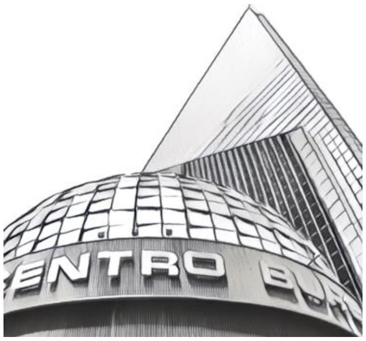
MexFix®

Fix Clients – MexFix Market Data

Layout FIX

External Interface Specification (EIS)

Version	Date
1.0	Dec, 2016



Revisions

DOCUMENT REVISION HISTORY

Version	Author(s)	Reason For Issue	Date
1.0	Jose Alberto Gonzalez – DGA Tecnología	Initial version	Dec, 2016

COMMENTS AND SUGGESTIONS

Please forward any comment or suggestion *to ftsc@grupobmv.com.mx*

Information contained in this document is subject to change without prior notice. The companies, names and data used in the examples are not real, unless otherwise stated. You may not reproduce or transmit any part of this document in any way, nor by any means, either electronic or mechanic, for any purpose, without MexDer express written consent.

Index

1. Introduction

1.1 Purpose

The present document quotes the specification and flow of messages of the interface that is relevant to the functionality of MexFix Market Data. This solution is based on the FIX (Financial Information eXchange) protocol standard, version 4.4.

1.2 Target

This document is addressed to those members willing to receive MexDer market information through FIX using MexFix Market Data protocol.

1.3 Conventions

Any message related to the solution is duly specified in its entirety. Certain MexFix aspects differ from the FIX standard; therefore, tables specifying the tags contained in each message include information like the following:

Column	Meaning
Tag	Field Number. Fields added to the message by MexFix show an asterisk ("*") following this number.
Name	Field name according to FIX standard.
Req	"S" means the field is required; "N" means the field is optional. "S*" means the field is required in MexFix implementation, but optional in FIX standard 4.4.
Valid values	Field valid values in the context of the message. It may be a list of values or a range of numeric values, for instance ">=3, <= 10". This column also contains the field default value, for the optional fields that may require it. To avoid confusions with the terms, in the values related to codes the FIX original value has been kept and therefore it has not been translated.
Format	Type of field data. This is one of the types defined by FIX, or one of those 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. If you need more information on the String type please refer to point 2.4.
Description	Field description in the context of the message.

1.4 Related Documents

#	Title	Author	Version
1	Financial Information Exchange Protocol (FIX) 4.4 with errata 20030618	FIX Committee	June 18, 2003
2	Layout MexFix EIS English version 2 3	MexDer	Oct 2010

2. Implementation

2.1 Description

Given the fact that some modifications were made to FIX standard, the limitations implemented are detailed below:

2.2 Ignored Fields

In some cases, MexFix may ignore the content of some fields of the input messages. If this is the case, it is clearly explained in the field description.

2.3 Unsupported Fields

Unsupported fields in a message have not been included in its description.

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

No required field whatsoever has been declared unsupported.

2.4 String Type Length

FIX standard imposes no maximum length restriction on the String type. In the implementation carried out by MexFix, the maximum length for this type has been set to 255 characters.

In some fields, a lower maximum length of this value has been set. In these cases, the type is presented as String(n), where “n” is the maximum number of field characters. Sometimes “n” is the exact length of the field, in such case, it shall be explicitly mentioned in the “Valid Values" column.

2.5 Message Maximum Length

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

2.6 Encryption

MexFix does not use the encryption defined by FIX standard (by means of SecureData and SecureDataLen fields in the heading of the message). Encryption is implemented by using SSL (Secure Socket Layer).

2.7 FIX Protocol

FIX protocol’s version 4.4 is the only one that may be used with the implementation of MexFix.

2.8 Changes from the previous version

Trading and Market Data MexFix specifications were splitted in different and separated documents. The present document describes the MexFix Market Data Specification.

The criteria for requesting data were limited to selection through CFI Code [461] tag or a specific Instrument using Symbol [55] tag.

Since no longer being part of criteria selection, SecurityID[48] and MaturityMonthYear[200] tags were eliminated from messages: Security List Request [x], Security Status Request [e] and Market Data Request [V].

CFI Code for SWAPS was taken from new ISO 10962:2015 standard revision.

3. FIX Session

MexFix complies with the specifications in FIX standard 4.4 as far as Session level is concerned. In this way, this paragraph will show only the new messages layout due to certain MexFix delimitations. Some other differences, arising out of the protocol adaptation made by MexFix, will be shown afterwards. To find more details as to the way in which sessions are established and aspects related to FIX session level, it is necessary to refer to the relevant FIX documentation which is not rewritten in order to avoid any doubt about it.

3.1 Synchronize at an Application level

When a member stars a session (accepted Logon message), it receives a group of information related with the current Market Session (if available): receives all the messages that are not associated to the respective subscriptions of the current market session and that it have not received previously.

Messages coming from an explicit request of repetition (requested with a Resend Request message, as shown in section 3.2.7, will contain the "Y" value in the field PossDupFlag stating such situation.

Take into account that any subscription to information is cancelled at the end of the session. If when reconnecting a FIX session such service is desirable, it should be requested again.

3.2 List of Messages

3.2.1 Standard Message Header

Header that contains the FIX messages

Tag	Name	Req	Valid values	Format	Description
8	BeginString	Y	FIX 4.4	String	It indicates the beginning of a new message. It contains the FIX protocol version. It is always the first field of the message.
9	BodyLength	Y		int	Message length in bytes, from the end of this field until, and including, the previous limit character of the CheckSum field. It is always the second field of the message.
35	MsgType	Y	All message types supported by FIX	String	It identifies the type of message. It is always the third field of the message.
49	SenderCompID	Y		String	Identifier of the entity that sends the message. It has to contain the member code in the messages sent by the client's application.
56	TargetCompID	Y		String	Identifier of the entity to whom the message is target. It contains the member code in the messages sent by MexFix.

115	OnBehalfOfCompID	N		String	Ignored
34	MsgSeq Num	Y		int	Sequence number of the message within the current FIX session.
50	SenderSubID	Y*	For further detail regarding market codes, see Table 17 in the “Coding Tables” document.	String	The messages sent by MexFix contain the code assigned to the Market to which the connection was established. In messages sent to MexFix it must contain the trader code with which the FIX session was started.
57	TargetSubID	Y*	For further detail regarding market codes, see Table 17 in the “Coding Tables” document.	String	The messages sent by MexFix contain the trader code to whom it is addressed. In messages sent to MexFix it must contain the Market code with which the connection was established.
116	OnBehalfOfSubID	N		String	Ignored
129	DeliverToSubID	N		String	Ignored
43	PossDupFlag	N	N = Sending of the original message (default value) Y = Possibly duplicated	Boolean	It indicates if it is the first time, within the FIX session, that a message is sent (“N”) or if it is sending again the same message (“Y”), because of an explicit request on the other behalf or because there is a doubt about the reception of the original message.
52	SendingTime	Y		UTC Timestamp	Sending time of the message.
122	OrigSendingTime	N		UTC Timestamp	Sending time of the original message. Required in a resending. A message is considered a resending if the field PossDupFlag = “Y” and if the MsgType field is not a “4” (SequenceReset).

3.2.2 Standard Message Trailer

Final part of all FIX messages.

Tag	Name	Req	Valid values	Format	Description
10	CheckSum	Y		String(3)	Message Checksum, computed as it is described in the standards. It is always the last field of the message and its length is 3 bytes accurately.

3.2.3 Logon (Msg Type = A)

The Logon message is used by the client to start a session and to be accepted by the server.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = A		
98	EncryptMethod	Y	0 = None	int	Ignored by MexFix
108	HeartBtInt	Y	Assigned by MexDer	int	Sending interval of the connection verification message (Heartbeat message) expressed in seconds.
141	ResetSeqNumFlag	N	N	Boolean	Only the value “N” is allowed, since in the protocol implementation it is not necessary.
553	Username	Y*		String	User identifier assigned by MexDer.
554	Password	Y*		String	Password assigned by MexDer.
	Standard Trailer	Y			

3.2.4 Logout (Msg Type = 5)

The Logout message is used by both parties whether to request or notify the end of the communication session or to accept such request.

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

3.2.5 Heartbeat (Msg Type = 0)

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

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = 0		
112	TestReqID	N		String	If the message is the response to a Test Request message, it may contain the same value that the original TestReqID field contained.

	Standard Trailer	Y			

3.2.6 Test Request (Msg Type = 1)

The Test Request message is used by both parties to request the sending of the Heartbeat message.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = 1		
112	TestReqID	Y		String	Petition identifier. It has to be included in the answer Heartbeat message.
	Standard Trailer	Y			

3.2.7 Resend Request (Msg Type = 2)

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

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = 2		
7	BeginSeqNo	Y	Valid sequence number	int	Sequence number of the first message of the rank of messages of which the resent is requested. It must contain a lower value than the last sequence number received.
16	EndSeqNo	Y	0 = Infinite number of valid sequence	int	Sequence number of the last message of the rank of messages of which the resent is requested. It must contain a lower value than the last number of the received sequence. If the request is only of one message EndSeqNo = BeginSeqNo. If the request is of all messages from a given one EndSeqNo = 0
	Standard Trailer	Y			

3.2.8 Sequence Reset (Msg Type = 4)

The Sequence reset message is used by both parties to fill in the blank spaces in the messages that are being sent, through the re-assignation of the sequence number.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = 4		Have in mind that PossDupFlag must contain the "Y" value
123	GapFillFlag	Y*	Y = It indicates that the message is to fill in a blank space	Boolean	For further information consult the document with the FIX 4.4 specifications
36	NewSeqNo	Y		int	Sequence number of the message which will be sent
	Standard Trailer	Y			

3.2.9 Reject (Msg Type = 3)

The Reject message is used by MexFix for rejecting a message that does not comply with the FIX protocol specified in this document.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = 3		
45	RefSeqNum	Y		int	Sequence number of the rejected message

373	SessionRejectReason	N	0 Invalid tag number 1 Required tag missing 2 Tag not defined for this message type 3 Undefined Tag 4 Tag specified without a value 5 Value is incorrect (out of range) for this tag 6 Incorrect data format for value 9 CompID problem 11 Invalid MsgType 13 Tag appears more than once 14 Tag specified out of required order 15 Repeating group fields out of order 16 Incorrect NumInGroup count for repeating group 17 Non "data" value includes field delimiter (SOH character) 99 Other	int	It is the code that indicates the reason of rejection
58	Text	N		String	It contains a more specific description of the reason of rejection
	Standard Trailer	Y			

3.3 FIX 4.4 Delimitations and Adaptations

- When a request to start a session (Logon message) is rejected, MexFix will always send a Logout message as an answer.
- Fields **SenderSubID** and **TargetSubID** in the heading of messages (Standard Message Header) are no longer optional but required.
- PossResend field is not supported.
- FIX encryption method is not supported.
- Valid values of Reset SeqNumFlag field of Logon messages are limited to the value "N".

