FIX protocol specification for Indicative Quote System of Derivatives market

version 1.0.0

Moscow 2018

Table of Contents

History of changes

Date	Version	Changes

1. Introduction

1. Document purpose

The document below describes the protocol FIX provided by the Moscow Exchange for connection to Indicative Quote System of Derivatives market. The description is based on the standard FIX protocol (Financial Information Exchange, http://www.fixtrading.org, version 4.4) specification. The specification does not contain neither technical nor administrative details on network connection or security protection methods.

2. General description

Indicative Quote System (IQS) is a new service provided by Moscow Exchange, which allows trade participants to view and take actions with indicative quotes on the Derivatives Market instruments. An Indicative quote here stands for the buy (or sell) price indicating a trade participant's intention to perform a trade, without any obligation to do that (which means that no collateral will be pledged to cover the trades). Therefore, a trade participant is able to indicate their trading interest in a wide range of instruments, without having any financial expenses.

Indicative quotes can be added on all types of the instruments traded on the Derivatives Market, in accordance with the rights granted by the trading administrator. The following instrument types are supported:

- Futures;
- Options;
- Multi-leg instruments (calendar spreads).

The protocol provided is designed based on the standard FIX 4.4 protocol. The protocol consists of transport, session and application layers.

The transport layer defines rules of: a) transferring data as set of messages, b) composing message as set of fields, c) composing fields using field ID and field value. The transport layer description is a part of FIX 4.4 specification, therefore, it is not listed in this specification.

The session layer provides identification of each participant as well as guaranteed delivery and message consistent processing, connection status control and session recovery in case of a failure. This specification contains a brief description of the session layer protocol in order to cover certain parameters needed for establishing connection to FIX Gate.

This specification is based upon application layer protocol description, which defines rules of trading process.

3. Terms and definitions

This document contains the following terms, definitions and acronyms:

Term	Definition
IQS (Indicative Quote System)	Indicative Quote System of the Derivatives Market
Order	A trading instruction added into the SPECTRA trading system
Indicative Quote (or simply quote)	A trading instruction added into the IQS without collateral verification
Indicative trade	A trade performed as a result of matching indicative quotes within IQS.
Trade	A trade performed as a result of matching orders within SPECTRA trading system.

2. Data types

Within the protocol, the following data types are used:

Type	Details
char	Single character value, can include any alphanumeric character or punctuation except the delimiter. All char fields are case sensitive (i.e. m != M).
String	Alpha-numeric free format strings, can include any character or punctuation except the delimiter. All char fields are case sensitive (i.e. morstatt != Morstatt).
float	Sequence of digits with optional decimal point and sign character (ASCII characters "-", "0" - "9" and "."); the absence of the decimal point within the string will be interpreted as the float representation of an integer value. The number of decimal places used should be a factor of business/market needs and mutual agreement between counterparties. Note that float values may contain leading zeros (e.g. "00023.23" = "23.23") and may contain or omit trailing zeros after the decimal point (e.g. "23.0" = "23.0000" = "23" = "23.").
Price16.5	Float field representing a price. The number of significant digits is sixteen. The number of decimal places is five.
Int	Sequence of digits without commas or decimals and optional sign character (ASCII characters "-" and "0" - "9"). The sign character utilizes one byte (i.e. positive int is "99999" while negative int is "-99999"). Note that int values may contain leading zeros (e.g. "00023" = "23"). Examples: 723 in field 21 would be mapped int as |21=723|, -723 in field 12 would be mapped int as |12=-723|.
NumInGroup	Int field representing the number of entries in a repeating group. Value must be positive.
Int32	Integer signed, 4 bytes.
Int64	Integer signed, 8 bytes.
UInt32	Integer unsigned, 4 bytes
UInt64	Integer unsigned, 8 bytes
StringN	String of symbols with fixed length.
UTCTimestamp	Time/date combination represented in UTC (Universal Time Coordinated) in YYYYMMDD-HH:MM:SS.sss (milliseconds) or YYYYMMDD-HH:MM:SS.sssssssss (nanoseconds) format.
SeqNum	Int32 field representing a message sequence number. Value must be positive.
Boolean	Char field containing one of two values: 'Y' = True/Yes, 'N' = False/No.
LocalMktDate	Date of Local Market (vs. UTC) in YYYYMMDD format. Valid values: YYYY = 0000-9999, MM = 01-12, DD = 01-31.

3. Shared fields

There are some messages containing the same field sets, for example, the 'Standard Header' and 'Standard Trailer' fields which contain some service information. Some of such field sets are described below:

- **Tag** – the unique field ID, used for generating a FIX message.
- **Field** – the field name, not used for generating FIX messages and described for your reference only.
- **Mandatory** – a field attribute: specifies whether the field in a message is mandatory or optional.
  - Y - mandatory field;
  - N - optional field;
  - C - mandatory, if meets the condition (see 'Details').
- **Type** - field type.
- **Details** – detailed description of the field.
- **Allowable values** - additional limitations.

