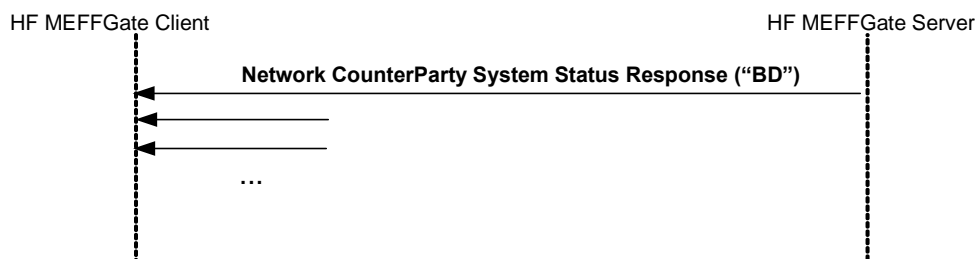
**Individual password change**

| HF MEFFGate Client | | HF MEFFGate Server |
|---|---|---|

**User Request ("BE")** →

← **User Response ("BF")**
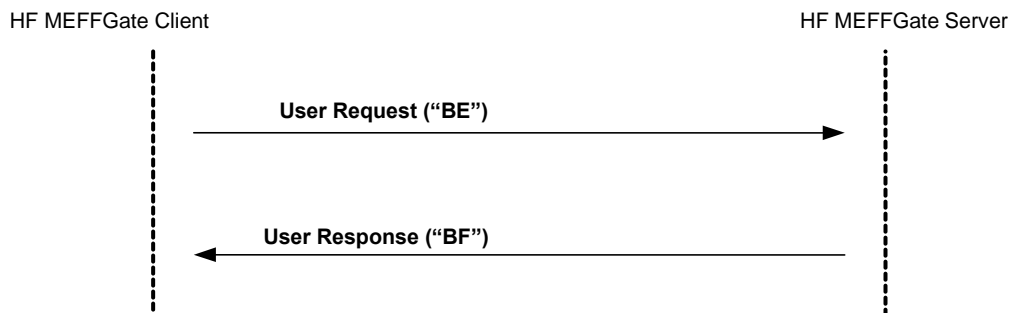
## 5.7 Annotations and adaptations of FIX 5.0

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 HF MEFFGate and the MEFF central systems.

| Tag | Name | Req | Valid values | Format | Description |
|-----|------|-----|--------------|--------|-------------|
| | Standard Header | Y | MsgType = BC | | |
| 935 | NetworkRequestType | Y | 2 = Subscribe | 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 HF MEFFGate as reply to a Network Counterparty System Status Request Message.

It has information about the connectivity between HF 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 | N | | String | Message identifier Network Counterparty System Status Request to which it is being responded |
| 932 | NetworkResponseID | Y | | String | Unique message identifier |
| 936 | NoCompIDs | Y | 1 | NumInGroup | |
| → 930 | RefCompID | Y | | 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 | See Identification of the FIX session | 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 | Y | 1 = Connected 2 = Not connected – down expected up 3 = Not connected – down expected down 4 = In Process | Int | Connection status This field is always included in the message |
| → 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 HF 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 |
| 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 HF 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<br><br>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 HF 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 a Network Counterparty System Status Request message.

| Tag | Name | Req | Valid values | Format | Description |
|-----|------|-----|--------------|--------|-------------|
| | Standard Header | Y | MsgType = j | | |
| 45 | RefSeqNum | N | | SeqNum | When present, indicates MsgSeqNum of the rejected message.<br><br>If value zero, the content of this field should not be considered. |
| 372 | RefMsgType | Y | . | String | MsgType of the rejected message |
| 379 | BusinessRejectRefID | N | | String | Optional Identifier of the rejected message |
| 380 | BusinessRejectReason | Y | 0 = Other<br>3 = Unsupported Message Type | Int | Reason for rejection |
| 58 | Text | N | | String | Explanation of rejection |
| | Standard Trailer | Y | | | |

# 6  Market Information

## 6.1 Introduction

Market information groups together various functionalities related to public market information, which are classified into three groups:

- **Session status.** Status of trading session

- **Security information**. Definition and status of securities selected

- **Prices**. Prices in selected securities

Each of these groups is covered in a separate section of this chapter. Section 6.5 provides details of the format of the corresponding messages.

## 6.2 Market information: Session status

### 6.2.1 Description

This functionality allows the client to obtain the status for the contract group associated to the current FIX session and to be notified of the changes of status that occur.

### 6.2.2 List of messages

| Message | Description |
|---|---|
| Trading Session Status Request (Msg Type = g) | Sent by the client to request the session status |
| Trading Session Status (Msg Type = h) | Sent by the server to return information on the session status or to notify that the request has been rejected |

### 6.2.3 Message flow

**Trading Session status request**

This request is initially answered, for every trading mode within the corresponding contract group that meets the selection criteria, with one or more Trading Session Status messages indicating the different situations up to this moment.

From this point on, a new Trading Session Status message is received, when there is a change in status, with the corresponding information. These later messages will have "Y" in the UnsolicitedIndicator field.
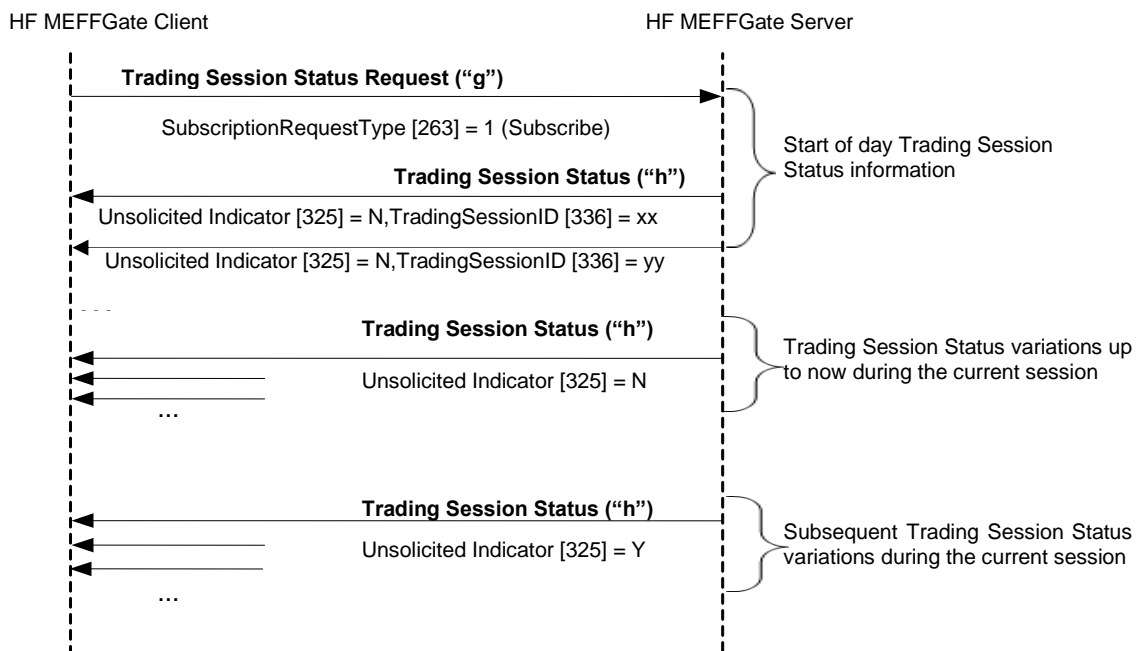


**Failed Trading Session status request**

A failed Trading Session Status request is answered by a Trading Session Status message with the field TradeSesStatus = 6.
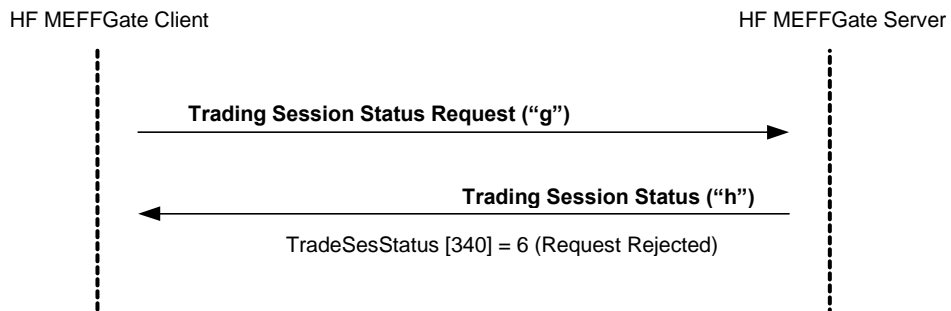
### 6.2.4 Annotations and adaptations of FIX 5.0

The tag TransactTime [60] has been added to message Trading Session Status

The optional field MoreSubscriptionsFollowing [21500] has been added to the Trading Session Status Request message to group market information subscription requests

## 6.3 Market information: Securities

### 6.3.1 Description

This functionality allows security information to be obtained. The information is organised in two groups:

- **Security definitions**. Static information of the definition of the securities as a snapshot (Security List) and the updates during the session (Security List Update Report).

- **Security status**. Dynamic information that shows the status of the securities (Security Status)

### 6.3.2 Request security information

The request for the definition of securities follows the criteria specified in the section Instrument block on this document.

### 6.3.3 Reception of security definitions

The information on the security definitions is received in the Security List and Security List Update Report messages. Keep in mind that, according to the FIX standard, the Security List Update Report message is automatically sent (as a result of the subscription via Security List Request or Security Status Request) each time there is an update, during the session, to the security definition. As a result, the client application should be ready to receive this kind of message.

For instance, it's possible to receive a Security List Update Report message saying there are new strikes in options.

These messages gives one security at a time. The TotNoRelatedSym field gives the total number of securities that meet the selection criteria and the NoRelatedSym field (always 1) gives the number of securities contained in that particular message.

### 6.3.4 Reception of security status

The information of the security status is received in the Security Status message. Each Security Status message contains information for one security. The reply to a Security Status Request message may consist of several Security Status messages. In this case, there is no mechanism to know when all the information has been received. If necessary, the FIX client will have to first request the list of securities using the Security List Request message to work out how many securities meet certain criteria.

If no information is received for a specific field then it should be considered that this has not changed since the last update (see chapter 4.4 for more detail).

A new Security Status message is received when there is a change in the security status with the new information.

A new Security List Update Report message is received when there is a change in the security definition with the new information.

### 6.3.5 List of messages

| Message | Description |
| --- | --- |
| Security List Request (Msg Type = x) | Sent by the client to request the definition of securities. It also allows information on the status of the securities to be requested |
| Security List (Msg Type = y) | Sent by the server to provide the security definitions as a snapshot. It is also used to inform about the rejection of requests for this information |
| Security List Update Report (Msg Type = BK) | Sent by the server to provide the security definitions as an update during the session. |
| Security Status Request (MsgType = e) | Sent by the client to request the status of securities |
| Security Status (MsgType = f) | Sent by the server to inform about the status of securities. It is also used to inform about the rejection of requests for this information, or to inform that there is no security meeting the selection criteria |

### 6.3.6 Message flow

**Request security definitions and security status**

This request is initially answered, for each security that meets the selection criteria, with one or more Security List + Security Status messages indicating the different situations of the security up to this moment.