- **Username** and **Password** fields are no longer optional but required.
- The field NextExpectedMsgSeqNum (789) of the Logon message (A) was eliminated. The sequence synchronization mechanism must be executed through the Resend Request message (“2”) as was always made by the FIX protocol.
- MexDer recommends periods of 30 seconds for sending a heartbeat
- MexFix has a successive Logon rejection mechanism for the same Member, in other words, after receiving a Logon, MexFix stars the Logon process and will reject eventual new connection messages during a term of 1 (one) minute.

3.4 Identification of the FIX session

Once a communication session has been established, MEXFI 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 market.

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.

4. General Conventions for Application Messages

4.1 Instrument Block

In some requests, FIX clients may specify contract selection criteria. In such cases, only the information related to the contracts matching these criteria is returned. Possible selection criteria correspond to the Instrument block fields.

In the following tables, the fields accepted by MexFix are indicated and the type of request in which they may intervene.

Field	Meaning	Contract information request	Price information request
CFICode	Financial instrument coding according to standard ISO 10962	X	X
Symbol	Contract code	-	X

The following subparagraphs explain in detail the use of these fields.

4.1.1 CFICode

CFICode field is the less selective criteria. It represents a type of contracts, for instance all futures or all options. The following table lists the values allowed for CFICode field as selection criteria. Please bear in mind that the values should contain 6 characters exactly.

CFICode	Meaning
XXXXXX	All supported contracts
FXXXSX	All futures
OXXXXS	All options
OCXXXS	All call options
OPXXXS	All put options
SXXXXX	Swaps
XMXXXX	All time-spread contracts (Multileg)

When MexFix fills in the CFICode field of a SecurityList message in order to describe the assets referred to, the level of details may be higher than the one described in the table above. The list of values used by MexFix is shown in Table 16 of the “Coding Tables” document. If you need more information on SecurityList messages please refer to Chapter 6 “Market Information”.

******* To identify “strategies” and “time spreads” just take in account the second character on the CFICode which is “M”

4.1.2 Contract Code (Symbol field)

This is the most selective criteria since it refers to a specific contract. It may only be used in the contract information request. When you do not want to specify a particular contract and use the remaining criteria, this field should be filled in with the value "[N/A]" as indicated in the FIX standard specifications. This is a mandatory field in the Instrument block.

5. Generic Messages at the Application Level

5.1 Communication Status

This functionality is not supported by MexFix.

5.2 Rejection of Application Messages

When MexFix receives a supported and syntactically correct message, in an unsupported situation, and there is no specific rejection message, the generic Business Message Reject message is used

This message may also be sent by the Member.

5.3 List of Messages

Message	Description
Business Message Reject (MsgType = j)	Message rejection at application level (used if there is no specific message)

5.4 Definition of Messages

5.4.1 Business Message Reject (MsgType = j)

The message sent by MexFix when receiving a supported and syntactically correct message in an unsupported situation, as to where there is no specific rejection message.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = j		
45	RefSeqNum	Y		int	MsgSeqNum of the rejected message.
372	RefMsgType	Y	.	String	MsgType of the rejected message.

379	BusinessRejectRefID	N		String	Optional identifier of the rejected message.
380	BusinessRejectReason	Y	3 = Unsupported Message Type	int	Reason for rejection
58	Text	N		String	Explanatory text
	Standard Trailer	Y			

6. Market Information

6.1 Introduction

Several functionalities related to market public information are grouped under the concept of market information. This information is classified in three groups:

- **Session Status.** Trading session status
- **Contract Information.** Selected contract definition and status
- **Prices.** Prices and indication of interest requests of the selected contracts

Each one of these groups is dealt with in a separate paragraph of this chapter.

6.2 Market Information: Trading session status

6.2.1 Description

This functionality allows the client to obtain the trading session status for the market related to the current FIX session, as well as to be notified of the status changes that may occur.

The client may request the current status or the current status plus the status changes that may occur. Please, take into account that there is no method to request the update without requesting the current status. This is of the utmost importance in order to guarantee the integrity of the received information.

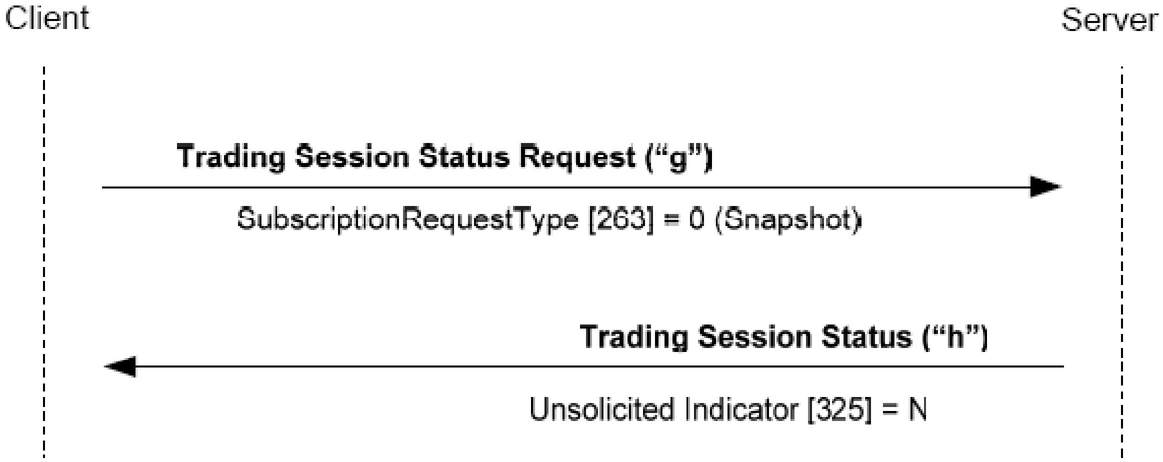
6.2.2 List of Messages

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

6.2.3 Flow of Messages

6.2.3.1 Session Status Request (without update)

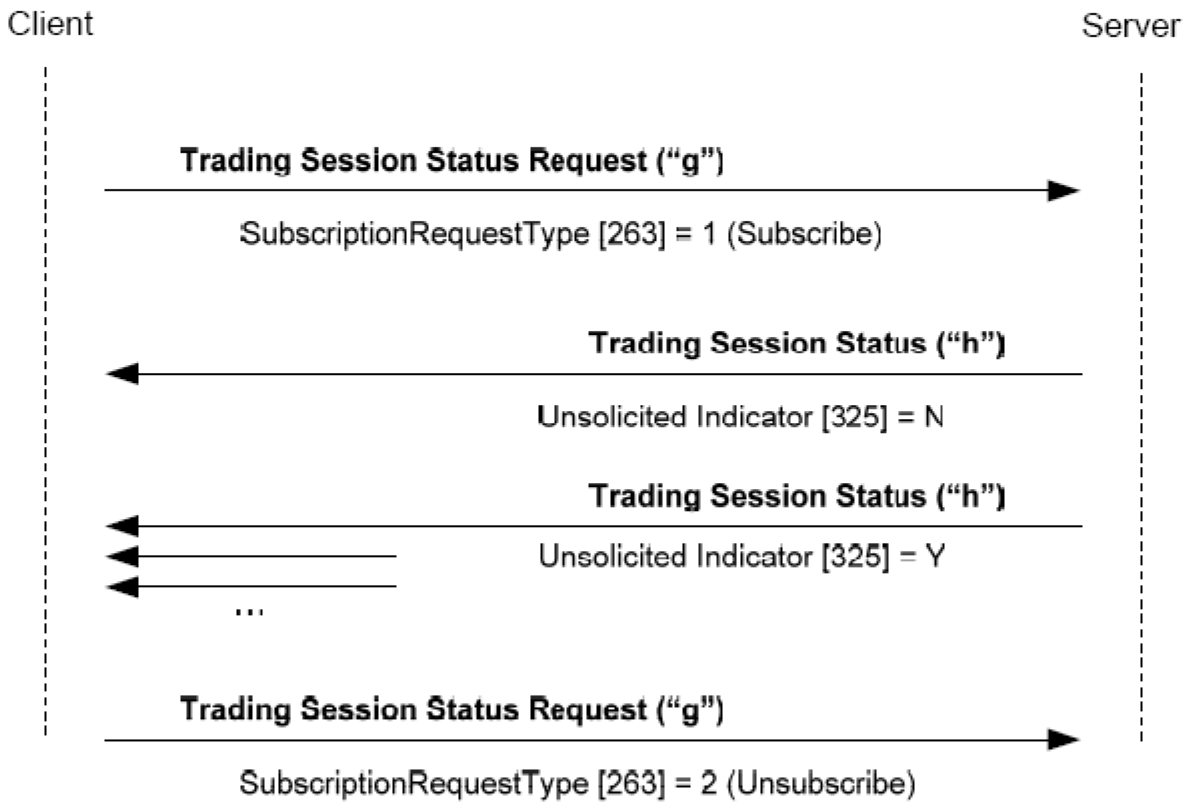
A session status request, without update, is answered by a unique Trading Session Status message.