The "*****" symbol - flag of difference from the standard FIX protocol.

3.1. Standard Header group

The standard header contained in every message contains.

Tag	Field name	Mandatory	Type	Details	Allowable values
8	BeginString	Y	String7	Specifies message start and protocol version.	="FIX.4.4"
9	BodyLength	Y	Length	Message body length. Calculated in accordance with the standards.	
35	MsgType	Y	String10	The MsgType ID which is unique for every message.	
49	SenderCompID	Y	String64	Sender ID. The allowable values are specified by the exchange individually for every trading firm (broker firm).	
56	TargetCompID	Y	String	Recipient ID. (FIX-gate ID).	
34	MsgSeqNum	Y	SeqNum	Message sequential number.	
52	SendingTime	Y	UTCTimestamp	Message sending time.	
122	OrigSendingTime	N*	UTCTimestamp	Original message transmission time when resending messages in reply to resend request (message Resend Request (2)), in UTC. Mandatory if a message is sent in reply to resend request (message Resend Request (2)).	
97	PossResend	N	Boolean	Indicates the message containing some data which had been already sent with another sequential number.	
43	PossDupFlag	N	Boolean	Indicates the allowance for resending message using the same sequential number.	

3.2. Standard Trailer group

The standard trailer (end) which every message contains.

Tag	Field name	Mandatory	Type	Details
10	CheckSum	Y	String3	Message checksum. For calculation method description see FIX, Volume 2: 'Checksum Calculation'.

3.3. Parties group

Group 'Parties' is a mandatory group which contains the unsolicited message sender login details.

The fields order is fixed.

Tag	Field name	Mandatory	Type	Details	Allowable values
453	NoPartyIDs	N	NumInGroup	Elements quantity in block.	Must be >= 1.
=> 448	PartyID	Y*	String64	Counterparty ID	Client login
=>447	PartyIDSource	Y*	char	PartyID source type	"C", Generally accepted market participant identifier
=>452	PartyRole	Y*	Int32	ID type.	"3" (Client ID) - client login

4. Session layer protocol

Session layer protocol which provides parties authentication, guaranteed messages delivery and sequential message processing, connection status and session recovery in case of any failure.

4.1. Supported messages

- **Logon** - Initiates session.
- **Logout** - Initiates or confirms session termination.
- **Heartbeat** - Ensures that session is up and running.
- **Test Request** - Used as part of session establishment procedure, must be replied with specific Heartbeat message.
- **Reject** – Informs party about incorrect or unknown message.
- **Resend Request** - Informs party that messages in particular range must be resent.
- **Sequence Reset** - Used to skip administrative messages on resend – 'Gap Fill mode'. Also used to reset messages sequence – 'Reset mode'.

All the messages can be sent in both directions.

4.1.1. Logon

Initiates or confirms session start. This message must be the first in every session.

Tag	Field name	Mandatory	Type	Details
<'Header' group>		Y		Message type 'A'.
98	EncryptMethod	Y	Int	Encryption method. Must be set to '0' – NONE_OTHER – no message encryption..
108	HeartBtInt	Y	Int	Heartbeat messages sending interval.
141	ResetSeqNumFlag	N	Boolean	Reset messages sequence for both parties.
<'Trailer' group>		Y		

4.1.2. Logout

Initiates or confirms session termination.

Tag	Field name	Mandatory	Type	Details
<'Header' group>		Y		Message type '5'.
58	Text	N	String	Reason for session termination
<'Trailer' group>		Y		

4.1.3. Heartbeat

Ensures that session is up and running. If the 'Heartbeat' message is sent in response to the 'Test Request' message, the 'TestReqID' field must contain the 'Test Request' message ID.

Tag	Field name	Mandatory	Type	Details
<'Header' group>		Y		Message type '0'.
112	TestReqID	N	String	Mandatory if sent in response to the 'Test Request' message.
<'Trailer' group>		Y		

4.1.4. Test Request

The message calls/initiates/requests the 'Heartbeat' message from the opposite party..

Tag	Field name	Mandatory	Type	Details
<'Header' group>		Y		Message type '1'.
112	TestReqID	Y	String	Request message ID, returned in the 'Heartbeat' message.

Tag	Field name	Mandatory	Type	Details
<'Trailer' group>		Y		

4.1.5. Reject

The reject message should be issued when a message is received but cannot be properly processed due to a session-level rule violation. An example of when a reject may be appropriate would be the receipt of a message with invalid basic data which successfully passes CheckSum and BodyLength checks.