From this point on, a new Security List Update Report or Security Status message is received when there is a change in status for any of the set of the securities with the corresponding information. These later messages will have "Y" in the UnsolicitedIndicator field.
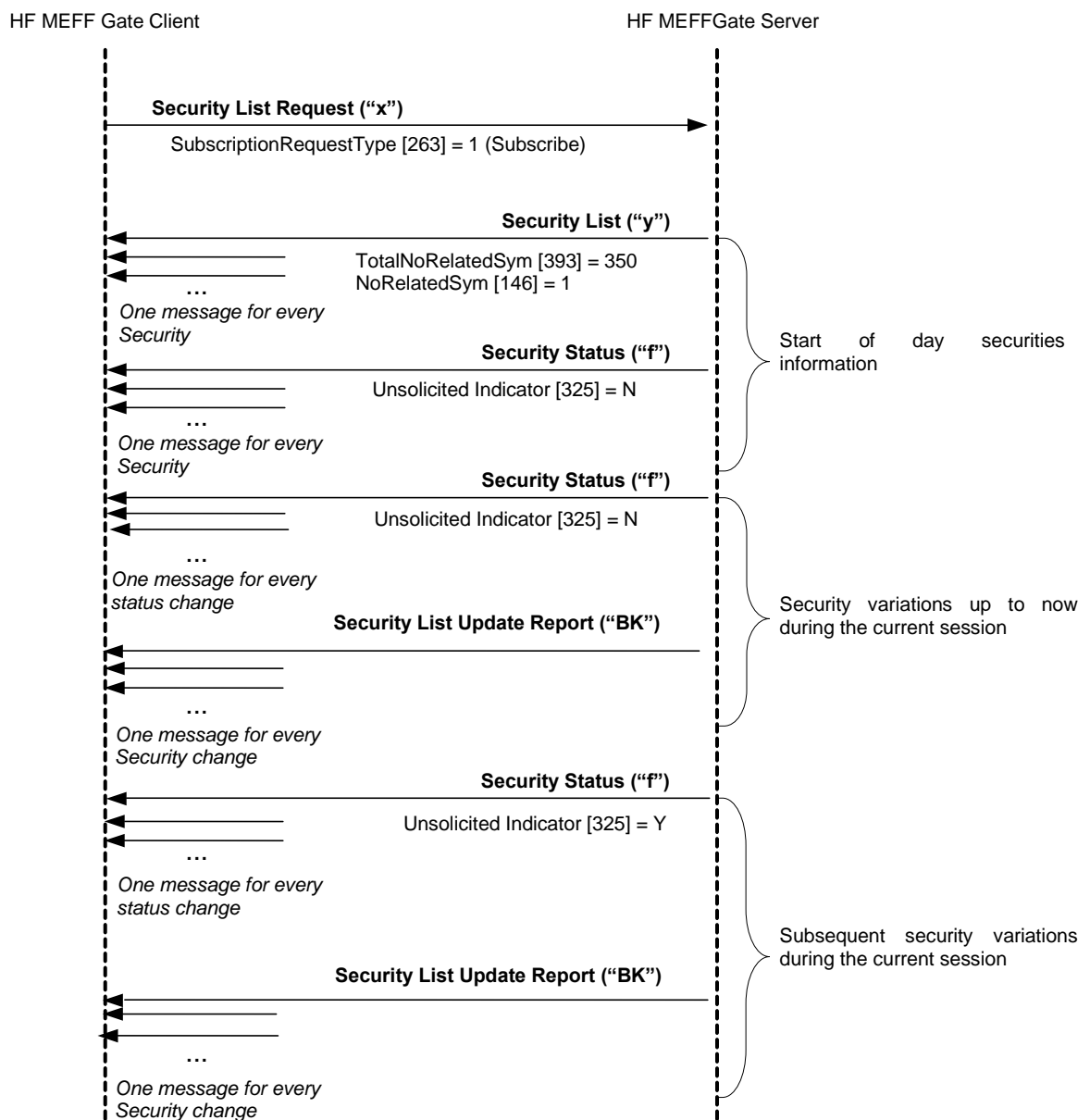
## Request security status

This request is initially answered, for each security that meets the selection criteria, with one or more Security Status messages indicating the different situations of the security up to this moment.

From this point on, a new Security List Update Report or Security Status message is received when there is a change in status for any of the set of the securities with the corresponding information.
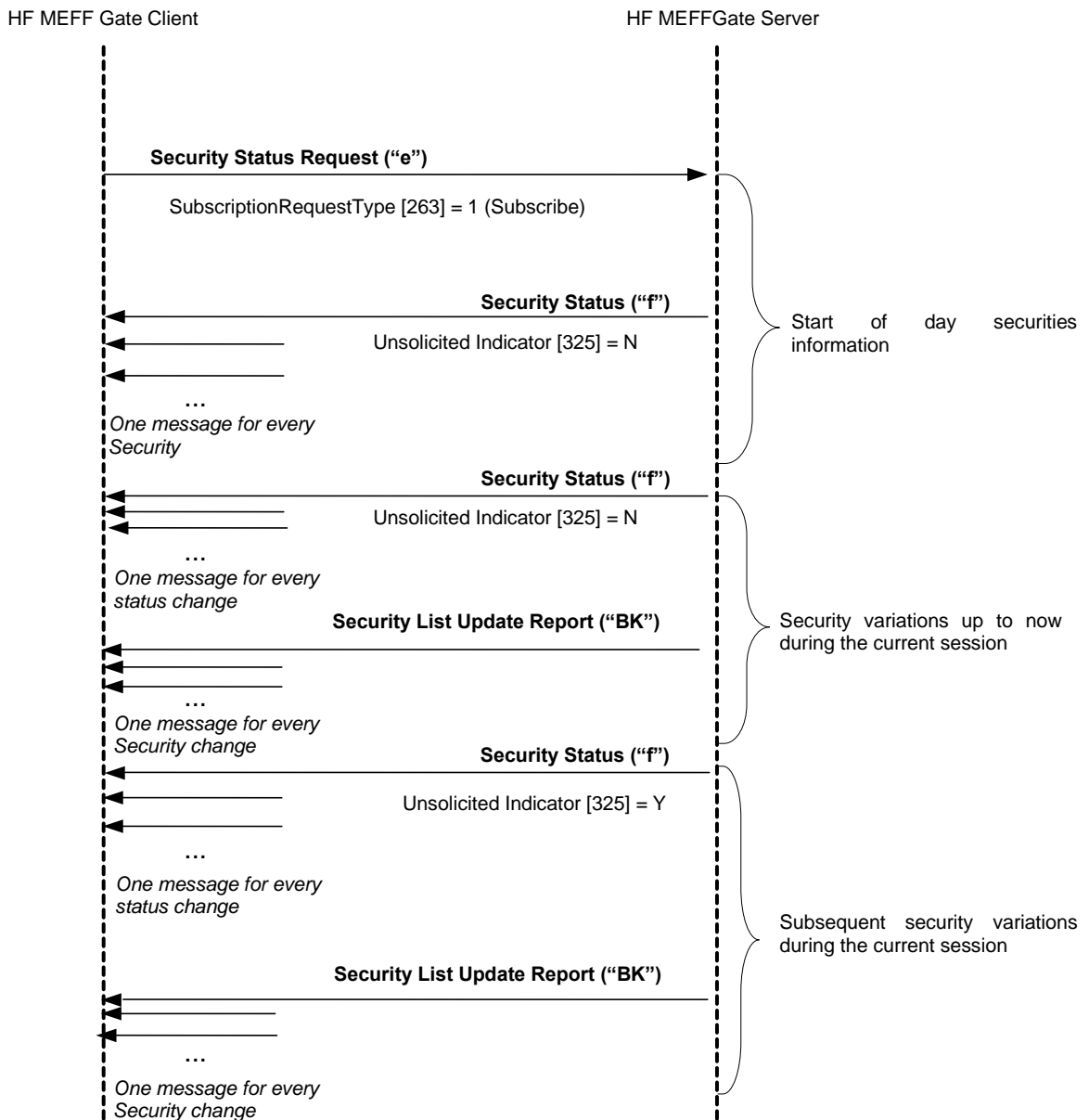
## Request security definitions, without securities that meet the selection criteria

When there are no securities that meet the selection criteria indicated in the security definition request, HF MEFFGate will reply with a Security List message where the field SecurityRequestResult = 2.

HF MEFFGate Client                               HF MEFFGate Server

**Security List Request ("x")** →

← **Security List ("y")**
SecurityRequestResult [560] = 2 (No instruments found that match selection criteria)
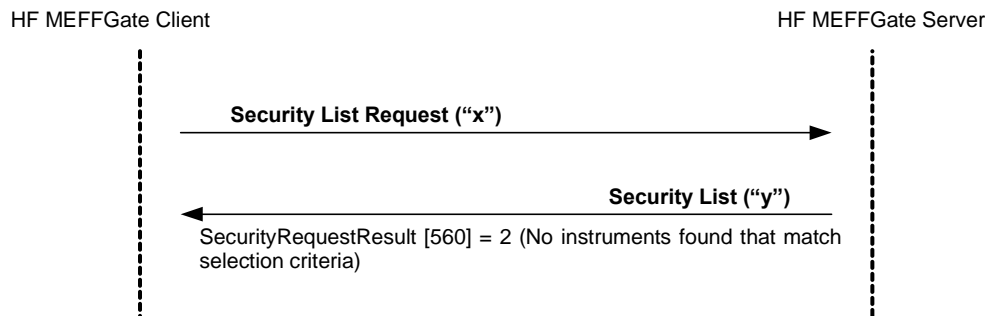
## Request security status, without securities that meet the selection criteria

When there are no securities that meet the selection criteria indicated in a security status request, HF MEFFGate replies with a SecurityStatus message where the field SecurityTradingStatus = 19.

HF MEFFGate Client                               HF MEFFGate Server

**Security Status Request ("e")** →

← **Security Status ("f")**
SecurityTradingStatus [326] = 19 (Not Traded on this segment)
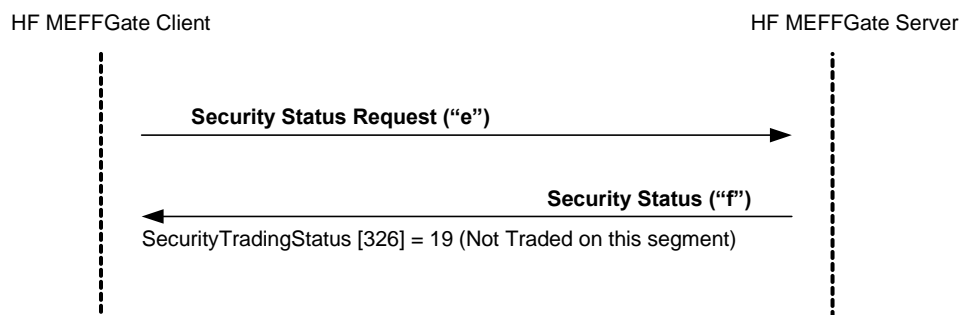
## Failed security definition request

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

HF MEFFGate Client                               HF MEFFGate Server

**Security List Request ("x")** →

← **Security List ("y")**
SecurityRequestResult [560] = 1 (Invalid or unsupported request)
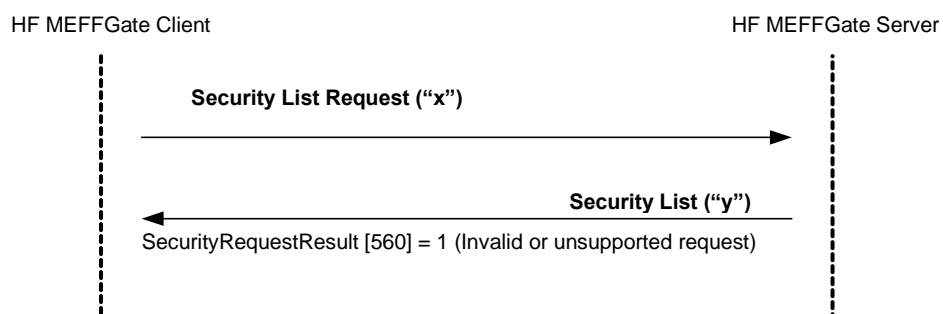
## Failed security status request

When a security status request is erroneous it is answered with a Security Status message where the field SecurityTradingStatus = 20.