Client
Server
Trading Session Status Request ("g")
SubscriptionRequestType [263] = 0 (Snapshot)
Trading Session Status ("h")
Unsolicited Indicator [325] = N

6.2.3.2 Session Status Request (with update)

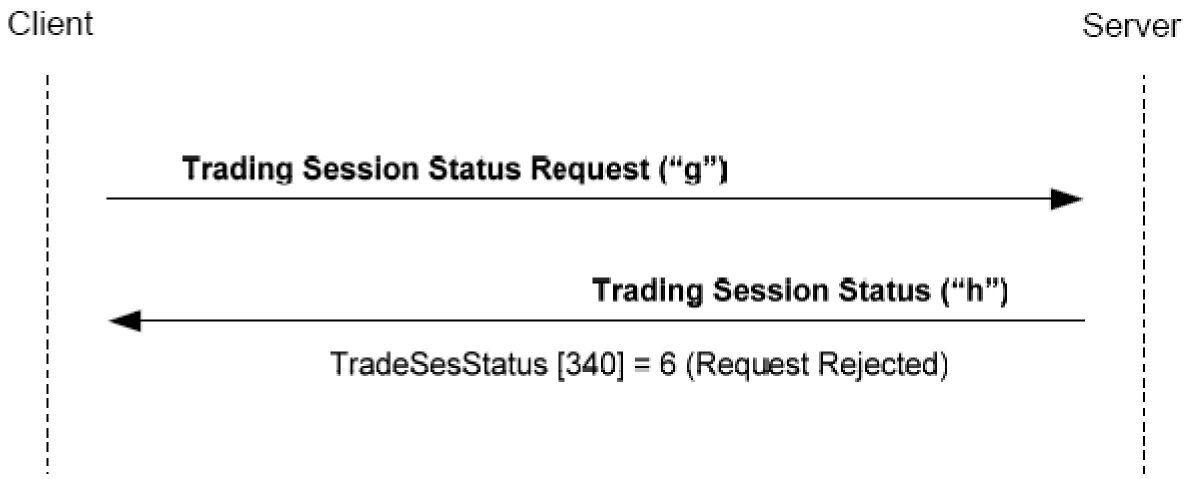
A session status request, with update, is answered with a Trading Session Status message reporting the market actual situation and a new message each time a status change takes place.

Please take into account that an end of subscription request is answered only when rejected.


Client
Server
Trading Session Status Request ("g")
SubscriptionRequestType [263] = 1 (Subscribe)
Trading Session Status ("h")
Unsolicited Indicator [325] = N
Trading Session Status ("h")
Unsolicited Indicator [325] = Y
...
Trading Session Status Request ("g")
SubscriptionRequestType [263] = 2 (Unsubscribe)

6.2.3.3 Failed Session Status Request

A failed session status request is answered by a Trading Session Status message with the TradeSesStatus field = 6.


Client
Server
Trading Session Status Request ("g")
Trading Session Status ("h")
TradeSesStatus [340] = 6 (Request Rejected)

6.3 Market Information: Contracts

6.3.1 Description

This functionality allows obtaining the information of market contracts. Data are grouped in two sets:

- **Contract definition.** Static information of contract definition
- **Contract status.** Dynamic information reflecting contract status.

6.3.2 Contract information request

Contract definition request is made with Security List Request and Security Status Request messages, as detailed in the following table:

	Contract Snapshot	Contratct Update	Contract status Snapshot	Contract status Update
Security List Request SubscriptionRequestType = 0	X		X	

Security List Request
SubscriptionRequestType = 0
NewSecuritySubscription = 1
	X	X	X	

Security List Request SubscriptionRequestType = 1	X		X	X

Security List Request
SubscriptionRequestType = 1
NewSecuritySubscription = 1
	X	X	X	X

urity Status Request SubscriptionRequestType = 0			X	
Security Status Request SubscriptionRequestType = 1			X	X

The meaning of each column is detailed below:

- **Contracts – Snapshot.** One or more Security List messages are obtained and contain the description of the available contracts complying with the selection criteria indicated in the request.
- **Contracts – Update.** A Security List message is obtained. It contains the information of a new contract when the latter is entered as an addition into the system. The system only receives information of new contracts that match the selection criteria indicated in the request.
- **Contract Status – Snapshot.** A Security Status message is obtained. It contains the status information of every contract that complies with the selection criteria indicated in the request.

- **Contract Status – Update.** A Security Status message is obtained whenever a status change takes place in one of the contracts that matches the selection criteria indicated in the request. Furthermore, a Security Status message is received when a new contract is entered as an addition and complies with the selection criteria indicated in the request.

It should be noted that the NewSecuritySubscription field has been added by MexFix to the Security List Request message in order to allow the subscription to contract definition contracts created during the session, typically new option and multileg contract strikes.

In order to have available the maximum information for the client application, a Security List Request message with NewSecuritySubscription = 1 and SubscriptionRequestType = 1 is recommended. In this way, you obtain information on all changes that may occur, including contract additions.

6.3.3 Reception of the contract definition

The information on contract definition is received through the Security List message. This message informs on one contract at a time. The TotNoRelatedSym field informs about the total contracts that match the selection criteria and the NoRelatedSym field informs about the number of contracts contained in the message in question.

When requesting the current status plus update, first, all messages related to current status should be sent and once this list is ended the update-related ones should be sent. In this way, we make sure there is no confusion with the list follow-up. Please bear in mind that update messages contain only one contract per message.

In case of strategies with more than 40 legs, the leg information and the SecurityAltID block won’t be sent in the SecurityList message (this is the case of some “Stapled” strategies); nevertheless the contract could be identified as strategy by using the CFICode tag (461).

6.3.4 Reception of the Contract Status

The information on contract status is received through the Security Status message. Each Security Status message contains information about only one contract. Therefore, the answer to a Security Status Request message may be formed by several Security Status messages. In this case, there is no mechanism to know when all information has been duly received. If necessary, the FIX client should request first the list of contracts through a Security List Request message, so as to determine the number of contracts that comply with a certain criteria.

When the update of this information has been requested, when a change in a contract status takes place, a new Security Status message is received containing the new information.

When in the market a contract is entered as an addition, and such contract matches the selection criteria specified in the request, and such request is an update request, a Security Status message containing the contract status is received, regardless of whether the recipient was subscribed or not to the definition of said contract. In this case, client applications should be prepared to receive information on contract status of which they have not received the relevant definition.

The subscription to status update of option contracts and combined options is not recommended unless subscription to contract update is also made.

6.3.5 End of Subscription

In order to end a subscription to contract definition, the Security List Request message with the NewSecuritySubscription field = 2 is used.

When a subscription to contract status has been made through the Security List Request message, the subscription may be ended with the same type of message with the SubscriptionRequestType field = 2.
When a Security List Request message contains at least an end of subscription request (NewSecuritySubscription = 2 or SubscriptionRequestType = 2) the system considers it as a message containing an end of subscription request only, and therefore any other values specified in these two fields are not taken into account.

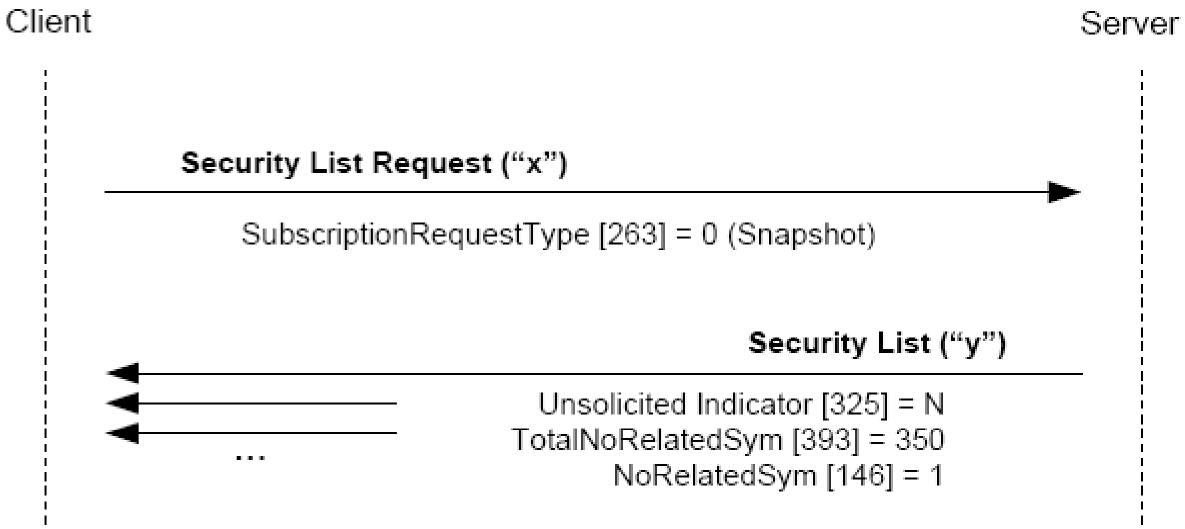
6.3.6 List of Messages

Message	Description
Security List Request (Msg Type = x)	Sent by the client to request contract definition. It also allows requesting the status information of such contracts.
Security List (Msg Type = y)	Sent by the server to inform about contract definition. It is also used to inform on the rejection of the request of such information.
Security Status Request (MsgType = e)	Sent by the client to request contract status.
Security Status (MsgType = f)	Sent by the server to inform about contract status. It is also used to inform about the rejection of such information request of such information, or to indicate there is no contract complying with the selection criteria