Tag	Field name	Mandatory	Type	Details
<'Header' group>		Y		Message type '3'.
45	RefSeqNum	Y	SeqNum	Rejected message number.
371	RefTagID	N	Int	Invalid field number.
372	RefMsgType	N	String	Rejected message type.
373	SessionRejectReason	N	Int	Rejection reason ID. • '0' – Incorrect tag. • '1' – Mandatory field missing. • '2' – Undefined tag for this message type. • ''3' – Undefined tag. • '4' – Value missing for this tag. • '5' – Incorrect value for this tag (vale limits exceeded). • '6' – Incorrect data format for this value. • '7' – Decoding error. • '8' – Signature error. • '9' – 'CompID' error. • '10' – 'SendingTime' accuracy error. • '11' – Incorrect message type. • '12' – XML validation error. • '13' – Tag already exists. • '14' – Tags definition order error. • '15' – Group fields definition order error. • '16' – Group elements number calculation error ('NumInGroup'). • '17' – Non-data field contains separator. • '99' - Other. • '7100' – Messages limit exceeded. • '7101' – System error.
58	Text	N	String	Rejection reason details.
<'Trailer' group>		Y		

4.1.6. Resend Request

The message initiates resending of a particular message range. Use 'BeginSeqNo=EndSeqNo' for a single message resending and 'EndSeqNo=0' for a range of messages starting from the particular one (where '0' indicates infinity).

Tag	Field name	Mandatory	Type	Details
<'Header' group>		Y		Message type '2'.

Tag	Field name	Mandatory	Type	Details
7	BeginSeqNo	Y	SeqNum	Number of the first message to resend.
16	EndSeqNo	Y	SeqNum	Number of the last message to resend.
<'Trailer' group>		Y		

4.1.7. Sequence Reset

Used to skip administrative messages on resend – 'Gap Fill mode'. Also used to reset messages sequence – 'Reset mode'.

Tag	Field name	Mandatory	Type	Details
<'Header' group>		Y		Message type '4'.
123	GapFillFlag	N	Boolean	Mode: • 'Y' - the 'Gap Fill' mode – the 'MsgSeqNum' field is used. If there are some administrative messages to be skipped, then the 'Sequence Reset' message is used for responding to the 'Resend Request' message. • 'N' - the 'Reset' mode - Messages sequence reset mode.
36	NewSeqNo	Y	SeqNum	New sequence number.
<'Trailer' group>		Y		

4.2. Session establishing and termination scenarios

4.2.1. Session establishing and termination

For establishing connection to FixGate a client must send the 'Logon' message including its 'SenderCompID'. If the 'Logon' message is valid and the sender was successfully authorized then FixGate sends the 'Logon' message in return, confirming that the connection has been successfully established.

For correct session termination, client must send the 'Logout' message to FixGate and receive one in return. Any other ways of session closing/termination are incorrect and may lead to an error.

Also, before sending the 'Logout' message it is recommended to send the 'Test Request' message to FixGate and receive the 'Heartbeat' message in return. This may help to avoid missing and/or lost messages.

When a connection has been established via FixGate, it is recommended to wait 30 seconds after closing the previous session before sending a new Logon message. Otherwise, the connection will be terminated by FixGate without any additional notifications.

4.2.2. Message resending request

During the initialization process or due to unexpected connection break there may be numeration error when the incoming message sequence number is greater than expected (while the common message number is always greater by 1 than that of the last message in log). In this case, a client must request the retransmission via sending the 'Resend Request' message including sequence number range for the missing messages (the 'BeginSeqNo', 'EndSeqNo' fields values).

4.2.3. Session status monitoring

The 'Heartbeat' message is used to monitor the FIX session status as well as gaps in messages sequence numbers in case of missing some incoming messages. In order to do this, the client application generates the 'Heartbeat' messages and sends it to FixGate in accordance with time interval specified by the 'HeartBtInt' field value in the 'Logon' Message.

If there is no reply from FixGate within the specified time interval (the 'HeartBtInt ' field value + transmission time), the client should generate and send the 'Test Request' message to FixGate. In case of no reply within the specified time interval the client should reestablish connection to the FixGate.

4.2.4. Resetting message sequence

The following methods are used to reset message sequence:

- Sending the 'Logon' message with the 'ResetSeqNumFlag' flag.
- Sending the 'Sequence Reset' in the 'Reset mode' mode.
- By schedule. For example, message sequence can be automatically reset by the Exchange before starting a trading session.

After message sequence was reset, there is no more option to resend any message via the 'Resend Request' procedure.

4.2.5. Session recovery after failure

In order to recover session after failure, the client should send the 'Logon' message which includes the sequence number 1 more than that of the last message in log (the 'MsgSeqNum' field). If the incoming 'Logon' message sequence number is greater than expected, then the client must request the retransmission via sending the 'Resend Request' message including sequence number range for the missing messages.

If the primary FixGate server is unreachable, the client is recommended to establish connection to the secondary server to continue working according to the rules stated above.

The primary and secondary servers do not synchronize message sequence numbers, so that a client will not be able to receive messages starting from the last received one once they have switched from one server to another. When trying to connect to another server, the client will receive a message with its sequence number less than expected. In this case, it is recommended to reset the message sequence number counter.

5. Application layer

5.1. List of supported messages

Messages sent from client into FixGate:

- **Quote** – Add/change a quote
- **QuoteCancel** – Cancel a quote
- **ExecutionReport** – Confirm an indicative trade.