HF MEFFGate Client            HF MEFFGate Server

**Security Status Request ("e")**
→

**Security Status ("f")**
←
SecurityTradingStatus [326] = 20 (Unknown or Invalid)
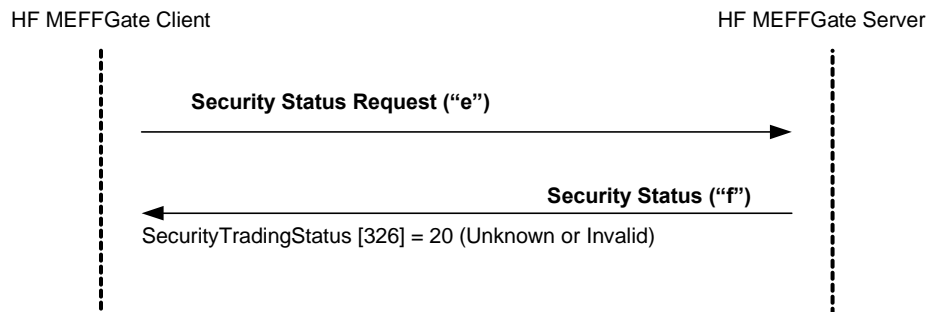
### 6.3.7   Annotations and adaptations of FIX 5.0

In the Security List and Security List Update Report messages the field EventType [865] with codes greater than 100 is used. The client application should be prepared to manage this situation in a correct way

The maximum number of subscriptions is limited (see section 4.2 for details)

The optional field MoreSubscriptionsFollowing [21500] has been added to the Security List Request y Security Status Request message to group market information subscription requests

## 6.4 Market information: Prices

### 6.4.1 Description

This functionality allows to request information on the prices for a number of securities.

### 6.4.2 Information request

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

A number of securities can be selected using a combination of fields of the Instrument block as explained in 4.1.

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.

Bid

Offer

Last Price

Opening Price (includes auction prices)

Settlement Price

Session High

Session Low

Session VWAP Price

Trade Volume

Open Interest at the end of the previous session

Prior settlement price

When a request includes Bid or Offer, it is possible to specify the depth in three modes: maximum, best prices or an exact depth.

In addition to the information listed here, the Bid or Offer request implies receiving RFQ for the contracts selected (See chapter "7 - RFQ" for a detailed explanation).

### 6.4.3 Receipt of information

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

In accordance with the FIX standard, messages in reply to the same request will not mix the Bid and Offer information with other information.

In the event that the request combines Bid or Offer information with other information, the reply will consist of two Market Data Snapshot Full Refresh messages per security.

A new Market Data Snapshot Full Refresh message will be received every time there is a change. For all fields, including bid and offer, if no information is received for an specific field then should be considered that this has not changed since the last update (see chapter 4.4 for more detail). Anyway, the restriction of not mixing Bid or Offer information with other fields is maintained.

Keep in mind that when there are no Bid or Offer prices for a security, this is notified by the value zero in the MDEntrySize [271] field.

### 6.4.4 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 |

### 6.4.5 Message flow

#### Request for price information

A request for price information initially receives a series of messages for the selected securities at the time of the request. From this moment on it receives messages notifying changes.
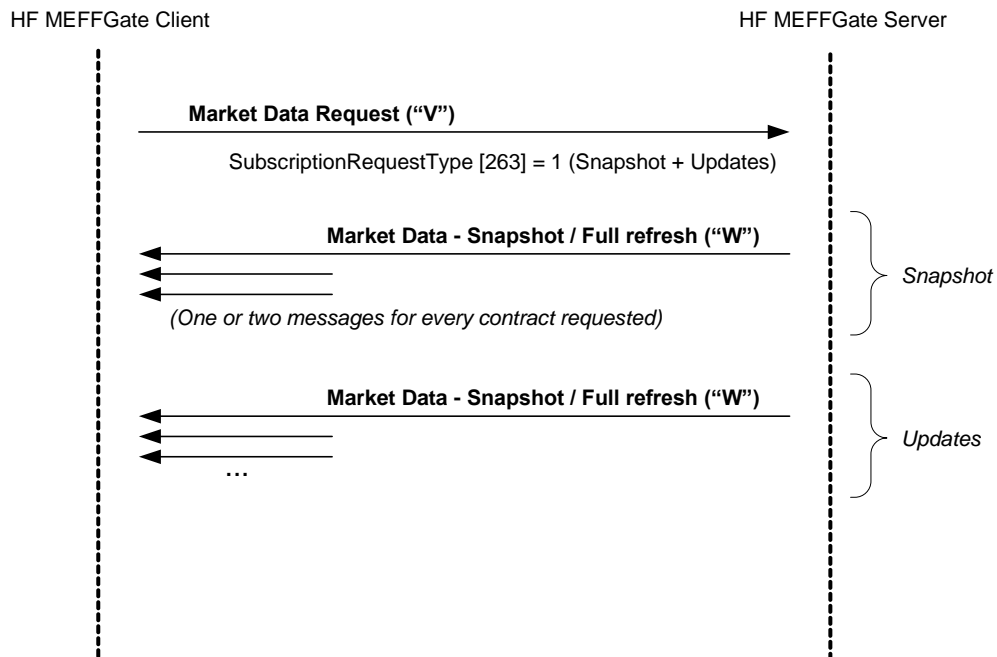
**Incorrect price and orders information request**

When a price and orders information request is incorrect the reply will be a Market Data Request Reject message.

HF MEFFGate Client                                                    HF MEFFGate Server

Market Data Request ("V") →

← Market Data Request Reject ("Y")
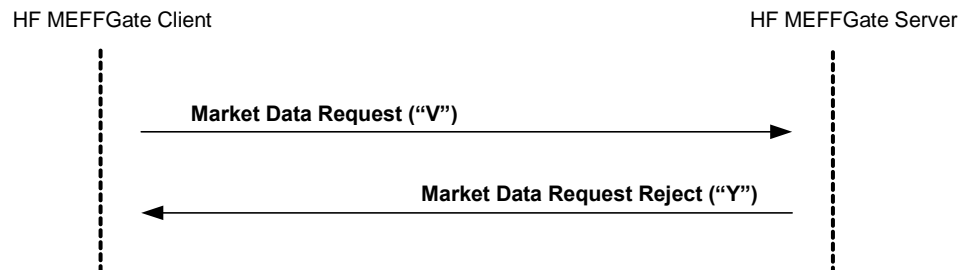
### 6.4.6    Annotations and adaptations of FIX 5.0

The maximum number of subscriptions is limited (see section 4.2 for details)

The Volatility [1188], GrossTradeAmt [381] and TrdMatchID [880] fields have been added to the Market Data Snapshot Full Refresh message

In the Market Data Snapshot Full Refresh message the field EventType [865] with codes greater than 100 is used. The client application should be prepared to manage this situation in a correct way

The optional field MoreSubscriptionsFollowing [21500] has been added to the Market Data Request message to group market information subscription requests

Component block TrdRegTimestamps has been added to the Market Data Snapshot Full Refresh message

## 6.5 Definition of messages

### 6.5.1 Trading Session Status Request (Msg Type = g)

Used by the client to request the session status.

| Tag | Name | Req | Valid values | Format | Description |
|---|---|---|---|---|---|
| | Standard Header | Y | MsgType = g | | |
| 335 | TradSesReqID | Y | | String (10) | Unique identifier for each Trading Session Status Request message |
| 263 | SubscriptionRequestType | Y | 1 = Subscribe | Char | If ApplID [1180] + ApplSeqNum [1181] has been provided in the Logon message, only updates from the point indicated will be sent |
| 21500* | MoreSubscriptionsFollowing | N | Y (suggested), N (default) | Boolean | It allows to group market information subscription requests. For more information see "3.6 - Synchronisation at application level" |
| | Standard Trailer | Y | | | |

### 6.5.2  Trading Session Status (Msg Type = h)

Sent by the server to inform on the session status or to reject a Trading Session Status Request message.

| Tag | Name | Req | Valid values | Format | Description |
|-----|------|-----|--------------|--------|-------------|
| | Standard Header | Y | MsgType = h | | |
| 1180 | ApplID | N | | String | Used in conjunction with ApplSeqNum [1181] to indicate, in subsequent connections, the point from which to receive information |
| 1181 | ApplSeqNum | N | | SeqNum | Used in conjunction with ApplID [1180] to indicate, in subsequent connections, the point from which to receive information |
| 335 | TradSesReqID | N | | String | Identifier of Trading Session Status Request message for reference. This field is always included in the message |
| 336 | TradingSessionID | Y | 100 = IBEX futures hours / FX<br><br>102 = Cross trades (IBEX futures hours)<br><br>105 = Normal hours<br><br>106 = Delta and Basis Trade<br><br>107 = Bono hours<br><br>108 = Cross trades (normal hours)<br><br>109 = Cross trades (Bono hours)<br><br>115 = RFQ (IBEX futures hours)<br><br>116 = RFQ (normal hours)<br><br>117 = RFQ (Bono hours) | String | Trading mode |

| Tag | Name | Req | Valid values | Format | Description |
|-----|------|-----|--------------|--------|-------------|
| | | | 118 = xRolling on Stocks | | |
| 625 | TradingSessionSubID | N | 1 = Pre-Trading (Not Started)<br><br>2 = Scheduled opening auction<br><br>3 = (Continuous) Trading<br><br>4 = Scheduled closing auction<br><br>5 = Post-Trading<br><br>6 = Scheduled intraday auction<br><br>9 = Unscheduled intraday auction<br><br>10 = Out of Main Session Trading<br><br>202 = Not Started<br><br>203 = Open<br><br>204 = Closed | String | Market assigned sub identifier for a trading phase within a trading session.<br><br>Valid values within each Trading Mode are:<br><br>**TradingSessionID [336] = 100 (IBEX futures hours / FX), 105 (Normal hours) and 107 (Bono hours):**<br>1 = Pre-Trading (Not Started)<br>2 = Scheduled opening auction<br>3 = (Continuous) Trading<br>5 = Post-Trading<br>9 = Unscheduled intraday auction<br><br>**TradingSessionID [336] = 102 (Cross trades – IBEX futures hours), 106 (Delta and Basis Trade), 108 (Cross trades - normal hours) and 109 (Cross trades – Bono hours), 115 (RFQ - IBEX futures hours), 116 (RFQ - normal hours), 117 (RFQ - Bono hours) and 118 (xRolling on Stocks):**<br>202 = Not Started<br>203 = Open<br>204 = Closed |
| 325 | UnsolicitedIndicator | N | N = The message is part of a snapshot<br>Y = The message is sent due to an update | Boolean | Contains "Y" when the message is sent as the result of a subscription |
| 340 | TradSesStatus | Y | 1 = Halted<br>2 = Open<br>3 = Closed<br>4 = Pre-Open (Not started)<br>5 = Pre-Close | Int | Status.<br><br>Contains the value 6 (Request Rejected) when the message is used to reject a request.. |

| Tag | Name | Req | Valid values | Format | Description |
|---|---|---|---|---|---|
| | | | 6 = Request Rejected | | The value 4 (Pre-Open) indicates that the Trading Mode is not open yet for trading.<br><br>The value 3 (Closed) indicates the end of a Trading Mode and this is always a final state. |
| 58 | Text | N | | String | Explanation of error. Provided if TradSesStatus = 6 |
| 60* | TransactTime | N | | UTCTimeStamp | Event time |
| | Standard Trailer | Y | | | |