6.3.7 Flow of Messages

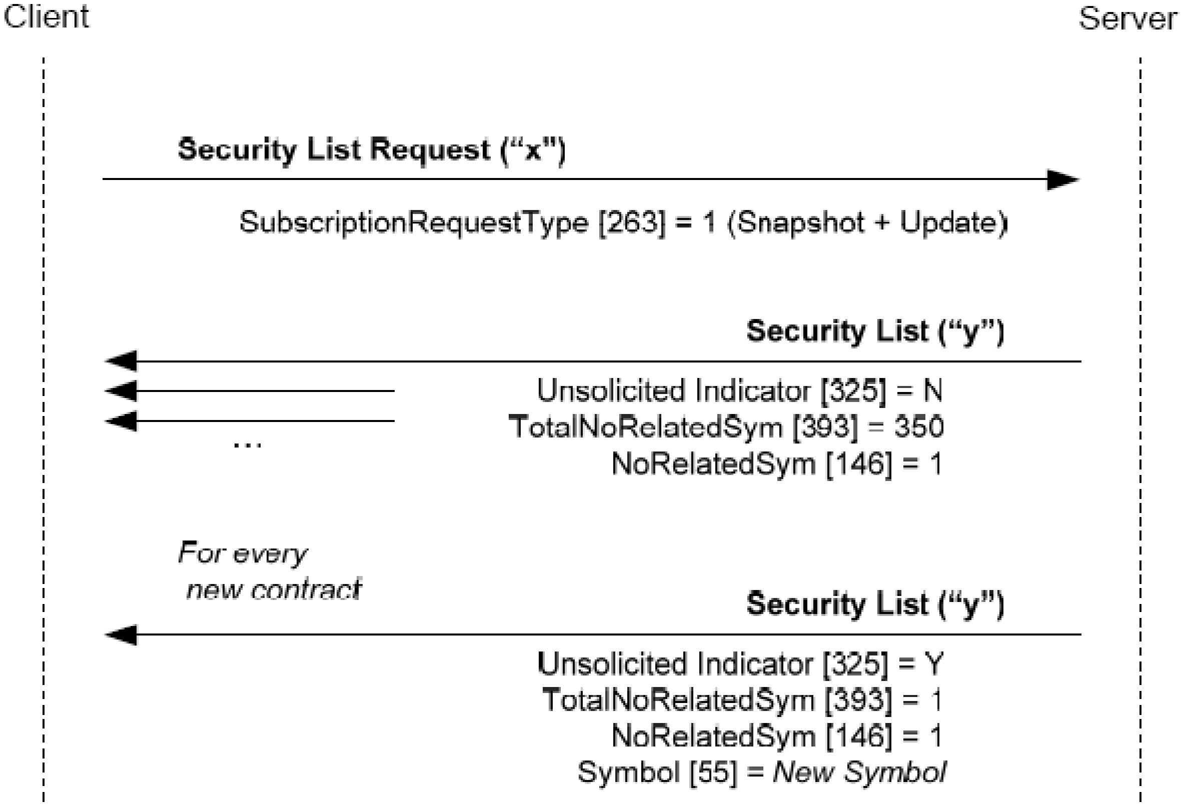
6.3.7.1 Contract definition request without update

After requesting a contract definition one or more Security List messages are received. Each one of these messages indicate the total contracts that match the selection criteria in the TotNoRelatedSym field and the number of contracts contained in the message in question in the NoRelatedSym field.


Client
Server
Security List Request ("x")
SubscriptionRequestType [263] = 0 (Snapshot)
Security List ("y")
Unsolicited Indicator [325] = N
TotalNoRelatedSym [393] = 350
NoRelatedSym [146] = 1
...

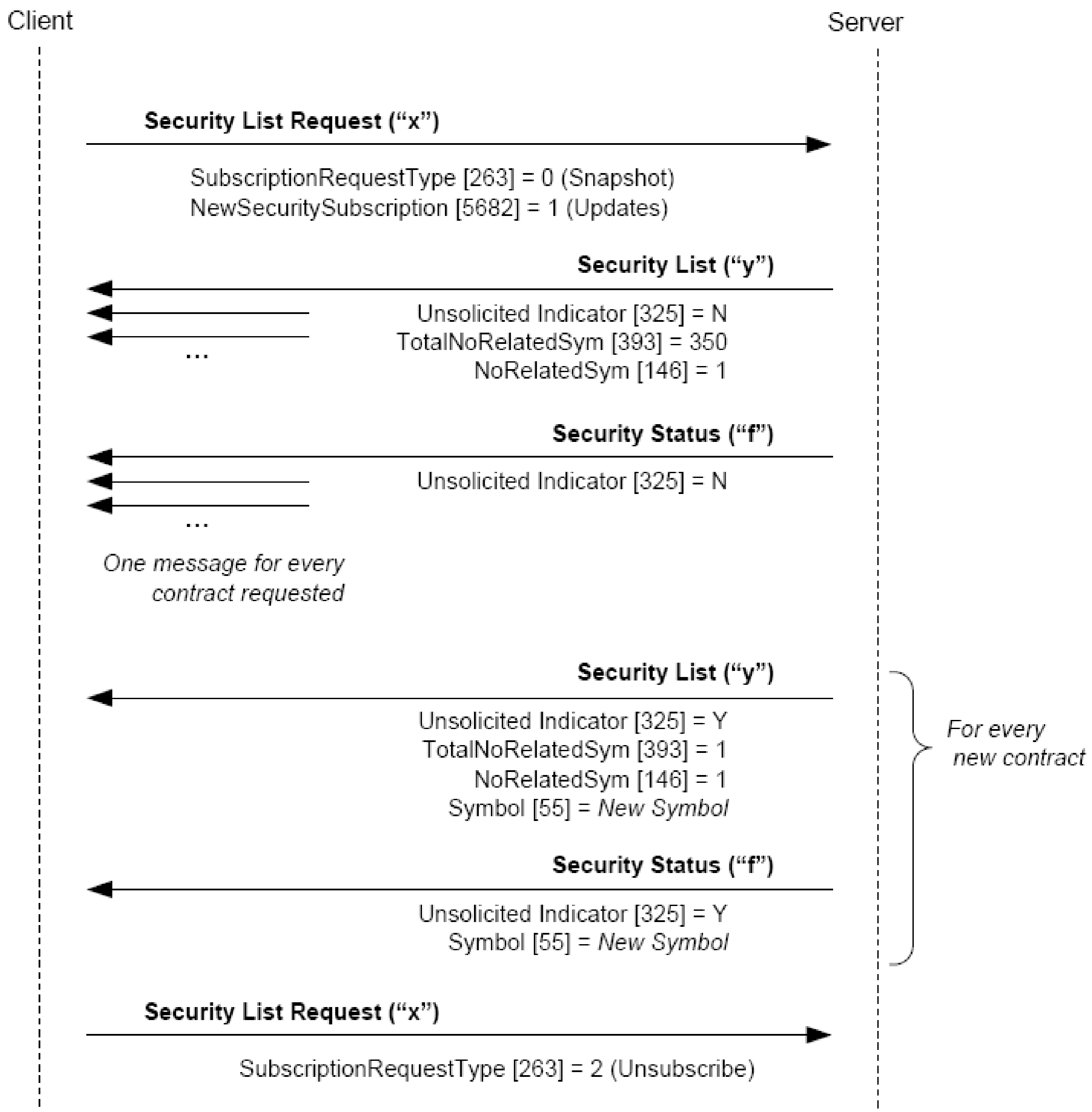
6.3.7.2 Contract definition request with update

When the request of contract definition includes contract updates (NewSecuritySubscription = 1), besides the messages explained in the preceding case, when a new contract is entered as an addition into the system, a Security List message containing this contract information is received.


Client
Server
Security List Request ("x")
SubscriptionRequestType [263] = 1 (Snapshot + Update)
Security List ("y")
Unsolicited Indicator [325] = N
TotalNoRelatedSym [393] = 350
NoRelatedSym [146] = 1
...
For every new contract
Security List ("y")
Unsolicited Indicator [325] = Y
TotalNoRelatedSym [393] = 1
NoRelatedSym [146] = 1
Symbol [55] = New Symbol

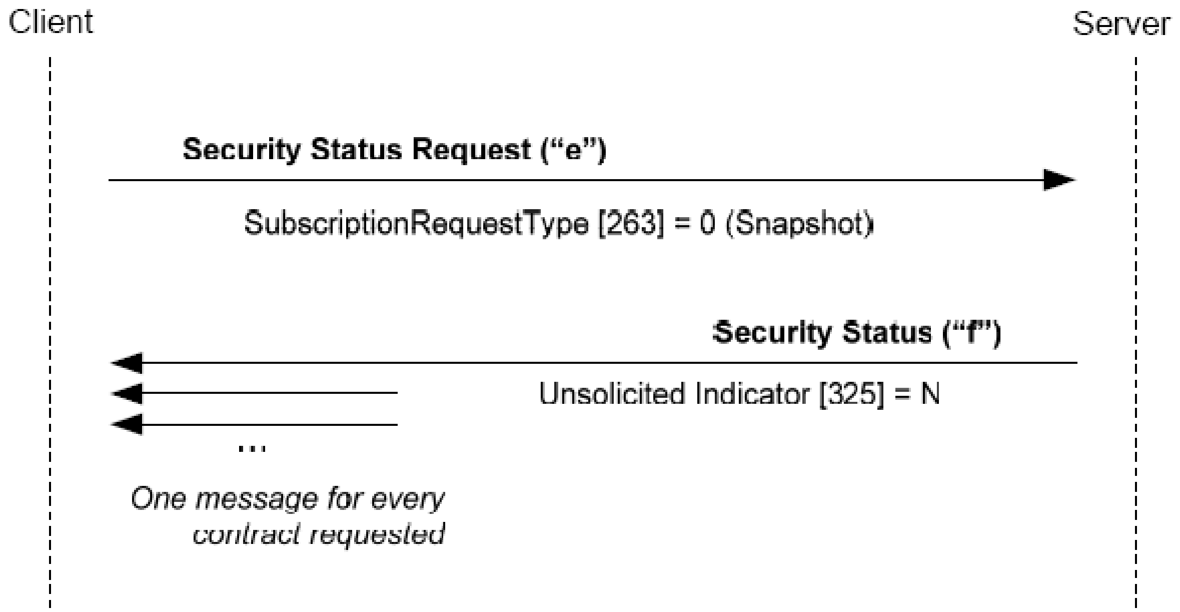
6.3.7.3 Contract definition request with update and contract status with update

If the request includes a contract status request a Security Status message is received. It includes such information for each contract. If the request also includes the update (SubscriptionRequestType = 1), a new message is received each time a status change takes place.