Messages sent from FixGate to client:

- **QuoteResponse** – Quote Add/change success.
- **ExecutionAck** – Indicative trade confirmation success/reject
- **QuoteAck** – Quote cancellation success/reject
- **ExecutionReport** – Indicative trade execution report

5.1.1. Quote

Request to add/change an indicative quote for any available instrument. A value '1' in field 'QuoteModelType' (Tag=2403) stands for adding quote, while '2' allows to change quote.

Each adding transaction must consists of 1 or 2 quotes. An adding transaction of 2 quotes must contain one buy and one sell quote on the same instrument, where buy price must exceed the sell one. Also, the quotes may be of different volume.

A trading participant is able to change an already added indicative quote by changing one (or more) parameters, i.e. price, volume, external ID. To change a quote, a trading participant must specify its ID assigned by the Exchange. Each changing transaction must consists of 1 or 2 quotes. An changing transaction of 2 quotes must contain one buy and one sell quote on the same instrument, where buy price must exceed the sell one. Also, the quotes may be of different volume.

Tag	Field	Manda-tory	Type	Details
<Group Header>		Y		Message type 'S'.
2403	QuoteModelType	Y	Int8	Usage: • '1' (Quote entry) - add a quote. • '2' (Quote modification) - change a quote.
1	Account	Y	String7	Client 7-digit code
48	SecurityID	C	Int32	Instrument ID. Mandatory for messages with QuoteModelType=1.
59	TimeInForce	C	char	Quote type: • '0' Day – (remains in order-book after being partially filled). • '3' IOC – (cancels at auction end). Mandatory for messages with QuoteModelType=1.
574	MatchType	C	String2	Confirmation type for indicative trade: • '4' – confirm automatically. • '10' – confirm with sending a message. Mandatory for messages with QuoteModelType=1.
20023	DontCheckMoney	C	Boolean	Risk calculation for the given quote for client section: • 'Y' – on • 'N' – off Mandatory for messages with QuoteModelType=1.
1867	OfferID	C	UInt64	Unique ID of the sell quote to change, assigned by Exchange. Mandatory for messages with QuoteModelType=2.
135	OfferSize	C	UInt64	Sell quote volume. Mandatory for messages with field 'OfferPx'.
133	OfferPx	C	Price16.5	Sell volume price. Mandatory for messages with field 'OfferSize'.
20025	OfferExternalID	N	UInt64	Sell quote external ID.

Tag	Field	Mandatory	Type	Details
20032	OfferText	N	String20	Sell quote comment.
390	BidID	C	UInt64	Unique ID of the buy quote to change, assigned by Exchange. Mandatory for messages with QuoteModelType=2.
134	BidSize	C	UInt64	Buy quote volume. Mandatory for messages with field 'BidPx'.
132	BidPx	C	Price16.5	Buy volume price. Mandatory for messages with field 'BidSize'.
20024	BidExternalID	N	UInt64	Buy quote external ID.
20031	BidText	N	String20	Buy quote comment.
1166	QuoteMsgID	Y	String20	Client ID for add/change quote request.
20019	Mode	C	Int8	Change quote command work mode: • '0' – Do not change the initial quote volume. All new amounts will be ignored. • '1' – Change quote volume. The quote volume will be replaced with the new one. • '2' – Delete previously added quotes. If there is at least one quote volume that does not match the new one, both quotes will be deleted. Otherwise, the previously added quotes will be replaced with the new ones. • '3' – Replace quote volumes with the new ones, excluding the part (not less than 0) that has been already matched. If a new volume is less than the part matched, both quotes will be deleted. Mandatory for messages with QuoteModelType=2.
<Group Trailer>		Y		

5.1.2. QuoteCancel

Quote cancel request. One is able to perform either single cancellation or mass cancellation, depending on value in field QuoteCancelType (Tag=298): "5" (single quote cancel), "1" (mass quote cancel).

On single quote cancel, the quote to cancel is specified with the ID assigned by the Exchange.

Upon mass quote cancel, you can select quotes by specifying any set of settings (including empty subset), i.e.:

- Client code;
- Underlying asset code;
- Instrument code;
- Direction (Buy/Sell);
- Quote external ID.

Tag	Field	Mandatory	Type	Details
<Group Header>		Y		Message type 'Z'.
298	QuoteCancelType	Y	Int8	Usage: • '5' - single quote cancel. • '1' - mass quote cancel
1	Account	N	String7	Client 7-symbol ID. If the last three symbols in the field ‘Account’ are equal to the string ‘%%%’, then all quotes for all client accounts will be cancelled.
117	QuoteID	C	UInt64	Quote ID assigned by the Exchange. Mandatory for messages with QuoteCancelType=5.
1166	QuoteMsgID	Y	String20	Client ID for request to cancel quotes
=> 295	NoQuoteEntries	N	NumInGroup	Possible values: '0', '1'.
=> 54	Side	C	char	Quote direction: • "1" – Buy.