### 6.5.3 Security List Request (Msg Type = x)

Used by the client to request the security definitions and the security status

| 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 |
| 559 | SecurityListRequest Type | Y | 1 | Int | Selection criteria used |
|  | Start <Instrument> |  |  |  |  |
| 55 | Symbol | Y | [N/A] |  | Always [N/A] |
| 48 | SecurityID | N | See table 7 in document "Codification Tables" for a list of possible values | String | Underlying asset |
| 22 | SecurityIDSource | N | 8 = Exchange Symbol | String | Required if SecurityID is present. |
| 167 | SecurityType | N | See table 6 in document "Codification Tables" | String | Product type |
| 200 | MaturityMonthYear | N | YYYYMM or YYYYMMDD or YYYYMMwW | Month-Year | Contract expiration |
|  | End <Instrument> |  |  |  |  |
| 263 | SubscriptionReques tType | Y | 1 = Subscribe | Char | Indicates the type of security status request. If ApplID [1180] + ApplSeqNum [1181] has been provided in the Logon message, only updates from the point indicated will be sent |
| 2150 0* | MoreSubscriptionsF ollowing | N | Y (suggested), N (default) | Boolean | It allows to group market information subscription requests. For more information see "3.6 - Synchronisation at application level" |
|  | Standard Trailer | Y |  |  |  |

### 6.5.4 Security List (Msg Type = y)

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

| Tag | Name | Req | Valid values | Format | Description |
|-----|------|-----|--------------|--------|-------------|
| | Standard Header | Y | MsgType = y | | |
| 1180 | ApplID | N | | String | Used in conjunction with ApplSeqNum [1181] to indicate, in subsequent connections, the point from which to receive information |
| 1181 | ApplSeqNum | N | | SeqNum | Used in conjunction with ApplID [1180] to indicate, in subsequent connections, the point from which to receive information |
| 320 | SecurityReqID | N | | String | Identifier of Security List Request message that it is replying to |
| 322 | SecurityResponseID | N | | String | Unique identifier for each Security List message |
| 560 | SecurityRequestResult | N | 0=Valid request<br><br>1=Invalid or unsupported request<br><br>2=No instruments found that match selection criteria<br><br>4=Instrument data temporarily unavailable<br><br>5=Request was rejected because the SecurityType specified is not supported | Int | Result of request identified by SecurityReqID |
| 393 | TotNoRelatedSym | N | | Int | Total number of securities that meet the selection criteria in the request. The number of securities that the message contains is indicated in the NoRelatedSym field. This field can be present when SecurityRequestResult = 0 |
| 1301 | MarketID | N | | Exchange | Operating MIC |
| 1300 | MarketSegmentID | N | | String | Segment MIC |

| Tag | Name | Req | Valid values | Format | Description |
|-----|------|-----|--------------|--------|-------------|
| 893 | LastFragment | N | | Boolean | Indicates when the message is the last in a sequence in response to a single request. This field can be present when SecurityRequest-Result = 0 |
| 146 | NoRelatedSym | N | 1 | NumInGroup | Indicates the number of securities contained in this message |
| | Start <Instrument> | | | | |
| →55 | Symbol | N | [N/A] or security code | String(22) | |
| →48 | SecurityID | N | See table 7 in document "Codification Tables" for a list of possible values | String | Underlying asset |
| →22 | SecurityIDSource | N | 8 = Exchange Symbol | String | |
| | Start <SecAltIDGrp> | | | | |
| → 454 | NoSecurityAltID | N | | NumInGroup | |
| →→ 455 | SecurityAltID | N | | String | • When SecurityAltIDSource [456] = 4, it contains the ISIN code for the contract<br><br>• When SecurityAltIDSource [456] = J, it contains the FISN for the contract (Finantial Instrument short name in compliance with ISO 18774)<br><br>• When SecurityAltIDSource [456] = T, it contains the LEI of the issuer |
| →→ 456 | SecurityAltIDSource | N | 4 = ISIN number<br><br>J = FISN<br><br>T = LEI of the issuer | String | |
| | End <SecAltIDGrp> | | | | |
| → 1151 | SecurityGroup | N | See table 8 in document "Codification Tables" for a list of values | String | Product family |
| → 461 | CFICode | N | | String(6) | Contract type in accordance with the ISO 10962 standard |

| Tag | Name | Req | Valid values | Format | Description |
|---|---|---|---|---|---|
| → 167 | SecurityType | N | See table 6 in document "Codification Tables" | String | Product type |
| → 762 | SecuritySubType | N | See table 9 in document "Codification Tables" for a list of possible values | String | Strategy type |
| → 200 | MaturityMonthYear | N | YYYYMM or YYYYMMDD or YYYYMMwW | Month-Year | Security expiration |
| → 541 | MaturityDate | N |  | LocalMktDate | Expiration date |
| → 225 | IssueDate | N |  | LocalMktDate | Date security issued |
| → 202 | StrikePrice | N |  | Price | Exercise price. Only present for options |
| → 968 | StrikeValue | N |  | Float | For stocks derivatives, number of shares for each security |
| → 206 | OptAttribute | N |  | Char | Security version number, provided to support versioning of securities as a result of corporate actiopns or events |
| → 231 | ContractMultiplier | N |  | Float | Conversion factor between price units and monetary units |
| → 969 | MinPriceIncrement | N |  | Float | Minimum amount allowed for price change when sending an order request |
| → 996 | UnitOfMeasure | N | Mwh = Megawatt hours | String | The unit of measure of the underlying commodity upon which the contract is based |
| → 1193 | SettlMethod | N | C = Cash settlement required<br><br>P = Physical settlement required | Char | Settlement method for this security |
| → 1194 | ExerciseStyle | N | 0 = European<br><br>1 = American | Int | Type of exercise of this security |
| → 201 | PutOrCall | N | 0 = Put<br><br>1 = Call | Int | Indicates whether an option contract is a put or call |
| → 1244 | FlexibleIndicator | N | Y = Flexible<br><br>N = Standard (default) | Boolean | Used to indicate if this security has been defined as flexible according to "non-standard" means. |

| Tag | Name | Req | Valid values | Format | Description |
|-----|------|-----|--------------|--------|-------------|
| | | | | | When not informed, means "N = Standard " |
| → 107 | SecurityDesc | N | See table 5 in document "Codification Tables" | String | Description of the contract subgroup |
| | Start <EvntGrp> | | | | |
| → 864 | NoEvents | N | | NumInGroup | |
| →→ 865 | EventType | N | 101 = Last trading day<br><br>114 = Number of decimals in the price for this security<br><br>132 = Maximum number of decimals allowed in orders<br><br>146 = LIS-pre limit (Large in Scale)<br><br>147 = SSTI-pre limit (Size Specific to Instrument)<br><br>148 = LIS-post limit (Large in Scale)<br><br>149 = SSTI-post limit (Size Specific to Instrument)<br><br>150 = Liquid instrument<br><br>151 = Adjustments rule<br><br>152 = Limit cap above which orders are not permitted | Int | |

| Tag | Name | Req | Valid values | Format | Description |
|-----|------|-----|--------------|--------|-------------|
| | | | 153 = Security admits self-match prevention | | |
| | | | 154 = Security request for admission to trading by issuer | | |
| | | | 155 = Commodity derivative Indicator to indicate whether the security falls within the definition of commodities derivative under Article 2(1)(30) of Regulation (EU) No 600/2014 | | |
| | | | 156 = Trading obligation. Indicates whether the security has to be traded in a regulated exchange | | |
| | | | 159 = Contains the stock in case the underlying is the dividend of the stock | | |
| | | | 160 = Base product | | |
| | | | 161 = Sub product | | |
| | | | 162 = Further sub product | | |

| Tag | Name | Req | Valid values | Format | Description |
|---|---|---|---|---|---|
| | | | 167 = xRolling closing type | | |
| | | | 168 = xRolling Buyer financing rate | | |
| | | | 169 = xRolling buyer rate markup | | |
| | | | 170 = xRolling seller financing rate | | |
| | | | 171 = xRolling seller rate markup | | |
| | | | 172 = xRolling Dividend settlement percentage | | |
| | | | 173 = xRolling differential between payment date and receipt of ordinary dividend flow | | |
| →→ 866 | EventDate | N | | LocalMktDate | Last trading day, when EventType = 101 |
| →→ 868 | EventText | N | | String | If EventType = 114, it contains the number of decimals in the price for this security |
| | | | | | If EventType = 132, it contains the maximum number of decimals allowed in orders |
| | | | | | If EventType = 146, it contains the LIS-pre limit (Large in Scale) |
| | | | | | If EventType = 147, it contains the SSTI-pre limit (Size Specific to Instrument) |
| | | | | | If EventType = 148, it contains the LIS-post limit (Large in Scale) |

| Tag | Name | Req | Valid values | Format | Description |
|-----|------|-----|--------------|--------|-------------|
| | | | | | If EventType = 149, it contains the SSTI-post limit (Size Specific to Instrument) |
| | | | | | If EventType = 150, indicates whether the security is Liquid or Illiquid:<br>   Y – Liquid<br>   N – Illiquid |
| | | | | | If EventType = 151, it contains the adjustments rule:<br>   E – Extraordinary dividend adjustments only<br>   T - Total |
| | | | | | If EventType = 152, it contains the Nominal limit cap above which orders are not permitted |
| | | | | | If EventType = 153, it indicates whether the Security admits self-match prevention or not:<br>   Y – It admits self-match prevention<br>   N – It doesn't admit self-match prevention |
| | | | | | If EventType = 154, indicates whether the security is request for admission to trading by issuer or by Exchange own initiative:<br>   Y – Request for admission to trading by issuer<br>   N – Request for admission to trading by Exchange own initiative |
| | | | | | If EventType = 155, indicates whether the security falls within the definition of commodities derivative under Article 2(1)(30) of Regulation (EU) No 600/2014:<br>   Y – It is a Commodity derivative<br>   N – It is NOT a Commodity derivative |
| | | | | | If EventType = 156, indicates whether the security has to be traded in a regulated exchange (Trading Obligation):<br>   Y – Yes |

| Tag | Name | Req | Valid values | Format | Description |
|-----|------|-----|--------------|--------|-------------|
| | | | | | N – No |
| | | | | | If EventType = 159, it contains If EventType = 159, it contains the stock in case the underlying is the dividend of the stock |
| | | | | | If EventType = 160, 161 or 162, it contains the classification of commodity derivatives (see table 11 in document "Codification Tables") |
| | | | | | If EventType = 167, it indicates if the xRolling can be closed by any of the counterparties before the expiry date.<br>M: Market (Default value ). Value to be used in FLEX xRolling with a closed expiry date and no possibility to modify it.<br>1: By buyer<br>2: By seller<br>A: By both |
| | | | | | If EventType = 168, it contains the Buyer Financing Rate<br>S: €STR (Default value for non-FLEX xRolling)<br>F: FISAnalitics<br>M: MEFF rate<br>0: Zero<br>' ': N/A  (used in FLEX, in the financing leg) |
| | | | | | If EventType = 169, it contains the buyer rate markup<br>From -100.0000 to 100.0000 |
| | | | | | If EventType = 170, it contains the Seller Financing Rate<br>S: €STR (Default value for non-FLEX xRolling)<br>F: FISAnalitics<br>M: MEFF rate<br>0: Zero<br>' ': N/A  (used in FLEX, in the financing leg) |
| | | | | | If EventType = 171, , it contains the seller rate markup<br>From -100.0000 to 100.0000 |