Client
Server
Security List Request (“x”)
SubscriptionRequestType [263] = 0 (Snapshot)
NewSecuritySubscription [5682] = 1 (Updates)
Security List (“y”)
Unsolicited Indicator [325] = N
TotalNoRelatedSym [393] = 350
NoRelatedSym [146] = 1
…
Security Status (“f”)
Unsolicited Indicator [325] = N
…
One message for every contract requested
Security List (“y”)
Unsolicited Indicator [325] = Y
TotalNoRelatedSym [393] = 1
NoRelatedSym [146] = 1
Symbol [55] = New Symbol
Security Status (“f”)
Unsolicited Indicator [325] = Y
Symbol [55] = New Symbol
For every new contract
Security List Request (“x”)
SubscriptionRequestType [263] = 2 (Unsubscribe)

6.3.7.4 Contract status request without update

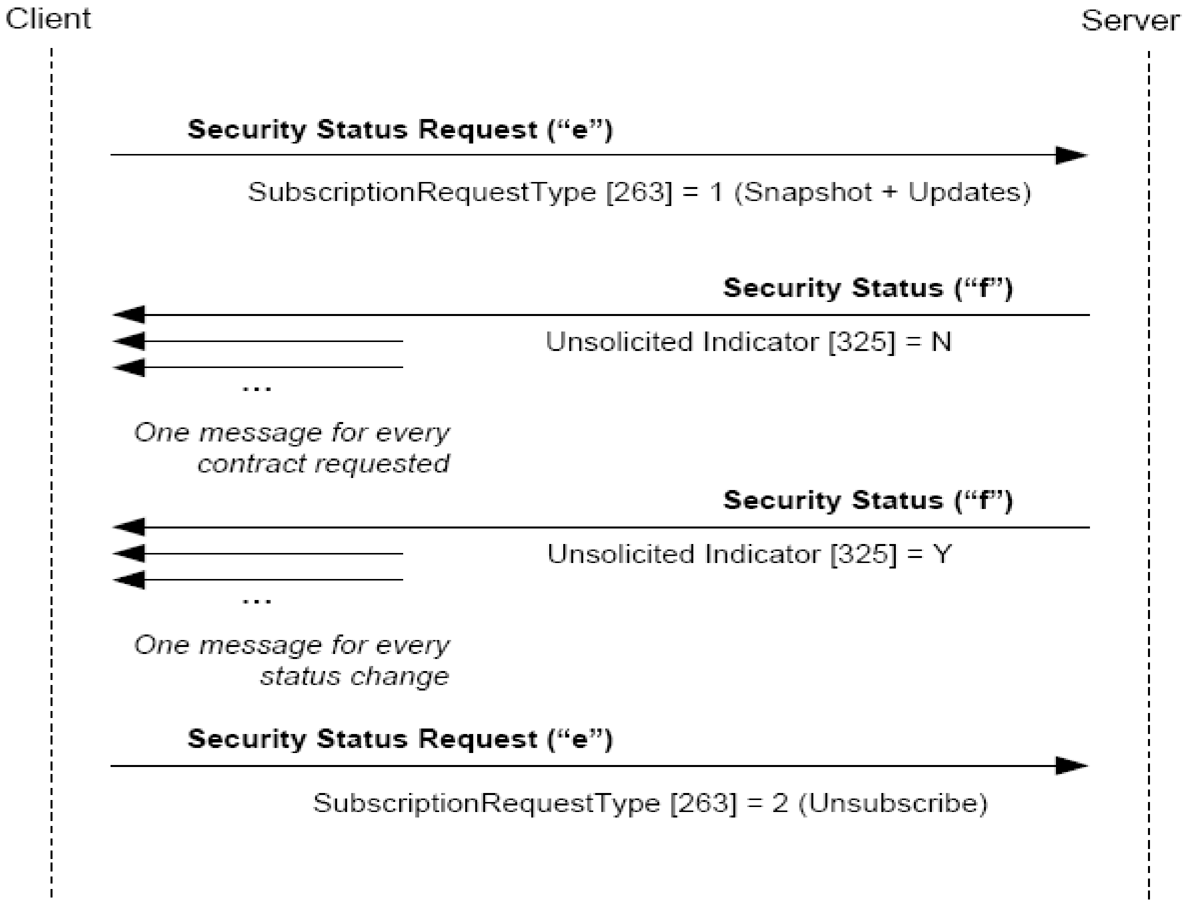
One contract status request is answered with a Security Status message for each one of the contracts that complies with the selection criteria.


Client
Server
Security Status Request ("e")
SubscriptionRequestType [263] = 0 (Snapshot)
Security Status ("f")
Unsolicited Indicator [325] = N
...
One message for every
contract requested

6.3.7.5 Contract status request with update

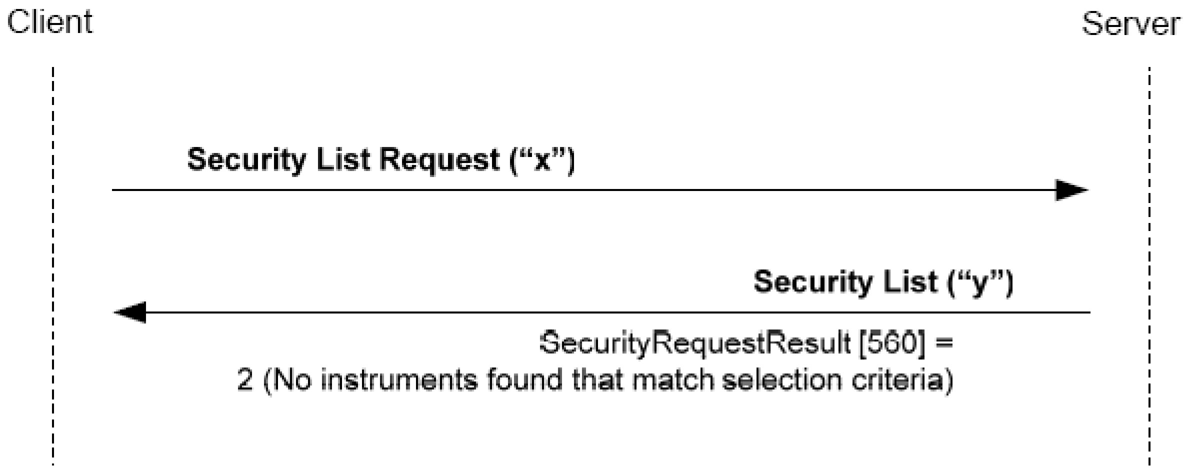
A contract status request with update is initially answered with a Security Status message for each one of the existing contracts that complies with the selection criteria. From this point on, when a status change takes place in some of the contracts, a new Security Status message with the relevant information is received. These last messages contain the value "Y" in the UnsolicitedIndicator field.

Furthermore, when a new contract matching the selection criteria is entered as an addition into the system, the server sends the relevant Security Status message, with the UnsolicitedIndicator field = "Y" too.


Client
Server
Security Status Request ("e")
SubscriptionRequestType [263] = 1 (Snapshot + Updates)
Security Status ("f")
Unsolicited Indicator [325] = N
...
One message for every contract requested
Security Status ("f")
Unsolicited Indicator [325] = Y
...
One message for every status change
Security Status Request ("e")
SubscriptionRequestType [263] = 2 (Unsubscribe)

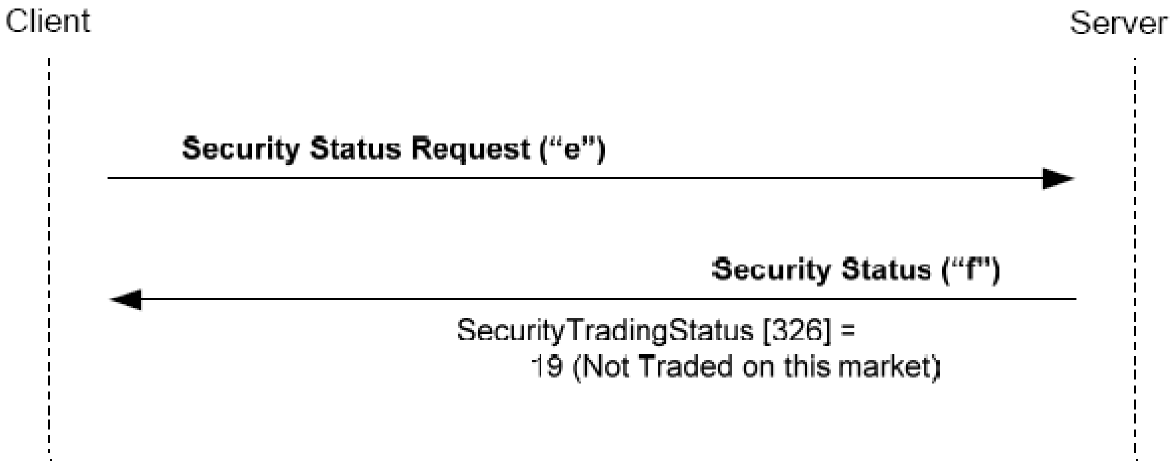
6.3.7.6 Contract definition request, without contracts that comply with the selection criteria

When there are no contracts matching the selection criteria indicated in a contract definition request, MexFix answers with a Security List message having the SecurityRequestResult field = 2. Please take into account that in this case, if the request contained a subscription, said subscription remains active and therefore, if the addition of any contract complying with the selection criteria is entered, the relevant messages will be received.


Client
Server
Security List Request ("x")
Security List ("y")
SecurityRequestResult [560] =
2 (No instruments found that match selection criteria)

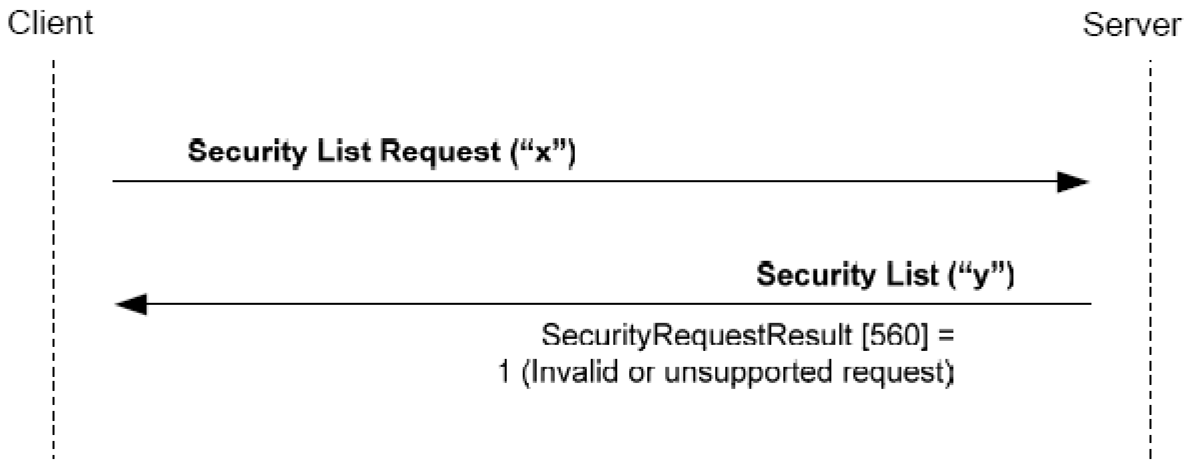
6.3.7.7 Contract status request, without contracts that comply with the selection criteria

When there are no contracts matching the selection criteria indicated in a contract status request, MexFix answers with a SecurityStatus message having the SecurityTradingStatus field = 19. Please bear in mind that in this case, if the request contained a subscription, said subscription remains active and therefore, if the addition of any contract complying with the selection criteria is entered, the relevant messages will be received.