Tag	Field	Mandatory	Type	Details
				• "2" – Sell. • "Y" – All. Mandatory for messages with NoQuoteEntries=1.
=> 1151	SecurityGroup	N	String25	Contract code. If not specified, or contains '%', then all quotes for all contracts will be cancelled.
=> 20027	ExternalID	N	UInt64	Quote external ID. If not 0, then all quotes with the appropriate ExternalID value will be cancelled. All values of other parameters will be ignored.
=> 48	SecurityID	N	Int32	Instrument ID
<Group Trailer>		Y		

5.1.3. ExecutionReport (output)

Confirmation of an indicative trade.

Before performing an indicative trade, the Initiator side should confirm this indicative trade with one of two methods, as follow:

- Automatically, by adding an indicative quote;
- Manually, by sending a confirmation message.

To confirm indicative trades automatically, one should specify value 'MatchType=4' in field 'MatchType (Tag=574)' of message **Quote**.

To confirm indicative trades manually, one should send a message **ExecutionReport** containing the indicative trade ID, in reply to received message **ExecutionReport** with the indicative trade status '1' (wait for confirmation). Note that the time for sending the reply message is strictly limited. Once the message could not be sent in the time frame specified, the indicative trade will be considered as unconfirmed.

Tag	Field	Mandatory	Type	Details
<Group Header>		Y		Message type '8'.
527	SecondaryExecID	Y	UInt64	Indicative trade ID
1166	QuoteMsgID	Y	String20	Client ID of the request to confirm the indicative trade.
<Group Trailer>		Y		

5.1.4. QuoteResponse

A message sent in reply to successful adding/changing a quote. The field set in the message may differ depending on the quote direction and/or command type (add/change).

Tag	Field	Mandatory	Type	Details
<Group Header>		Y		Message type 'AJ'.
297	QuoteStatus	Y	Int8	0 (Accepted)
1	Account	Y	String7	Client 7-digit code
60	TransactTime	Y	UTCTimestamp	Date and time of transaction. Format: YYYYMMDD-HH:MM:SS.sssssssss
48	SecurityID	Y	Int32	Instrument ID.
336	TradingSessionID	Y	Int32	Trading session ID.
1166	QuoteMsgID	Y	String20	Client ID of the request to add/change quotes
20023	DontCheckMoney	N	char	Risk calculation for client account for the given quote: • 'Y' – on. • 'N' – off.
20029	PrevOfferID	N	UInt64	Unique ID of the deleted sell quote. Mandatory only on quote change.
1867	OfferID	N	UInt64	Unique ID of the sell quote, assigned by the Exchange
135	OfferSize	N	UInt64	Sell quote volume
133	OfferPx	N	Price16.5	Sell quote price
20025	OfferExternalID	N	UInt64	External ID of the sell quote
20032	OfferText	C	String20	Client comment for the sell quote. Mandatory only when there is a client comment added into the quote add/change request.

Tag	Field	Mandatory	Type	Details
20028	PrevBidID	N	UInt64	Unique ID of the deleted buy quote. Mandatory only on quote change.
390	BidID	N	UInt64	Unique ID of the buy quote, assigned by the Exchange
134	BidSize	N	UInt64	Buy quote volume
132	BidPx	N	Price16.5	Buy quote price
20024	BidExternalID	N	UInt64	External ID of the buy quote
20031	BidText	C	String20	Client comment for the buy quote. Mandatory only when there is a client comment added into the quote add/change request.
<Group Trailer>		Y		

5.1.5. ExecutionAck

Message on successful/unseccesfull confirmation of an indicative trade for the given indicative quote.

Tag	Field	Mandatory	Type	Details
<Group Header>		Y		Message type 'BN'.
1036	ExecAckStatus	Y	char	Confirmation transaction status: • '1' (Accepted) - Success. • '2' (Don't know / Rejected) - Failure
60	TransactTime	Y	UTCTimestamp	Date and time of operation. Format: YYYYMMDD-HH:MM:SS.sssssssss
527	SecondaryExecID	Y	UInt64	Indicative trade ID.
1166	QuoteMsgID	Y	String20	Client ID of the request confirmation of an indicative trade for the given indicative quote.
300	QuoteRejectReason	C	Int32	Confirmation request rejection code. Mandatory only for messages with ExecAckStatus=2.
58	Text	C	String255	Confirmation request rejection details. Mandatory only for messages with ExecAckStatus=2.
<Group Trailer>		Y		

5.1.6. QuoteAck

Quote delete report/Reject in deletion. This message will be sent in reply at:

- Successful single/mass quote cancel
- Reject in add/change/delete quote

QuoteAck - Quote deletion report

Tag	Field	Mandatory	Type	Details
<Group Header>		Y		Message type 'CW'.
1865	QuoteAckStatus	Y	Int8	'1' (Accepted)
298	QuoteCancelType	Y	Int8	Type of delete transaction: • '5' - cancel single quote by its ID • '1' - mass quote cancel
1	Account	C	String7	Client 7-digit code. Mandatory only for message with QuoteCancelType=5.
117	QuoteID	C	UInt64	Quote ID assigned by Exchange. Mandatory only for message with QuoteCancelType=5.
20026	QuoteSize	C	UInt64	Volume of the quote to delete. Mandatory only for message with QuoteCancelType=5.
60	TransactTime	Y	UTCTimestamp	Date and time of transaction. Format: YYYYMMDD-HH:MM:SS.sssssssss
20027	ExternalID	C	UInt64	Quote external ID. Mandatory only for message with QuoteCancelType=5.