| Tag | Name | Req | Valid values | Format | Description |
|---|---|---|---|---|---|
| | | | | | If EventType = 172, it contains the Dividend settlement percentage From 0.00 to 100.00 |
| | | | | | If EventType = 173, it contains the differential between payment date and receipt of ordinary dividend flow 0-999 |
| | End <EvntGrp> | | | | |
| | 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 |
| →→ 2125 | ComplexEventCurrencyTwo | N | | Currency | Quoted currency code. Follows ISO 4217 standard |
| | End < ComplexEvents > | | | | |
| | End <Instrument> | | | | |
| | Start <SecurityTradingRules> | | | | |
| | Start <BaseTradingRules > | | | | |
| → 562 | MinTradeVol | N | | Qty | The minimum trading volume for an order of this security |
| → 561 | RoundLot | N | | Qty | The trading lot size. The order volumes of this security must be a multiple of this quantity. |
| | End <BaseTradingRules > | | | | |
| | End <SecurityTradingRules> | | | | |
| | Start <StrikeRules> | | | | |
| → 1201 | NoStrikeRules | N | 1 | NumInGroup | |
| →→ 1223 | StrikeRuleID | N | [N/A] | String | |
| | Start <MaturityRules> | | | | |
| →→ 1236 | NoMaturityRules | N | 1 | NumInGroup | |

| Tag | Name | Req | Valid values | Format | Description |
|---|---|---|---|---|---|
| →→→ 1222 | MaturityRuleID | N | [N/A] | String | |
| →→→ 1302 | MaturityMonthYear IncrementUnits | N | 0 = Months<br>1  1 = Days<br>2 = Weeks<br>3 = Years | Int | Periodicity |
| →→→ 1241 | StartMaturityMonth Year | N | YYYYMMDD | Month-Year | Start delivery date for Energy segment contracts |
| →→→ 1226 | EndMaturityMonthY ear | N | YYYYMMDD | Month-Year | End delivery date for Energy segment contracts |
| →→→ 1229 | MaturityMonthYear Increment | N | | Int | |
| | End < MaturityRules > | | | | |
| | End <StrikeRules> | | | | |
| → 711 | NoUnderlyings | N | 1 | NumInGrou p | Present if the security has another security as its underlying |
| | Start <UnderlyingInstrum ent> | | | | |
| →→ 311 | UnderlyingSymbol | N | | String(22) | Symbol for underlying security |
| →→ 457 | NoUnderlyingSecuri tyAltID | N | | NumInGrou p | |
| →→→ 458 | UnderlyingSecurity AltID | N | | String | When UnderlyingSecurityAltIDSource [459] = T, it contains the LEI of the underlying issuer |
| →→→ 459 | UnderlyingSecurity AltIDSource | N | T = LEI of the underlying issuer | String | |
| | | | | | |
| →→ 318 | UnderlyingCurrency | N | | Currency | Currency code of the underlying security. Follows ISO 4217 standard |
| | End <UnderlyingInstrum ent> | | | | |
| →15 | Currency | N | | Currency | Currency code. Follows ISO 4217 standard |
| | Start <Stipulations> | | | | |
| → 232 | NoStipulations | N | | NumInGrou p | |
| →→ 233 | StipulationType | N | 100 = IBEX futures hours / FX | String | Trading Mode |

| Tag | Name | Req | Valid values | Format | Description |
|-----|------|-----|--------------|--------|-------------|
| | | | 102 = Cross trades (IBEX futures hours) | | |
| | | | 105 = Normal hours | | |
| | | | 106 = Delta and Basis Trade | | |
| | | | 107 = Bono hours | | |
| | | | 108 = Cross trades (normal hours) | | |
| | | | 109 = Cross trades (Bono hours) | | |
| | | | 115 = RFQ (IBEX futures hours) | | |
| | | | 116 = RFQ (normal hours) | | |
| | | | 117 = RFQ (Bono hours) | | |
| | | | 118= xRolling on Stocks | | |
| →→ 234 | StipulationValue | N | | String | Indicates the valid Trading Modes for this security. The possible values are Y/N. If it's not sent, means 'N' |
| | End <Stipulations> | | | | |
| → 555 | NoLegs | N | | NumInGroup | Only present in time-spread or strategies contracts |
| →→ | Start <InstrumentLeg> | | | | |
| →→ 600 | LegSymbol | N | | String(22) | Contract code. Present if NoLegs has been specified |
| →→ 623 | LegRatioQty | N | | Float | The ratio of quantity for this individual leg relative to the entire multileg security |
| →→ 624 | LegSide | N | 1 = Buy 2 = Sell | Char | Indicates if the contract LegSymbol is to buy or sell. |

| Tag | Name | Req | Valid values | Format | Description |
|---|---|---|---|---|---|
| | | | | | Present if NoLegs has been specified |
| →→ 566 | LegPrice | N | | Price | Price for this leg |
| | End <InstrumentLeg> | | | | |
| →58 | Text | N | | String | Security description<br><br>If SecurityRequestResult [560] > 0 contains an explanation of the error |
| | Standard Trailer | Y | | | |

### 6.5.5 Security List Update Report (Msg Type = BK)

Message sent by the server for reporting updates, during the session, to the security definition.

| Tag | Name | Req | Valid values | Format | Description |
|---|---|---|---|---|---|
| | Standard Header | Y | MsgType = BK | | |
| 1180 | ApplID | N | | String | Used in conjunction with ApplSeqNum [1181] to indicate, in subsequent connections, the point from which to receive information |
| 1181 | ApplSeqNum | N | | SeqNum | Used in conjunction with ApplID [1180] to indicate, in subsequent connections, the point from which to receive information |
| 964 | SecurityReportID | N | | String | Unique identifier for each Security List Update Report message |
| 320 | SecurityReqID | N | | String | Identifier of Security List Request message that it is replying to |
| 980 | SecurityUpdateAction | N | A – Add<br><br>D – Delete<br><br>M - Modify | Char | |
| 1301 | MarketID | N | | Exchange | Operating MIC |
| 1300 | MarketSegmentID | N | | String | Segment MIC |
| 60 | TransactTime | N | | UTCTimeStamp | Event time |
| 146 | NoRelatedSym | N | 1 | NumInGroup | Indicates the number of securities contained in this message |
| | Start <Instrument> | | | | |
| →55 | Symbol | N | | String(22) | Security code |
| →48 | SecurityID | N | See table 7 in document "Codification Tables" for a list of possible values | String | Underlying asset |
| →22 | SecurityIDSource | N | 8 = Exchange Symbol | String | |
| | Start <SecAltIDGrp> | | | | |
| →454 | NoSecurityAltID | N | | NumInGroup | |
| →→455 | SecurityAltID | N | | String | • When SecurityAltIDSource [456] = 4, it contains the ISIN code for the contract<br><br>• When SecurityAltIDSource [456] = J, it contains the FISN for the contract (Finantial Instrument short name in compliance with ISO 18774) |

| Tag | Name | Req | Valid values | Format | Description |
|---|---|---|---|---|---|
| | | | | | • When SecurityAltIDSource [456] = T, it contains the LEI of the issuer |
| →→ 456 | SecurityAltIDSource | N | 4 = ISIN number<br><br>J = FISN<br><br>T = LEI of the issuer | String | |
| | End <SecAltIDGrp> | | | | |
| → 1151 | SecurityGroup | N | See table 8 in document "Codification Tables" for a list of values | String | Product family |
| → 461 | CFICode | N | | String(6) | Contract type in accordance with the ISO 10962 standard |
| → 167 | SecurityType | N | See table 6 in document "Codification Tables" | String | Product type |
| → 762 | SecuritySubType | N | See table 9 in document "Codification Tables" for a list of possible values | String | Strategy type |
| → 200 | MaturityMonthYear | N | YYYYMM or YYYYMMDD or YYYYMMwW | Month-Year | Security expiration |
| → 541 | MaturityDate | N | | LocalMktDate | Expiration date |
| → 225 | IssueDate | N | | LocalMktDate | Date security issued |
| → 202 | StrikePrice | N | | Price | Exercise price. Only present for options |
| → 968 | StrikeValue | N | | Float | For stocks derivatives, number of shares for each security |
| → 206 | OptAttribute | N | | Char | Security version number, provided to support versioning of securities as a result of corporate actions or events |
| → 231 | ContractMultiplier | N | | Float | Conversion factor between price units and monetary units |
| → 969 | MinPriceIncrement | N | | Float | Minimum amount allowed for price change when sending an order request |
| → 996 | UnitOfMeasure | N | Mwh = Megawatt hours | String | The unit of measure of the underlying commodity upon which the contract is based |

| Tag | Name | Req | Valid values | Format | Description |
|---|---|---|---|---|---|
| → 1193 | SettlMethod | N | C = Cash settlement required <br><br> P = Physical settlement required | Char | Settlement method for this security |
| → 1194 | ExerciseStyle | N | 0 = European <br><br> 1 = American | Int | Type of exercise of this security |
| → 201 | PutOrCall | N | 0 = Put <br><br> 1 = Call | Int | Indicates whether an option contract is a put or call |
| → 1244 | FlexibleIndicator | N | Y = Flexible <br><br> N = Standard (default) | Boolean | Used to indicate if this security has been defined as flexible according to "non-standard" means. <br><br> When not informed, means "N = Standard " |
| → 107 | SecurityDesc | N | See table 5 in document "Codification Tables" | String | Description of the contract subgroup |
| | Start < EvntGrp > | | | | |
| → 864 | NoEvents | N | | NumInGroup | |
| →→ 865 | EventType | N | 101 = Last trading day <br><br> 114 = Number of decimals in the price for this security <br><br> 132 = Maximum number of decimals allowed in orders <br><br> 146 = LIS-pre limit (Large in Scale) <br><br> 147 = SSTI-pre limit (Size Specific to Instrument) <br><br> 148 = LIS-post limit (Large in Scale) | Int | |

| Tag | Name | Req | Valid values | Format | Description |
|-----|------|-----|--------------|--------|-------------|
| | | | 149 = SSTI-post limit (Size Specific to Instrument) | | |
| | | | 150 = Liquid instrument | | |
| | | | 151 = Adjustments rule | | |
| | | | 152 = Nominal limit cap above which orders are not permitted | | |
| | | | 153 = Security admits self-match prevention | | |
| | | | 154 = Security request for admission to trading by issuer | | |
| | | | 155 = Commodity derivative Indicator to indicate whether the security falls within the definition of commodities derivative under Article 2(1)(30) of Regulation (EU) No 600/2014 | | |

| Tag | Name | Req | Valid values | Format | Description |
|-----|------|-----|--------------|--------|-------------|
| | | | 156 = Trading obligation. Indicates whether the security has to be traded in a regulated exchange | | |
| | | | 159 = Contains the stock in case the underlying is the dividend of the stock | | |
| | | | 160 = Base product | | |
| | | | 161 = Sub product | | |
| | | | 162 = Further sub product | | |
| | | | 167 =  xRolling closing type | | |
| | | | 168 =  xRolling Buyer financing rate | | |
| | | | 169 =  xRolling buyer rate markup | | |
| | | | 170 =  xRolling seller financing rate | | |
| | | | 171 =  xRolling seller rate markup | | |
| | | | 172 =  xRolling Dividend settlement percentage | | |