Client
Server
Security Status Request ("e")
Security Status ("f")
SecurityTradingStatus [326] =
19 (Not Traded on this market)

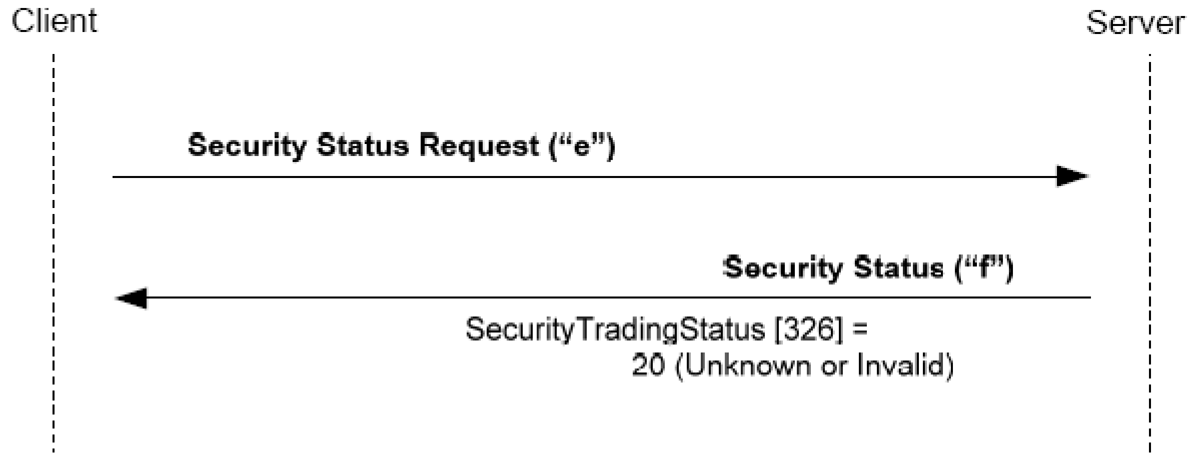
6.3.7.8 Failed contract definition request

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


Client
Server
Security List Request ("x")
Security List ("y")
SecurityRequestResult [560] =
1 (Invalid or unsupported request)

6.3.7.9 Failed contract status request

When a contract definition request is erroneous, it is answered by a Security Status message having the SecurityTradingStatus field = 20.



6.3.8 FIX 4.4 Delimitations and Adaptations

- The NewSecuritySubscription (5682) user field has been added to the Security List Request message, in order to support the functionality in charge of receiving the contract definition when a new one is entered as an addition into the system.
- The UnsolicitedIndicator field has been added to the Security List message.
- The TickSize (6138) user field has been added to the Security List message to indicate the minimum allowed quantity in the price change. This field is expressed in the same unit as the Price field.

6.4 Market Information: Prices

6.4.1 Description

This functionality allows requesting the information related to the prices of a set of contracts.

6.4.2 Information Request

The price-related information request is made by means of the Market Data Request message.

A set of contracts may be selected by combining the fields of the Instrument block as explained in section 4.1. As explained in the message detailed description several Instrument blocks may be included in order to request simultaneously more than one selection. A contract is deemed selected if it matches any one of the selection criteria.

The types of information MexFix offered are related below. A client may include a combination of these types in the same request.

- Bid
- Offer
- Price of the last trade
- Closing price
- Opening price (including auction price)
- Session high price
- Session low price
- Volume of trade
- Open position

When a request includes a Bid or Offer, depth may be specified in two ways: maximum number of price levels or only better prices.
Information reported as closing price (Settlement Price) during the session contains the precious session closing price of the contract. Once the current session closing price has been set, this value is reported. The OpenCloseSettleFlag field allows determining when the closing price has been set.

Request may include only the current situation (snapshot) or the actions involved (snapshot + update). Please take into account that it is not possible to request only updates without the snapshot. This is of the utmost importance in order to guarantee the integrity of the received information.

6.4.3 Reception of Information

MexFix returns the information requested by means of Market Data Snapshot Full Refresh messages.

According to FIX Standard, answer messages to the same request will not mix Bid and Offer information with the remaining information.

In a snapshot request, if this information is not mixed, a single message for each selected contract will be received. If the request combines Bid or Offer and other type of information, the answer will be two Market Data Snapshot Full Refresh messages.

If an information update has been requested, whenever a change occurs a new Market Data Snapshot Full Refresh message is received. This message contains both the changed information as well as the remaining fields requested when subscribing. In this case, the restriction imposed on mixing Bid or Offer information with other fields remains in force.

Please bear in mind that if there are no Bid or Offer prices for a given contract, a value zero is notified in the MDEntrySize field.

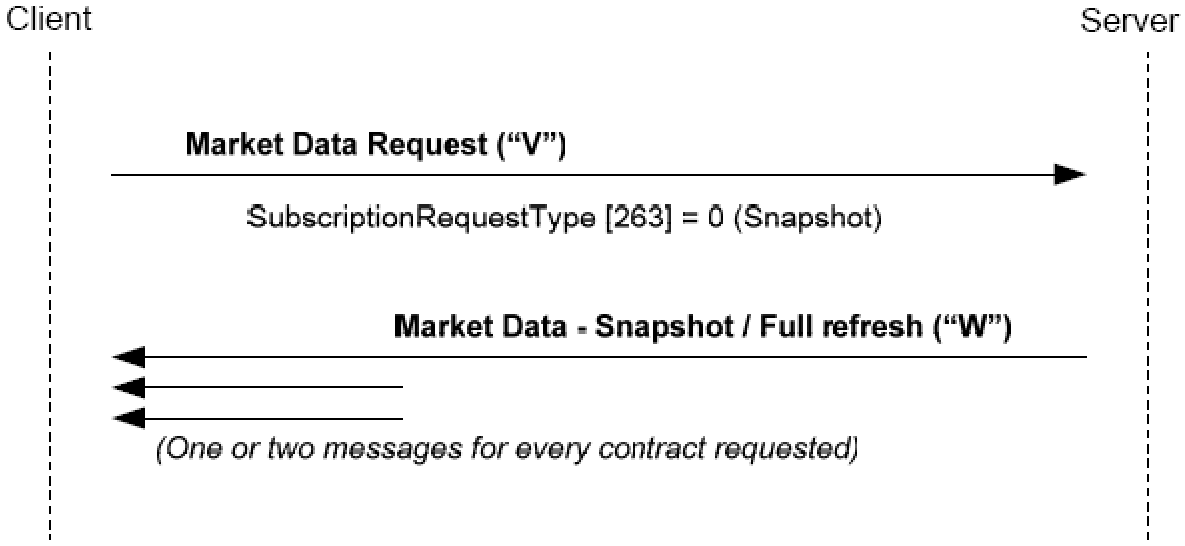
6.4.4 List of Messages

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 report that a Market Data Request has been rejected.

6.4.5 Flow of Messages

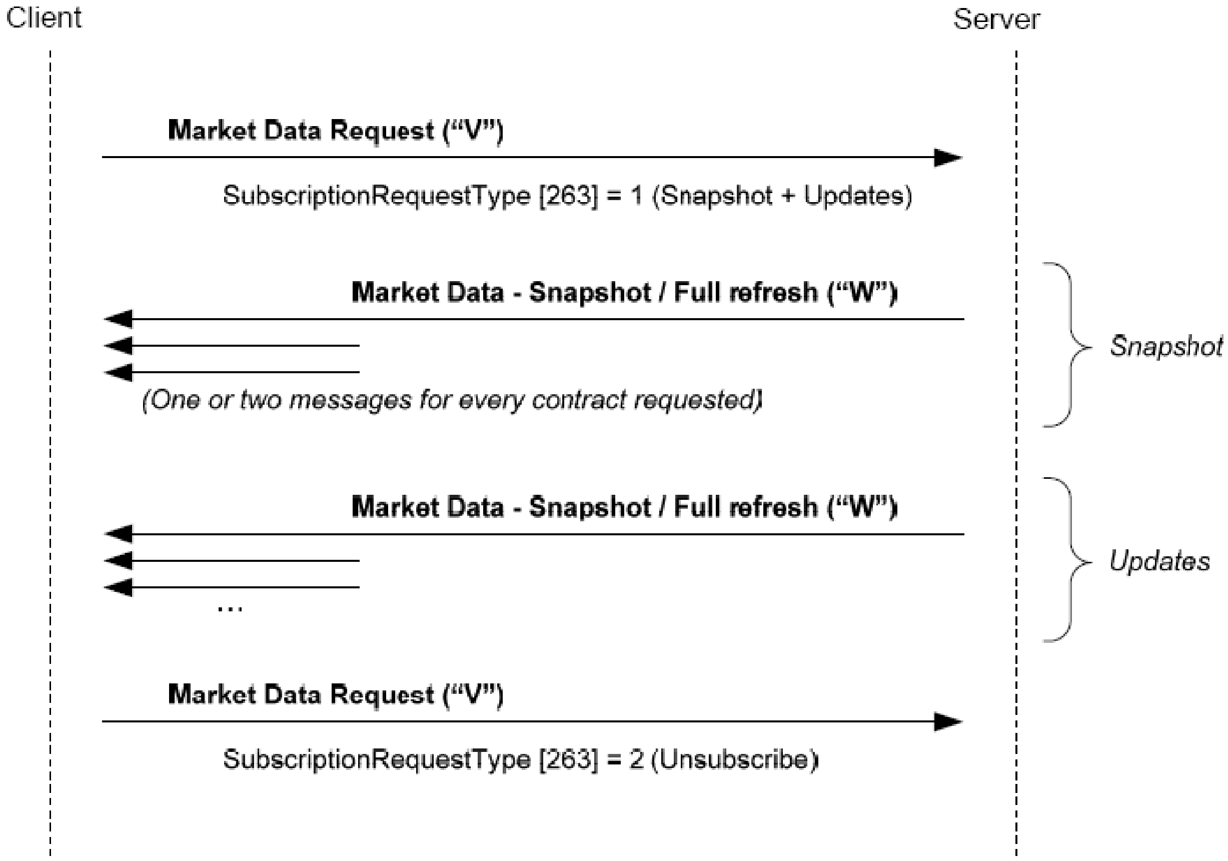
6.4.5.1 Price information request without update

A price information request, without updated, is answered by MexFix with a message for each one of the contracts. Please take into account that if Bid and Offer information combined with other information is requested, in this case MexFix returns two messages for each one of the contracts, separating these two sets of information.