Tag	Field	Mandatory	Type	Details
1166	QuoteMsgID	C	String20	Client ID of the request to cancel/mass cancel quotes. Missed for messages with QuoteCancelType=5 when a quote cancelled due to mass cancel transaction.
1168	TotNoCxldQuotes	C	NumInGroup	Number of cancelled quotes. Mandatory only for message with QuoteCancelType=5.
<Group Trailer>		Y		

QuoteAck - Rejection of adding/changing/canceling quotes

Tag	Field	Mandatory	Type	Details
<Group Header>		Y		Message type 'CW'.
1865	QuoteAckStatus	Y	Int8	'2' (Rejected)
298	QuoteCancelType	C	Int8	Rejection transaction type: • '5' - cancel single quote. • '1' - mass quote cancel. Mandatory only for messages with quote cancel request
300	QuoteRejectReason	Y	Int32	Rejection reason code
1328	RejectText	Y	String255	Rejection reason details
60	TransactTime	Y	UTCTimestamp	Date and time of transaction. Format: YYYYMMDD-HH:MM:SS.sssssssss
1166	QuoteMsgID	Y	String20	Client ID of the request.
54	Side	C	char	Erroneous quote: • '1' – Sell quote. • '2' – Buy quote. • 'Y' – Both quotes. Mandatory only for messages with quote add or change requests.
<Group Trailer>		Y		

5.1.7. ExecutionReport (incoming)

Indicative trade performance report message. The message is sent on:

- Indicative trade performed within IQS.
- Indicative trade status change within IQS.

Tag	Field	Mandatory	Type	Details
<Group Header>		Y		Message type '8'.
150	ExecType	Y	char	'F' (Trade)
39	OrdStatus	Y	char	Quote status: • '1' - Partially filled. • '2' - Filled.
1	Account	Y	String7	Client 7-digit code.
60	TransactTime	Y	UTCTimestamp	Date and time of transaction. Format: YYYYMMDD-HH:MM:SS.sssssssss
48	SecurityID	Y	Int32	Instrument ID.
336	TradingSessionID	Y	Int32	Trading session ID.
1166	QuoteMsgID	Y	String20	Client ID for add/change quote request.
117	QuoteID	Y	UInt64	Quote ID, assigned by Exchange.
32	LastQty	Y	UInt64	Filled part volume.

Tag	Field	Mandatory	Type	Details
31	LastPx	Y	Price16.5	Filled part price.
527	SecondaryExecID	Y	UInt64	Indicative trade ID.
151	LeavesQty	Y	UInt64	Remaining volume.
20027	ExternalID	Y	UInt64	Quote external ID.
1057	AggressorIndicator	Y	Boolean	Participant role in indicative trade, i.e. Initiator (initial quote) or Contractor (matching quote). Possible values: • 'Y' – Contractor. • 'N' – Initiator.
574	MatchType	Y	String2	Indicative trade confirmation method: • '4' – automatic confirmation. • '10' – manual confirmation (by sending message).
20030	Status	Y	Int32	Indicative trade status: • '0' – Indicative trade is being processed. • '1' – Wait for confirmation from Initiator side. • '2' – Confirmation from Initiator side received. • '3' – Indicative trade has not performed due to an error. • '4' – Indicative trade performed.
20033	RejectReason	C	Int32	Error code from IQS. Mandatory only for messages with Status=3. Possible values: • '1' – Indicative trade not confirmed. • '2' – Error adding order for Contractor side. • '3' – Error adding order for Initiator side. • '4' – Order not found on SPECTRA. • "5" – Trade time-out on SPECTRA. • "6" – SPECTRA is unavailable.
54	Side	Y	char	Quote direction: • '1' – Buy quote. • '2' – Sell quote.
20023	DontCheckMoney	Y	char	Risk calculation for client account for the given quote: • 'Y' – on. • 'N' – off.
37	OrderID	Y	UInt64	Negotiated order ID.
880	TrdMatchID	Y	UInt64	Trade ID.
103	OrdRejReason	N	Int32	Negotiated order adding request rejection reason code. For the error code available in field OrdRejReason, see **p2gate_en.pdf** (ftp://ftp.moex.com/pub/ClientsAPI/Spectra/CGate/docs/). Please note that to find a proper code value in the list one should subtract 100 the value in field OrdRejReason .
<Group Trailer>		Y		

5.1.8. Unsolicited messages

Unsolicited message is a message not associated with any input message. They can be resulted from a third-party transaction (change/cancel a quote added by the FIX login) via FixGate, P2gate, or a trading terminal. Therefore, an unsolicited message is a standard message to cancel (**QuoteAck**), or change (**QuoteResponse**) a quote, where field QuoteMsgID is missing, and an optional group **Parties** containing the transaction initiator login is added.