| Tag | Name | Req | Valid values | Format | Description |
|---|---|---|---|---|---|
| | | | 173 = xRolling differential between payment date and receipt of ordinary dividend flow | | |
| →→ 866 | EventDate | N | | LocalMktDate | Last trading day, when EventType = 101 |
| →→ 868 | EventText | N | | String | If EventType = 114, it contains the number of decimals in the price for this security<br><br>If EventType = 132, it contains the maximum number of decimals allowed in orders<br><br>If EventType = 146, it contains the LIS-pre limit (Large in Scale)<br><br>If EventType = 147, it contains the SSTI-pre limit (Size Specific to Instrument)<br><br>If EventType = 148, it contains the LIS-post limit (Large in Scale)<br><br>If EventType = 149, it contains the SSTI-post limit (Size Specific to Instrument)<br><br>If EventType = 150, indicates whether the security is Liquid or Illiquid:<br>  Y – Liquid<br>  N – Illiquid<br><br>If EventType = 151, it contains the adjustments rule:<br>  E – Extraordinary dividend adjustments only<br>  T - Total<br><br>If EventType = 152, it contains the Nominal limit cap above which orders are not permitted<br><br>If EventType = 153, it indicates whether the Security admits self-match prevention or not:<br>  Y – It admits self-match prevention |

| Tag | Name | Req | Valid values | Format | Description |
|-----|------|-----|--------------|--------|-------------|
| | | | | | N – It doesn't admit self-match prevention |
| | | | | | If EventType = 154, indicates whether the security is request for admission to trading by issuer or by Exchange own initiative: |
| | | | | |   Y – Request for admission to trading by issuer |
| | | | | |   N – Request for admission to trading by Exchange own initiative |
| | | | | | If EventType = 155, indicates whether the security falls within the definition of commodities derivative under Article 2(1)(30) of Regulation (EU) No 600/2014: |
| | | | | |   Y – It is a Commodity derivative |
| | | | | |   N – It is NOT a Commodity derivative |
| | | | | | If EventType = 156, indicates whether the security has to be traded in a regulated exchange (Trading Obligation): |
| | | | | |   Y – Yes |
| | | | | |   N – No |
| | | | | | If EventType = 159, it contains the stock in case the underlying is the dividend of the stock |
| | | | | | If EventType = 160, 161 or 162, it contains the classification of commodity derivatives (see table 11 in document "Codification Tables") |
| | | | | | If EventType = 167, it indicates if the xRolling can be closed by any of the counterparties before the expiry date. |
| | | | | | M: Market (Default value ). Value to be used in FLEX xRolling with a closed expiry date and no possibility to modify it. |
| | | | | | 1: By buyer |
| | | | | | 2: By seller |
| | | | | | A: By both |

| Tag | Name | Req | Valid values | Format | Description |
|---|---|---|---|---|---|
| | | | | | If EventType = 168, it contains the Buyer Financing Rate<br>S: €STR (Default value for non-FLEX xRolling)<br>F: FISAnalitics<br>M: MEFF rate<br>0: Zero<br>' ': N/A  (used in FLEX, in the financing leg)<br><br>If EventType = 169, it contains the buyer rate markup<br>From -100.0000 to 100.0000<br><br>If EventType = 170, it contains the Seller Financing Rate<br>S: €STR (Default value for non-FLEX xRolling)<br>F: FISAnalitics<br>M: MEFF rate<br>0: Zero<br>' ': N/A  (used in FLEX, in the financing leg)<br><br>If EventType = 171, , it contains the seller rate markup<br>From -100.0000 to 100.0000<br><br>If EventType = 172, it contains the Dividend settlement percentage<br>From 0.00 to 100.00<br><br>If EventType = 173, it contains the differential between payment date and receipt of ordinary dividend flow<br>0-999 |
| | End < EvntGrp > | | | | |
| | Start < ComplexEvents > | | | | |
| →<br>1483 | NoComplexEvents | N | 1 | NumInGroup | |
| →→<br>1484 | ComplexEventType | N | 16 = Foreign exchange cross currency | Int | |
| →→<br>2124 | ComplexEventCurrencyOne | N | | Currency | Base currency code. Follows ISO 4217 standard |
| →→<br>2125 | ComplexEventCurrencyTwo | N | | Currency | Quoted currency code. Follows ISO 4217 standard |

| Tag | Name | Req | Valid values | Format | Description |
|---|---|---|---|---|---|
| | End < ComplexEvents > | | | | |
| | End < Instrument > | | | | |
| | Start <SecurityTradingRules> | | | | |
| | Start <BaseTradingRules > | | | | |
| → 562 | MinTradeVol | N | | Qty | The minimum trading volume for an order of this security |
| → 561 | RoundLot | N | | Qty | The trading lot size. The order volumes of this security must be a multiple of this quantity. |
| | End <BaseTradingRules > | | | | |
| | End <SecurityTradingRules> | | | | |
| | Start <StrikeRules> | | | | |
| → 1201 | NoStrikeRules | N | 1 | NumInGroup | |
| →→ 1223 | StrikeRuleID | N | [N/A] | String | |
| | Start <MaturityRules> | | | | |
| →→ 1236 | NoMaturityRules | N | 1 | NumInGroup | |
| →→ → 1222 | MaturityRuleID | N | [N/A] | String | |
| →→ → 1302 | MaturityMonthYearIncrementUnits | N | 0 = Months 2   1 = Days 2 = Weeks 3 = Years | Int | Periodicity |
| →→ → 1241 | StartMaturityMonthYear | N | YYYYMMDD | Month-Year | Start delivery date for Energy segment contracts |
| →→ → 1226 | EndMaturityMonthYear | N | YYYYMMDD | Month-Year | End delivery date for Energy segment contracts |
| →→ → 1229 | MaturityMonthYearIncrement | N | | Int | |
| | End < MaturityRules > | | | | |
| | End <StrikeRules> | | | | |
| → 711 | NoUnderlyings | N | 1 | NumInGroup | Present if the security has another security as its underlying |

| Tag | Name | Req | Valid values | Format | Description |
|---|---|---|---|---|---|
| | Start <UnderlyingInstrument> | | | | |
| →→ 311 | UnderlyingSymbol | N | | String(22) | Symbol for underlying security |
| →→ 457 | NoUnderlyingSecurityAltID | N | | NumInGroup | |
| →→→ 458 | UnderlyingSecurityAltID | N | | String | When UnderlyingSecurityAltIDSource [459] = T, it contains the LEI of the underlying issuer |
| →→→ 459 | UnderlyingSecurityAltIDSource | N | T = LEI of the underlying issuer | String | |
| →→ 318 | UnderlyingCurrency | N | | Currency | Currency code of the underlying and strike. Follows ISO 4217 standard |
| | End <UnderlyingInstrument> | | | | |
| →15 | Currency | N | | Currency | Currency code. Follows ISO 4217 standard |
| | Start <Stipulations> | | | | |
| → 232 | NoStipulations | N | | NumInGroup | |
| →→ 233 | StipulationType | N | 100 = IBEX futures hours / FX<br><br>102 = Cross trades (IBEX futures hours)<br><br>105 = Normal hours<br><br>106 = Delta and Basis Trade<br><br>107 = Bono hours<br>108 = Cross trades (normal hours)<br><br>109 = Cross trades (Bono hours)<br><br>115 = RFQ (IBEX futures hours) | String | Trading Mode |

| Tag | Name | Req | Valid values | Format | Description |
|-----|------|-----|--------------|--------|-------------|
| | | | 116 = RFQ (normal hours)<br><br>117 = RFQ (Bono hours)<br><br>118 = xRolling on Stocks | | |
| →→ 234 | StipulationValue | N | | String | Indicates the valid Trading Modes for this security. The possible values are Y/N. If it's not sent, means 'N' |
| | End <Stipulations> | | | | |
| → 555 | NoLegs | N | | NumInGroup | Only present in time-spread or strategies contracts |
| →→ | Start <InstrumentLeg> | | | | |
| →→ 600 | LegSymbol | N | | String(22) | Contract code.<br><br>Present if NoLegs has been specified |
| →→ 623 | LegRatioQty | N | | Float | The ratio of quantity for this individual leg relative to the entire multileg security |
| →→ 624 | LegSide | N | 1 = Buy<br><br>2 = Sell | Char | Indicates if the contract LegSymbol is to buy or sell.<br><br>Present if NoLegs has been specified |
| →→ 566 | LegPrice | N | | Price | Price for this leg |
| | End <InstrumentLeg> | | | | |
| →58 | Text | N | | String | Security description |
| | Standard Trailer | Y | | | |

### 6.5.6 Security Status Request (MsgType = e)

Used by the client to request the status of securities.

| Tag | Name | Req | Valid values | Format | Description |
|---|---|---|---|---|---|
| | Standard Header | Y | MsgType = e | | |
| 324 | SecurityStatusReqID | Y | | String (10) | Unique identifier for each Security Status Request message |
| | Start <Instrument> | | | | |
| 55 | Symbol | Y | [N/A] | | Always [N/A] |
| 48 | SecurityID | N | See table 7 in document "Codification Tables" for a list of possible values | String | Underlying asset |
| 22 | SecurityIDSource | N | 8 = Exchange Symbol | String | Required if SecurityID is present |
| 167 | SecurityType | N | See table 6 in document "Codification Tables" | String | Product type |
| 200 | MaturityMonthYear | N | YYYYMM or YYYYMMDD or YYYYMMwW | Month-Year | Contract expiration |
| | End <Instrument> | | | | |
| 263 | SubscriptionRequestType | Y | 1 = Subscribe | Char | If ApplID [1180] + ApplSeqNum [1181] has been provided in the Logon message, only updates from the point indicated will be sent |
| 21500* | MoreSubscriptionsFollowing | N | Y (suggested), N (default) | Boolean | It allows to group market information subscription requests. For more information see "3.6 - Synchronisation at application level" |
| | Standard Trailer | Y | | | |

### 6.5.7 Security Status (MsgType = f)

Message sent by the server to inform on the status of one security.