Client
Server
Market Data Request ("V")
SubscriptionRequestType [263] = 0 (Snapshot)
Market Data - Snapshot / Full refresh ("W")
(One or two messages for every contract requested)

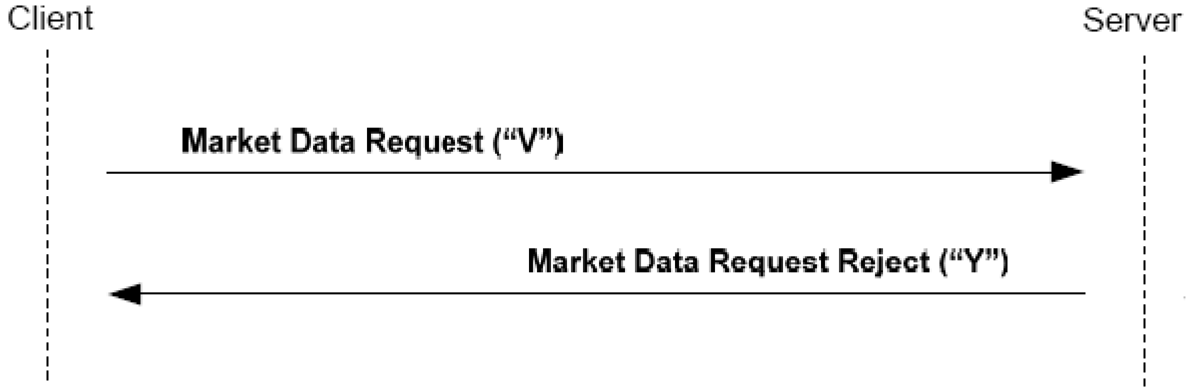
6.4.5.2 Price information request with update

A price information request with update, initially receives a set of messages for the contracts selected when making the request. From this point on, messages notifying any changes occurred in the market are received.


Client
Server
Market Data Request ("V")
SubscriptionRequestType [263] = 1 (Snapshot + Updates)
Market Data - Snapshot / Full refresh ("W")
(One or two messages for every contract requested)
Snapshot
Market Data - Snapshot / Full refresh ("W")
...
Updates
Market Data Request ("V")
SubscriptionRequestType [263] = 2 (Unsubscribe)

6.4.5.3 Erroneous price information request

When a price information request is erroneous, it is answered by a Market Data Request Reject message.



6.4.6 FIX 4.4 Delimitations and Adaptations

- Incremental refreshment is not supported
- In the Market Data Request Reject message, the meaning of value 0 (invalid symbol) of the MDReqRejReason field has been widened to indicate an invalid selection criteria.

6.5 Definition of Messages

6.5.1 Trading Session Status Request (Msg Type = g)

Used by the client to request the trading session status.

Tag	Name	Req	Valid values	Format	Description
	Standard Header0	Y	MsgType = g		
335	TradSesReqID	Y		String	Unique identifier for each Trading Session Status Request message. If SubscriptionRequestType = 2 it should contain the original request identifier
263	SubscriptionRequestType	Y	0 = Snapshot 1 = Snapshot + Updates	Char	
	Standard Trailer	Y			

6.5.2 Trading Session Status (Msg Type = h)

It is sent by the server to report the trading session status or to reject a Trading Session Status Request message.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = h		
335	TradSesReqID	N		String	Identifier of the referenced Trading Session Status Request message. This field is always present in the message.
336	TradingSessionID	Y		String	Market descriptive text
338	TradSesMethod	N	1 = Electronic	int	
325	UnsolicitedIndicator	N	N = El mensaje es parte de un snapshot Y = El mensaje es enviado como resultado de una suscripción	Boolean	It contains a "Y" when the message is sent as a result of a subscription

Tag	Name	Req	Valid values	Format	Description
340	TradSesStatus	Y	0 = Unknown 1 = Halted 2 = Open 3 = Closed 4 = Pre-Open (Not started) 6 = Request Rejected	int	Session Status It contains the value "6" when the message is used to reject a request. Value 4 (Pre-Open) indicates that the market is not open yet for trading. Should not be confused this value with the auction period, which is stated by contract in MexDer. (see field 326, SecurityTradingStatus, of the Security Status message)
58	Text	N		String	Error explanation. It is present if TradSesStatus = 6
	Standard Trailer	Y			

6.5.3 Security List Request (Msg Type = x)

Used by the client to request contract definition and specially to request their status.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = x		
320	SecurityReqID	Y		String	Unique identifier for each Security List Request message. If SubscriptionRequestType = 2 or NewSecuritySubscription = 2, it should contain the original request value.
559	SecurityListRequestType	Y	0 = Symbol 1 = CFICode	int	Selection criteria used
	Start <Instrument>				
55	Symbol	Y	[N/A] or contract code	String(16)	Contract code if SecurityListRequestType = 0, [N/A] if SecurityList-RequestType = 1
22	SecurityIDSource	N	8 = Exchange Symbol	String	It is required if the SecurityID is present. It is not allowed if SecurityListRequestType = 0

Tag	Name	Req	Valid values	Format	Description
461	CFICode	N	Exact length. Refer to 4.4.1 where a list of possible values is included.	String(6)	Contract type It is not allowed if SecurityListRequestType = 0
	End <Instrument>				
263	SubscriptionRequestType	N	0 = Snapshot (default value) 1 = Snapshot + Updates 2 = Unsubscribe	char	It indicates the type of request regarding contract status
5682 *	NewSecuritySubscription	N	0 = Snapshot (default value) 1 = Updates (Subscribe) 2 = Unsubscribe	Char	It indicates the type of request regarding contract definition
	Standard Trailer	Y			

6.5.4 Security List (Msg Type = y)

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

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = y		
320	SecurityReqID	Y		String	Identifier of the Security List Request message being answered
322	SecurityResponseID	Y		String	Unique identifier for each Security List message

Tag	Name	Req	Valid values	Format	Description
560	SecurityRequestResult	Y	0 = Valid request 1 = Invalid or unsupported request 2 = No instruments found that match selection criteria 4 = Instrument data temporarily unavailable 5 = Request was rejected because the CFICode specified is not supported	int	Result of the request identified by SecurityReqID
325*	UnsolicitedIndicator	N	N = The message is part of a snapshot Y = The message is sent as a result of a subscription (new contract)	Boolean	It contains a "Y" when the message is sent as a result of a subscription
393	TotNoRelatedSym	N		int	Total number of contracts matching the request selection criteria. The number of contracts contained by the message is indicated in the NoRelatedSym field. This field is always present when SecurityRequestResult = 0
893	LastFragment	N		Boolean	It indicates when the message is the last one in an answer sequence to a single request. This field is always present when SecurityRequestResult = 0
146	NoRelatedSym	N	1	NumInGroup	It indicates the number of contracts included in this message. It is omitted when there are no contracts meeting the selection criteria
	Start <Instrument>				
☞55	Symbol	N	[N/A] or contract code	String(16)	Contract code It is present if NoRelatedSym has been specified

Tag	Name	Req	Valid values	Format	Description
☞48	SecurityID	N	If you need more detail on market codes, refer to Table 20 in the “Coding Tables” document.	String	Contract group
☞22	SecurityIDSource	N	8 = Exchange Symbol	String	It is present if NoRelatedSym has been specified
☞454	NoSecurityAltID	N	1	NumInGro up	Number of Alternate ID codes for the instrument. Only one
☞☞455	SecurityAltID	N		String	Alternate Id Code
☞☞456	SecurityAltIDSource	N	8 = Exchange code	String	
☞461	CFICode	N	Exact length. Refer to Table 16 in the “Coding Tables” document where a list of possible values is included	String(6)	Type of contract according to standard ISO 10962
☞200	MaturityMonthYear	N	YYYYMM or YYYYMMwN	Month-Year	Contract maturity
☞541	MaturityDate	N		LocalMktD ate	Maturity date
☞225	IssueDate	N		UTCDate	Contract issuance date
☞202	StrikePrice	N		Price	Exercise price. It is present only in options
☞231	ContractMultiplier	N		Float	It indicates the ratio or multiplication factor to convert “nominal” units (for instance contracts) into total units (for instance shares).
☞107	SecurityDesc	N		String	Description of the contract group Refer to Table 20 in the “Coding Tables” document where a list of groups is included
☞6138*	TickSize	N		Price	Minimum quantity allowed in price change. This field is always present in the message.
	End <Instrument>				

Tag	Name	Req	Valid values	Format	Description
☞711	NoUnderlyings	N	1	NumInGroup	It is present if the contract has another underlying contract.
	Start <UnderlyingInstrument>				
☞☞311	UnderlyingSymbol	Y		String(16)	Symbol of the underlying contract
	End <UnderlyingInstrument>				
☞15	Currency	N		Currency	Foreign exchange code Expressed according to standard ISO 4217
☞555	NoLegs	N		NumInGroup	The number of contracts constituting the contract. It is present only in time-spread contracts
☞☞	Start <InstrumentLeg>				
☞☞600	LegSymbol	N		String(16)	Contract code. It is present if NoLegs has been specified
☞☞623	LegRatioQty	N	Always an integer	Float	Number of LegSymbol contracts included in a Symbol contract
☞☞624	LegSide	N	1 = Buy 2 = Sell	char	It indicates if the LegSymbol for the contract is to buy or sell It is present if NoLegs has been specified
	End <InstrumentLeg>				
	Standard Trailer	Y			

6.5.5 Security Status Request (MsgType = e)

Message used by the client to request the contract status.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = e		
324	SecurityStatusReqID	Y		String	Unique identifier for each Security Status Request message If SubscriptionRequestType = 2 it should contain the original request identifier
	Start <Instrument>				

55	Symbol	Y	[N/A] or contract code	String(16)	Contract code
22	SecurityIDSource	N	8 = Exchange Symbol	String	It is required if the SecurityID is present.
461	CFICode	N	Exact length. Refer to 4.4.1 where a list of possible values is included.	String(6)	Contract type
	End <Instrument>				
263	SubscriptionRequestType	Y	0 = Snapshot 1 = Snapshot + Updates	char	
	Standard Trailer	Y			

6.5.6 Security Status (MsgType = f)

Message sent by the server to inform about the status of one or more contracts.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = f		
324	SecurityStatusReqID	N		String	Identifier of the Security List Request message being answered This field is always present in the message.
	Start <Instrument>				
55	Symbol	Y	Contract code	String(16)	Contract code. The value is ignored if the SecurityTradingStatus is 19 or 20.
	End <Instrument>				
325	UnsolicitedIndicator	N	N = The message is part of a snapshot Y = The message is sent as a result of a subscription	Boolean	It contains a “Y” when the message is sent as a result of a subscription and an “N” otherwise. It contains an “N” if the SecurityTradingStatus is 19 or 20. This field is always present in the message.