Tag	Field	Manda-tory	Type	Details
453	NoPartyIDs	Y	NumInGroup	'1'
=> 448	PartyID	Y	String20	Transaction initiator login
=> 447	PartyIDSource	Y	char	'C' - Generally accepted market participant identifier.
=> 452	PartyRole	Y	Int32	'3' - Client ID.

5.2. Trading interaction scenarios

5.2.1. Adding quotes

One can add one (1) or two (2) quotes of different directions via a single message.

5.2.1.1. Adding one (1) quote

The system waits for quote adding confirmation. Also, the quote can be rejected by the system (**QuoteAck**).

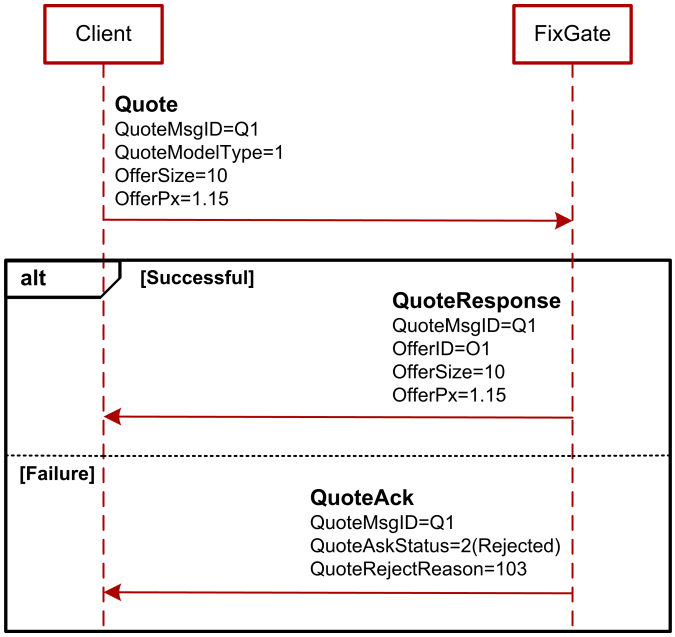

Figure 1. Diagram. Adding one (1) quote

5.2.1.2. Adding two quotes via a single message

The client side adds new quotes into the system (message **Quote**). If both quotes have been added successfully, then the system issues two messages **QuoteResponse** to the client side. If one quote of the pair is rejected, then the other one will be rejected too, which will result in single message **QuoteAck** sent by FixGate to the client side. Field 'Side' in the message indicates the erroneous quote (1 - Buy, 2 - Sell). If both quotes are rejected due to the same reason, FixGate will send a single message **QuoteAck** with Side=3 (All quotes) to the client side.

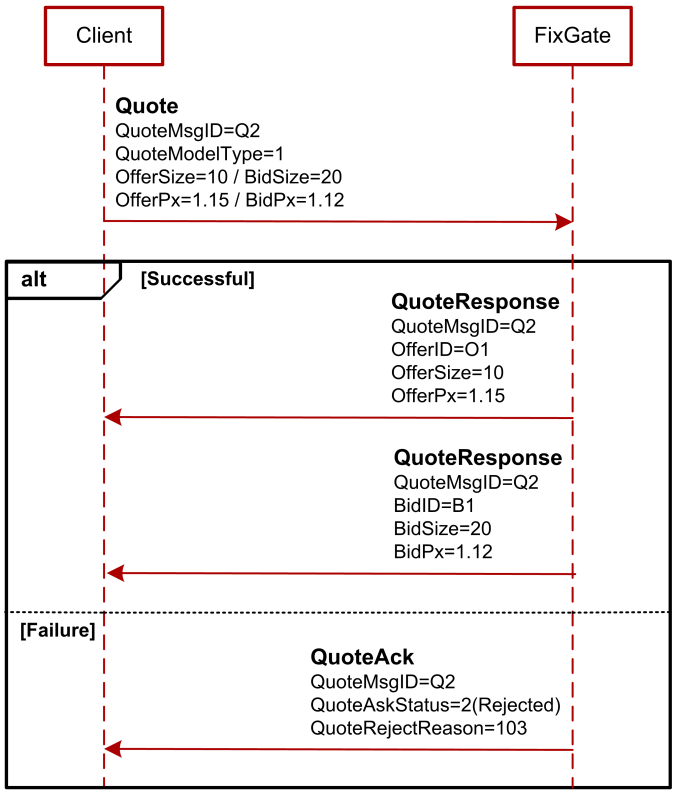

Figure 2. Diagram. Adding two quotes via single message

5.2.2. Cancelling quotes

A successfully added quote can be deleted (cancelled) by the Client side. One can cancel either a single quote, or perform mass quote cancel.