| Tag | Name | Req | Valid values | Format | Description |
|---|---|---|---|---|---|
| | Standard Header | Y | MsgType = f | | |
| 1180 | ApplID | N | | String | Used in conjunction with ApplSeqNum [1181] to indicate, in subsequent connections, the point from which to receive information |
| 1181 | ApplSeqNum | N | | SeqNum | Used in conjunction with ApplID [1180] to indicate, in subsequent connections, the point from which to receive information |
| 324 | SecurityStatusReqID | N | | String | Identifier of the Security Status Request message being replied to. This field is always included in the message |
| | Start <Instrument> | | | | |
| 55 | Symbol | Y | [N/A] or security code | String(22) | Security code.<br><br>It contains [N/A] when the message corresponds to a set of contracts or when SecurityTradingStatus [326] = 20 (Unknown or invalid) |
| 48 | SecurityID | N | See table 7 in document "Codification Tables" for a list of possible values | String | Underlying asset.<br><br>If not specified means "for all the underlying assets" |
| 454 | NoSecurityAltID | N | | NumInGroup | |
| → 455 | SecurityAltID | N | | String | • When SecurityAltIDSource [456] = 4, it contains the ISIN security code |
| → 456 | SecurityAltIDSource | N | 4 = ISIN number | String | |
| 22 | SecurityIDSource | N | 8 = Exchange Symbol | String | Present if SecurityID has been specified |
| 1151 | SecurityGroup | N | See table 8 in document "Codification Tables" for a list of values | String | Product family.<br><br>If not specified means "for all the product families" |
| 167 | SecurityType | N | See table 6 in document "Codification Tables" | String | Product type.<br><br>If not specified means "for all the product types" |

| Tag | Name | Req | Valid values | Format | Description |
|---|---|---|---|---|---|
| 200 | MaturityMonthYear | N | YYYYMM or YYYYMMDD or YYYYMMwW | Month-Year | Contract expiration.<br><br>If not specified means "for all the contract expirations" |
| | End \<Instrument\> | | | | |
| 325 | UnsolicitedIndicator | N | N = The message is part of a snapshot<br><br>Y = The message is sent as the result of an update | Boolean | Contains "Y" when the message is sent due to a subscription, and otherwise "N".<br><br>This field is always present in the message |
| 326 | SecurityTradingStatus | N | 17 = Ready to trade<br><br>18 = Not available for trading<br><br>19 = Not Traded on this Segment<br><br>20 = Unknown or Invalid<br><br>21 = Pre-Open<br><br>23 = Fast Market<br><br>100 = Extraordinary Market Conditions | Int | Informs on the security status. The value "21" indicates that the security or product family is under auction. This value must not be confused with the "Pre-Open" segment status, which indicates that no security can be traded. (See field 340, TradSesStatus, of the Trading Session Status message).<br><br>To evaluate this tag, TradSesStatus [340] in the Trading Session Status message must also be taken into account. |
| 327 | HaltReason | N | 100 = Halted by Regulator<br><br>101 = Halted by Market Surveillance | Int | Halt reason |
| 332 | HighPx | N | | Price | Maximum price accepted for a contract. This value may vary during a trading session |
| 333 | LowPx | N | | Price | Minimum price accepted for a contract. This value may vary during a trading session |
| 60 | TransactTime | N | | UTCTimeStamp | Event time |

| Tag | Name | Req | Valid values | Format | Description |
|-----|------|-----|--------------|--------|-------------|
| 58 | Text | N | | String | Contains an explanation of the error. May be provided if SecurityTradingStatus = 19 or 20 |
| | Standard Trailer | Y | | | |

### 6.5.8 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 |
| 263 | SubscriptionRequestType | Y | 1 = Subscribe | Char | If ApplID [1180] + ApplSeqNum [1181] has been provided in the Logon message, only updates from the point indicated will be sent |
| 264 | MarketDepth | Y | 0 = Full Book<br>1 = Top of Book<br>n = exact depth (n>1) | Int | Prices depth<br>Ignored if none of the MDEntryType occurrences are Bid or Offer |
| 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 | 0 = Bid<br>1 = Offer<br>2 = Trade (last)<br>4 = Opening Price<br>6 = Settlement Price<br>7 = Trading Session High Price<br>8 = Trading Session Low Price<br>9 = Trading session VWAP price<br>B = Trade Volume (total volume for security in session)<br>C = Open Interest<br>M = Prior Settle Price<br>N = Session High Bid<br>O = Session Low Offer | Char | Type of market information requested |
| 146 | NoRelatedSym | Y | 1 | NumInGroup | Number of selection criteria |
|  | Start <Instrument> |  |  |  |  |

| Tag | Name | Req | Valid values | Format | Description |
|---|---|---|---|---|---|
| →55 | Symbol | Y | [N/A] | String | Always [N/A] |
| →48 | SecurityID | N | See table 7 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 |
| →167 | SecurityType | N | See table 6 in document "Codification Tables" | String | Product type |
| →200 | MaturityMonthYear | N | YYYYMM or YYYYMMDD or YYYYMMwW | Month-Year | Contract expiration |
| | End <Instrument> | | | | |
| 21500* | MoreSubscriptionsFollowing | N | Y (suggested), N (default) | Boolean | It allows to group market information subscription requests. For more information see "3.6 - Synchronisation at application level" |
| | Standard Trailer | Y | | | |

### 6.5.9 Market Data Request Reject (Msg Type = Y)

Used by HF 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<br>1 = Duplicate MDReqID<br>4 = Unsupported SubscriptionRequestType<br>5 = Unsupported MarketDepth<br>6 = Unsupported MDUpdateType<br>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 | | | |

### 6.5.10 Market Data Snapshot Full Refresh (Msg Type = W)

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

| Tag | Name | Req | Valid values | Format | Description |
|-----|------|-----|--------------|--------|-------------|
| | Standard Header | Y | MsgType = W | | |
| 1180 | ApplID | N | | String | Used in conjunction with ApplSeqNum [1181] to indicate, in subsequent connections, the point from which to receive information |
| 1181 | ApplSeqNum | N | | SeqNum | Used in conjunction with ApplID [1180] to indicate, in subsequent connections, the point from which to receive information |
| 262 | MDReqID | N | | String | Identifier of the Market Data Request message that is being replied to |
| 1500 | MDStreamID | N | | String | In case of information about RFQ responses, it contains the corresponding IOIID |
| 1301 | MarketID | N | | Exchange | Operating MIC |
| 1300 | MarketSegmentID | N | | String | Segment MIC |
| | Start <Instrument> | | | | |
| 55 | Symbol | Y | Security code | String(22) | Security code |
| 454 | NoSecurityAltID | N | | NumInGroup | |
| → 455 | SecurityAltID | N | | String | • When SecurityAltIDSource [456] = 4, it contains the ISIN security code |
| → 456 | SecurityAltIDSource | N | 4 = ISIN number | String | |
| 864 | NoEvents | N | | NumInGroup | May be present in a trade or in settlement prices |
| → 865 | EventType | N | 201 = Original trade type (in a countertrade or trade amendment case) | Int | |

| Tag | Name | Req | Valid values | Format | Description |
|-----|------|-----|--------------|--------|-------------|
| | | | 204 = Trade registration number of the countertrade or trade amendment. In a trade corresponding to legs of a strategy it contains the Trade registration number of the trade in the strategy | | |
| | | | 205 = Price of the trade in the case where it does not change the last price | | |
| | | | 206 = Origin of the trade | | |
| | | | 211 = Transaction category MMT:<br>• Package Trade (excluding Exchange for Physicals) "TPAC"<br>• Exchange for Physicals Trade "XFPH" | | |
| | | | 212 = Pre- and Post-transparency flags | | |
| | | | 213 = Current Forward price | | |

| Tag | Name | Req | Valid values | Format | Description |
|---|---|---|---|---|---|
| | | | 214 = Previous Forward price | | |
| → 867 | EventPx | N | | Price | Present when EventType = 205, 213 or 214 |
| → 868 | EventText | N | | String | Present when EventType [865] = 201, 204, 206, 211, 212<br><br>When EventType [865] = 201 the valid values are:<br>• 0 (for a Market trade type),<br>• TrdSubType [829] (for the rest of the trade types)<br><br>When EventType [865] = 206 the valid values are:<br>For trades originated from orders:<br>1 (the trade comes from the Continuous Trading),<br>2 (the trade comes from an Opening Auction),<br>3 (the trade comes from a Closing Auction),<br>4 (the trade comes from a Volatility Auction),<br>5 (the trade comes from a Manual Auction)<br><br>When EventType [865] = 211: Level 3.1 - Transaction category MMT model. Maybe informed when MDEntryType is 2:<br>• Z = Package Trade (excluding Exchange for Physicals) "TPAC"<br>• Y = Exchange for Physicals Trade "XFPH"<br><br>When EventType [865] = 212: It contains the trade pre-transparency and post-transparency flags accordingly MiFID II directive. Different flags are enclosed by doubled quotes (") and separated by a comma. Maybe informed when MDEntryType is 2 |
| | End \<Instrument\> | | | | |
| 268 | NoMDEntries | Y | | NumInGroup | Number of entries to follow |
| → 269 | MDEntryType | Y | 0 = Bid | Char | Type of information that the present entry contains. |

| Tag | Name | Req | Valid values | Format | Description |
|---|---|---|---|---|---|
| | | | 1 = Offer | | If the values 0 or 1 are present, the message does not contain any of the others |
| | | | 2 = Trade (last) | | |
| | | | 4 = Opening Price | | |
| | | | 6 = Settlement Price | | |
| | | | 7 = Trading Session High Price | | |
| | | | 8 = Trading Session Low Price | | |
| | | | 9 = Trading session VWAP price | | |
| | | | B = Trade Volume (total volume for security in session) | | |
| | | | C = Open Interest | | |
| | | | E = Estimated buy mid-price (RFQ) | | |
| | | | F = Estimated sell mid-price (RFQ) | | |
| | | | M = Prior Settle Price | | |
| | | | N = Session High Bid | | |
| | | | O = Session Low Offer | | |
| → 270 | MDEntryPx | N | | Price | Price. Present when the MDEntryType is (0,1,2,4,6,7,8,9,E,F,M,N,O). |

| Tag | Name | Req | Valid values | Format | Description |
|---|---|---|---|---|---|
| | | | | | When it is not present and MDEntryType is 2, see EventPx [867] when EventType [865] = 205 |
| | | | | | When it is not present and MDEntryType is 6, it should be considered as a value 0 |
| | | | | | When MDEntryType = 0 or 1 and there are only Market orders: MDEntryPx [270] = 0 (MDEntrySize [271] > 0) |
| → 271 | MDEntrySize | N | | Qty | Volume. Present when the MDEntryType is (0,1,2,B,C,E,F) For value "C", it contains the open interest at the beginning of the trading session. |
| → 273 | MDEntryTime | N | | UTCTimeOnly | Time of Market Data entry for MDEntryType [269] = 0 (Bid), 1 (Offer), E (Estimated buy mid-price - RFQ), F (Estimated sell mid-price - RFQ) or 6 (Settlement Price) When MDEntryType [269] = 0 (Bid) or 1 (Offer), It is only present for one of the values (MDPriceLevel = 1) and it refers to the update of Bid and Offer in general. |
| → 336 | TradingSessionID | N | 100 = IBEX futures hours / FX<br><br>102 = Cross trades (IBEX futures hours)<br><br>105 = Normal hours<br><br>106 = Delta and Basis Trade<br><br>107 = Bono hours<br><br>108 = Cross trades (normal hours) | String | Trading mode Present when MDEntryType = 0,1,E,F and also when MDEntryType =2 (Trade) with TrdMatchID [880] informed |

| Tag | Name | Req | Valid values | Format | Description |
|---|---|---|---|---|---|
| | | | 109 = Cross trades (Bono hours) | | |
| | | | 115 = RFQ (IBEX futures hours) | | |
| | | | 116 = RFQ (normal hours) | | |
| | | | 117 = RFQ (Bono hours) | | |
| | | | 118 = xRolling on Stocks | | |
| → 277 | TradeCondition | N | 6 = Benchmark Trade "BENC" | MultipleStringValue | Level 3.5 - Benchmark or Reference Price Indicator indicator MMT model<br><br>Maybe present if MDEntryType is 2 |
| | Start \<TradePriceConditionGrp\> | | | | |
| → 1838 | NoTradePriceConditions | N | | NumInGroup | |
| →→ 1839 | TradePriceCondition | N | 15 = Non-Price Forming Trade (formerly defined as a Technical Trade) "NPFT" | Int | Level 3.8 - Ordinary/Standard Trades or Trades Outside Price Formation / Discovery Process MMT model<br><br>Maybe present if MDEntryType is 2 |
| | End \<TradePriceConditionGrp\> | | | | |
| → 2667 | AlgorithmicTradeIndicator | N | 0 = No Algorithmic trade<br><br>1 = Algorithmic trade "ALGO" | MultipleStringValue | Level 3.9 - Algorithmic Indicator MMT model<br><br>Maybe present if MDEntryType is 2 |
| → 346 | NumberOfOrders | N | | Int | When MDEntryType = 0 or 1 indicates the number of orders at this price |
| → 1023 | MDPriceLevel | N | | Int | Level of a bid or offer at a given price level. Numbered from the most to the least competitive per market side, starting with 1. Present if MDEntryType is 0 or 1 |
| → 1024 | MDOriginType | N | 0 = Central Limit Order Book | Int | Level 1 - Market Mechanism MMT model |