Tag	Name	Req	Valid values	Format	Description
326	SecurityTradingStatus	N	2 = Trading Halt (Not available) 3 = Resume (Available) 15= New Price Indication 19 = Not Traded on this Market 20 = Unknown or Invalid 21 = Pre-Open	int	It informs about the Contract Status. The value "19" indicates there is no contract matching the selection criteria. The value "15" indicates that the contract is in "flashing" mode The value "20" indicates that the request was invalid. The value "21" indicates that the contract is in auction. This value should not be confused with the market "Pre-Open" status which indicates that no contract is negotiable. (Refer to field 340, TradSesStatus, of the message Trading Session Status)
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.
58	Text	N		String	It contains an error explanation if SecurityTradingStatus = 20. If SecurityTradingStatus = 15 then indicates the "flashing" side. 1=Buy, 2= Sell. If SecurityTradingStatus has anyother value it indicates the side that stop "flashing"(1=Buy, 2=Sell). If omitted the "flashing" status remains the same.
	Standard Trailer	Y			

6.5.7 Market Data Request (Msg Type = V)

Message used by the client to request price information.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = V		
262	MDReqID	Y		String	Unique identifier for each Market Data Request message. If SubscriptionRequestType = 2 it should contain the original request identifier
263	SubscriptionRequestType	Y	0 Snapshot 1 = Snapshot + Updates	Char	
264	MarketDepth	Y	0 = Full Book 1 = Top of Book n = exact depth (n>1)	Int	Price depth. It is ignored if none of the MDEntryType occurrences is Bid or Offer.
265	MDUpdateType	N	0 = Full refresh	Int	It is required if SubscriptionRequestType = 1
267	NoMDEntryTypes	Y		NumInGroup	Number of MDEntryType fields contained by the message
□ 269	MDEntryType	Y	0 = Bid 1 = Offer 2 = Trade 4 = Opening Price 6 = Settlement Price 7 = Trading Session High Price 8 = Trading Session Low Price B = Trade Volume (total volume for contract in session) C = Open Interest	Char	Type of market information requested
146	NoRelatedSym	Y		NumInGroup	Number of selection criteria
	Start <Instrument>				

Tag	Name	Req	Valid values	Format	Description
□ 55	Symbol	Y	[N/A] or contract code	String(16)	Contract code
□ 22	SecurityIDSource	N	8 = Exchange Symbol	String	It is required if the SecurityID has been specified
□ 461	CFICode	N	Exact length. Refer to 4.4.1 where a list of possible values is included.	String(6)	Contract type
	End <Instrument>				
	Standard Trailer	Y			

6.5.8 Market Data Request Reject (Msg Type = Y)

Message used by MexFix to reject a Market Data Request.

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

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

Message used by MexFix to communicate price information requested by means of a Market Data Request message.

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = W		
262	MDReqID	Y		String	Identifier of the Market Data Request message being answered
	Start <Instrument>				
55	Symbol	Y	Contract code	String(16)	Contract code
	End <Instrument>				
268	NoMDEntries	Y		NumInGroup	Number of the following entries
□269	MDEntryType	Y	0 = Bid 1 = Offer 2 = Trade 4 = Opening Price 6 = Settlement Price 7 = Trading Session High Price 8 = Trading Session Low Price B = Trade Volume (total volume for contract in session) C = Open Interest	char	Type of information contained by the present entry. If values 0 or 1 are present, the message contains none of the other values
□270	MDEntryPx	N		Price	Price. It is present when the MDEntryType is (0, 1, 2, 4, 6, 7, 8). Presente cuando MDEntryType está en (0, 1, 2, 4, 6, 7, 8). If it is not present when MDEntryType = 6 it should be understood that the price is 0.
□271	MDEntrySize	N		Qty	Volume. It is present when the MDEntryType is (0, 1, 2, B, C). For the value "C" it is the *Open Interest* at the beginning of the session.

Tag	Name	Req	Valid values	Format	Description
□273	MDEntryTime	N		UTCTimeOnly	Update time. It is present only if the MDEntryType is 0,1 or 2. In case of Bid (0) or Offer (1), it is only present for one of the values (MDEntryPositionNo = 1) and refers to the update of Bid and Offer in general.
□274	TickDirection	N	0 = Plus Tick 1 = Zero-Plus Tick 2 = Minus Tick 3 = Zero-Minus Tick	char	It is present when MDEntryType = 2.
□286	OpenCloseSettleFlag	N	1 = Session Open / Close / Settlement entry 4 = Entry from previous business day 5 = Theoretical Price Value	MultipleValueString	When the MDEntryType = 6, the values 1 and 4 are used to indicate if the closing price is the one of the previous session (value 4) or the final one of the current session (value 1). When the MDEntryType = 4, the values 1 and 5 are used to indicate if the closing price being reported is the session opening price (value 1) or the auction price (value 5). The field reporting auction price is sent only once per auction, when such price has been set.
□290	MDEntryPositionNo	N		int	Order number of a price within those of the same type (bid or offer). Numbered in the most to the less competitive order, beginning by 1. It is present if the MDEntryType is 0 or 1.
□811	PriceDelta	N		float	It may be present if the MDEntryType = 6
	Standard Trailer	Y			

7. Communication of Events

7.1 Introduction

In this chapter, two functionalities based in the News messages are described:

- Information given by the market supervisor to one or more clients
- Sending of messages from a client to the market supervisor

The information transmitted in both cases is a free format text.

A client program should not necessarily subscribe to receive these messages. Any client is implicitly subscribed from the beginning of the session.

There is no mechanism to verify if a message has been duly delivered to its addresses.

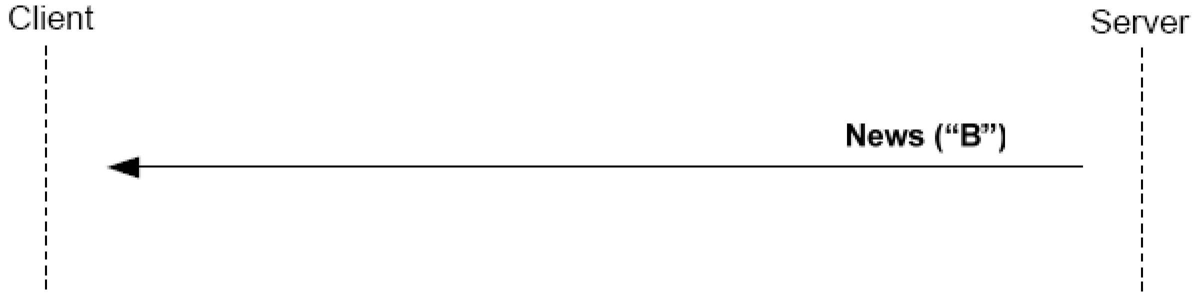
On establishing a communication connection, if the client continues the FIX session, it receives all News messages pending from the moment of disconnection. When the client chooses to begin a new FIX session, receives all News messages addressed to it, generated from the beginning of the session.

7.2 List of Messages

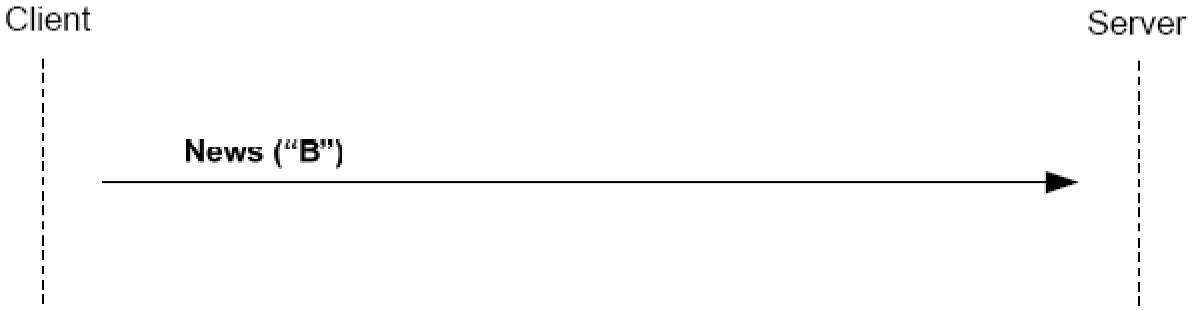
Message	Description
News (Msg Type = B)	It is used to receive text messages from the market supervisor. It is also used to send text messages to the market supervisor.

7.3 Flow of Messages

7.3.1.1 Reception of messages



7.3.1.2 Sending of messages



7.4 FIX 4.4 Delimitations and Adaptations

- It only allows a line of up to 78 characters per message

7.5 Definition of Messages

7.5.1 News (Msg Type = B)

Tag	Name	Req	Valid values	Format	Description
	Standard Header	Y	MsgType = B		
61	Urgency	N	0 = Normal 1 = Flash 2 = Background	char	The default value is 0
148	Headline	Y		String	Message heading.
33	LinesOfText	Y	1	NumInGroup	Number of text lines. Sólo se permite una línea
☞58	Text	Y		String(78)	One text line.
	Standard Trailer	Y			

8. Appendices

8.1 Appendix A- User fields

In the following table the user fields used in the messages of this manual are described:

Tag	Name	Format	Description
5682	NewSecuritySubscription	char	Field to request the subscription to the definition of new contracts
6138	TickSize	Price	Minimum difference between order prices for a contract