| Tag | Name | Req | Valid values | Format | Description |
|---|---|---|---|---|---|
| | | | | | Maybe present if MDEntryType is 2 |
| | | | 1 = Off Book (including Voice or Messaging Trading) | | |
| | | | 3 = Quote Driven Market | | |
| | | | 4 = Dark Order Book | | |
| | | | 5 = Periodic Auction | | |
| | | | 6 = Request for Quotes | | |
| → 811 | PriceDelta | N | | float | Maybe present if MDEntryType = 6 or M |
| → 828 | TrdType | N | See table 4 in document "Codification Tables" | Int | Trade Type. Maybe present if MDEntryType is 2.<br><br>This value is used in conjunction with TrdSubType [829] |
| → 829 | TrdSubType | N | See table 4 in document "Codification Tables" | Int | Maybe present if MDEntryType is 2.<br><br>This value is used in conjunction with TrdType [828] |
| → 1934 | RegulatoryReportType | N | 11 = Limited details trade "LMTF"<br><br>12 = Daily aggregated trade "DATF"<br><br>13 = Volume omission trade "VOLO"<br><br>14 = Four weeks aggregation trade "FWAF"<br><br>15 = Indefinite aggregation trade "IDAF" | Int | Level 4.2 - Post-Trade deferral or Enrichment MMT model<br><br>Maybe present if MDEntryType is 2 |

| Tag | Name | Req | Valid values | Format | Description |
|---|---|---|---|---|---|
| | | | 16 = Volume omission trade. Eligible for subsequent enrichment in aggregated form "VOLW" | | |
| | | | 17 = Full details of previous LMTF "FULF" | | |
| | | | 18 = Full details of previous DATF "FULA" | | |
| | | | 19 = Full details of previous VOLO "FULV" | | |
| | | | 20 = Full details of previous FWAF "FULJ" | | |
| | | | 21 = Full details of previous VOLW "COAJ" | | |
| → 1390 | TradePublishIndicator | N | 1 = Immediate Publication<br><br>2 = Non-Immediate Publication | Int | Level 4.1 - Publication Mode / Post-Trade Deferral Reason MMT model (see also TrdRegPublicationType [2669] + TrdRegPublicationReason [2670])<br><br>Maybe present if MDEntryType is 2 |
| | Start <TrdRegPublicationGrp> | | | | |
| → 2668 | NoTrdRegPublications | N | | NumInGroup | |
| →→ 2669 | TrdRegPublicationType | N | 0 = Pre-trade transparency waiver<br><br>1 = Post-trade deferral | Int | Value 0: Level 3.5 - Benchmark or Reference Price Indicator indicator MMT model (see also TrdRegPublicationReason [2670])<br><br>Value 1: Level 4.1 - Publication Mode / Post-Trade Deferral Reason MMT model (see also TradePublishIndicator [1390] + TrdRegPublicationReason [2670])<br><br>Maybe present if MDEntryType is 2 |

| Tag | Name | Req | Valid values | Format | Description |
|---|---|---|---|---|---|
| →→ 2670 | TrdRegPublicationReason | N | 6 = Non-Immediate Publication: Deferral for "Large in Scale" "LRGS"<br><br>7 = Non-Immediate Publication: Deferral for "Illiquid Instrument" (RTS 2 only) "ILQD"<br><br>8 = Non-Immediate Publication: Deferral for "Size Specific" (RTS 2 only) "SIZE" | Int | Values 6, 7 and 8: Level 4.1 - Publication Mode / Post-Trade Deferral Reason MMT model (related to TradePublishIndicator [1390] = 2 and TrdRegPublicationType [2669] = 1)<br><br>Maybe present if MDEntryType is 2 |
| | End <TrdRegPublication Grp> | | | | |
| → 1188 * | Volatility | N | | float | Maybe present if MDEntryType = 6 or M |
| → 381* | GrossTradeAmt | N | | Amt | Effective trade amount.<br><br>Maybe present when MDEntryType is 2 or B |
| → 880* | TrdMatchID | N | | String | Trade registration number. Identifier of partial fill or filled order, assigned by central system.<br><br>Maybe present when MDEntryType = 2 |
| | Start <TrdRegTimestamps> | | | | Maybe present when MDEntryType = 2 |
| → 768* | NoTrdRegTimestamps | N | | NumInGroup | |
| →→ 769* | TrdRegTimestamp | N | | UTCTimestamp | • When TrdRegTimestampType [770] = 1, it contains the trade execution date and time<br><br>• When TrdRegTimestampType [770] = 11, it contains the date and time publicly reported of the trade |

| Tag | Name | Req | Valid values | Format | Description |
|---|---|---|---|---|---|
| →→ 770* | TrdRegTimestampType | N | 1 = Execution time<br><br>11 = Publicly reported | Int | |
| | End < TrdRegTimestamps > | | | | |
| | Standard Trailer | Y | | | |

# 7 RFQ and Indication of Interest

## 7.1 Introduction

The RFQ functionality allows HF MEFFGate Gate clients to receive information about the RFQ entered to the central systems of MEFF

A client only receives information on the RFQ for those contracts on which it has requested price information (Bid or Offer) in the Market Data Request message

On the other hand, there is the possibility of making an Indication of Interest to the market prior to the introduction of orders in the opposite direction, with the intention of matching a previously agreed operation. This operation will be carried out through the Indication of Interest message.

## 7.2 List of messages

| Message | Description |
| --- | --- |
| Indication of Interest (Msg Type = 6) | Message sent by HF MEFFGate to inform about different RFQ in a security or the Indications of Interest to the market |
| Market Data Snapshot Full Refresh (Msg Type = W) | Message sent by HF MEFFGate to inform about the RFQ prices in a security |

## 7.3 Message flow

**Reception of RFQ / Indication of Interest to the market**



HF MEFFGate Client                                          HF MEFFGate Server

**Market Data Request ("V")**
MDEntryType [269] = 0 (Bid) and 1 (Offer)

**Indication of Interest ("6")**
Symbol [55] = 123456, IOIID [23] = AA, IOITransType [28] = N, Side [54] = 1, IOIQty [27] = 15000

**Market Data - Snapshot / Full refresh ("W")**
MDEntryType [269] = 0, 1, E or F, ...

**Indication of Interest ("6")**
Symbol [55] = 123456, IOIID [23] = AA, IOITransType [28] = R, Side [54] = 1, IOIQty [27] = 13000

**Market Data - Snapshot / Full refresh ("W")**
MDEntryType [269] = 0, 1, E or F, ...

**Indication of Interest ("6")**
Symbol [55] = 123456, IOIID [23] = AA, IOITransType [28] = C, Side [54] = 1, IOIQty [27] = 0

## 7.4 Annotations and adaptations of FIX 5.0

The tag TradingSessionID [336] has been added to message Indication of Interest

## 7.5 Definition of messages

### 7.5.1 Indication of Interest (Msg Type = 6)

Message sent by HF MEFFGate to notify an indication of interest on a specific contract.

| Tag | Name | Req | Valid values | Format | Description |
|-----|------|-----|--------------|--------|-------------|
| | Standard Header | Y | MsgType = 6 | | |
| 1180 | ApplID | N | | String | Used in conjunction with ApplSeqNum [1181] to indicate, in subsequent connections, the point from which to receive information |
| 1181 | ApplSeqNum | N | | SeqNum | Used in conjunction with ApplID [1180] to indicate, in subsequent connections, the point from which to receive information |
| 23 | IOIID | Y | | String | If the message comes from an RFQ, it will contain the RFQ identifier message<br>If the message comes from an Indication of Interest, it will contain the Identifier of the subscription made by the user |
| 28 | IOITransType | Y | N = New<br><br>C = Cancel<br><br>R = Replace | Char | |
| | Start <Instrument> | | | | |
| 55 | Symbol | Y | Contract code | String(22) | Contract code |
| 454 | NoSecurityAltID | N | | NumInGroup | |
| → 455 | SecurityAltID | N | | String | |
| → 456 | SecurityAltIDSource | N | 4 = ISIN number | String | When SecurityAltIDSource [456] = 4, it contains the ISIN code for the contract |
| | End <Instrument> | | | | |
| 54 | Side | Y | 1 = Buy<br><br>2 = Sell<br><br>7 = Undisclosed | Char | |
| 27 | IOIQty | Y | | String | RFQ volume requested |
| 44 | Price | N | | Price | RFQ price requested |
| 25 | IOIQltyInd | N | H = High (RFQ requested)<br><br>M = Medium (Order requested) | Char | Request Type<br><br>Value "H" indicates RFQ has been requested through new trading modes 115, 116, 117 or 118, for responses addressed to the petitioner. |

| Tag | Name | Req | Valid values | Format | Description |
|-----|------|-----|--------------|--------|-------------|
| | | | | | Value "M" indicates RFQ is asking for price quotations in the order book, addressed to all market participants. |
| 60 | TransactTime | N | | UTCTimeStamp | Event time |
| 336* | TradingSessionID | N | 100 = IBEX futures hours / FX<br><br>105 = Normal hours<br><br>107 = Bono hours<br><br>115 = RFQ (IBEX futures hours)<br><br>116 = RFQ (normal hours)<br><br>117 = RFQ (Bono hours)<br><br>118 = xRolling on Stocks | String | Trading mode |
| | Standard Trailer | Y | | | |

# 8 Communication of Events

## 8.1 Introduction

The News message is used to receive information from the Market Supervisor.

The information received has a free text format.

## 8.2 List of messages

| Message | Description |
|---|---|
| News (Msg Type = B) | Used to receive text messages from the market supervisor |

## 8.3 Message flow

**Message reception**

HF MEFFGate Client            HF MEFFGate Server

**News ("B")**

## 8.4 Annotations and adaptations of FIX 5.0

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

## 8.5 Definition of messages

### 8.5.1 News (Msg Type = B)

| Tag | Name | Req | Valid values | Format | Description |
|---|---|---|---|---|---|
| | *Standard Header* | Y | MsgType = B | | |
| 1180 | ApplID | N | | String | Used in conjunction with ApplSeqNum [1181] to indicate, in subsequent connections, the point from which to receive information |
| 1181 | ApplSeqNum | N | | SeqNum | Used in conjunction with ApplID [1180] to indicate, in subsequent connections, the point from which to receive information |
| 42 | OrigTime | N | | UTCTimeStamp | Event time |
| 61 | Urgency | N | 0 = Normal<br>1 = Flash<br>2 = Background | Char | The default value is 0 |
| 148 | Headline | Y | | String | Message header. Ignored by HF 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 | | | |

# User Fields

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

| Tag | Name | Format | Description |
|---|---|---|---|
| 21500 | MoreSubscriptionsFollowing | Boolean | Allows to group market information subscription requests.<br><br>For more information see "3.6 - Synchronisation at application level" |
| 21501 | LocalMktTimestamp | String | Indicates, for all tags in which a timestamp is included, the timestamp format:<br><br>Y – HF MEFFGate will send the local market time (all messages up to microseconds)<br><br>N – HF MEFFGate will send the the time in UTC format according to the FIX standard (all messages up to microseconds)<br><br>For more information see 4.5 |
| 21505 | BusinessSessionDate | LocalMktDate | Current business session date. |

BME
Plaza de la Lealtad,1
Palacio de la Bolsa
28014 Madrid

www.bolsasymercados.es