5.2.2.1. Single quote cancel

The client side sends message **QuoteCancel**, containing an Exchange-assigned quote ID in field 'QuoteID', into FixGate. On successful quote cancel, the system replies with message **QuoteAck** with QuoteAckStatus=1(Accepted). On unsuccessful cancel, the system replies with message **QuoteAck** with QuoteAckStatus=2(Rejected).

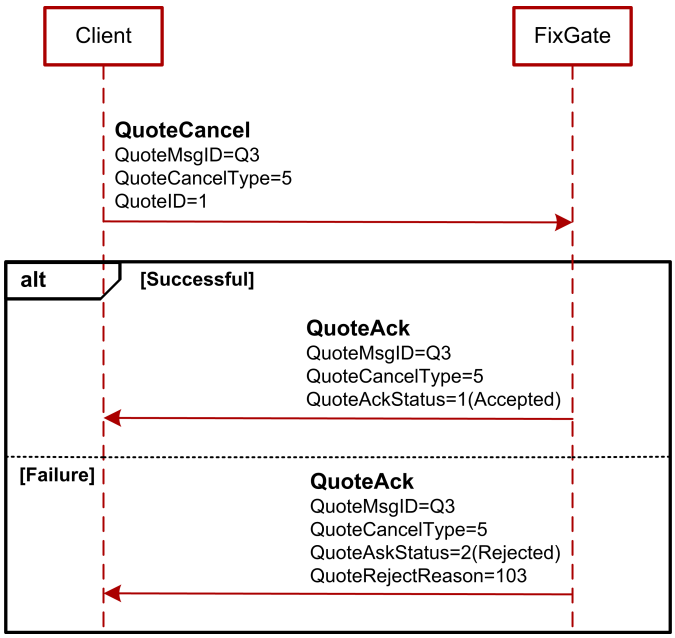

Figure 3. Diagram. SIngle quote cancel

5.2.2.2. Mass quote cancel

The client side sends message **QuoteCancel**, with mass cancel value specified in field QuoteCancelType. On successful cancel, the system replies with 5 messages **QuoteAck**, with QuoteCancelType=5 for each cancelled quote, along with cancellation report message **QuoteAck** with QuoteCancelType=1 (transaction completed). On unsuccessful cancel, the system replies with message **QuoteAck** with QuoteAckStatus=2(Rejected).

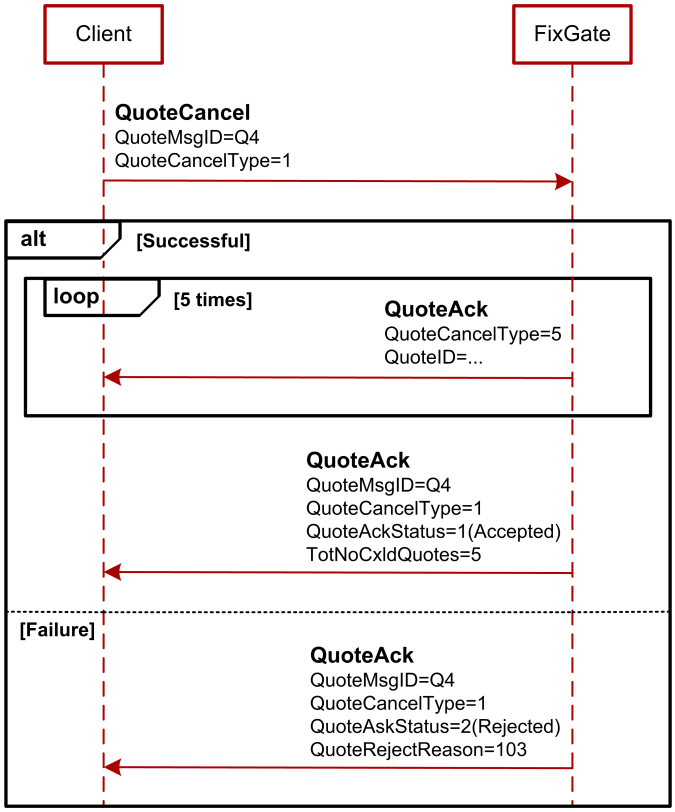

Figure 4. DIagram.Mass quote cancel

5.2.3. Changing quotes

You can change either one or two quotes via single message. For both transactions, the quotes are specified with the Exchange-assigned ID.

5.2.3.1. Changing one quote

The client side sends message **Quote** with QuoteModelType=2 (change quote) into the system. On successful change, the system replies either with message QuoteResponse (new quote added) or with message **QuoteAck** (quote deleted from order-book). On unsuccessful change, the system replies with message **QuoteAck** with QuoteAckStatus=2(Rejected).

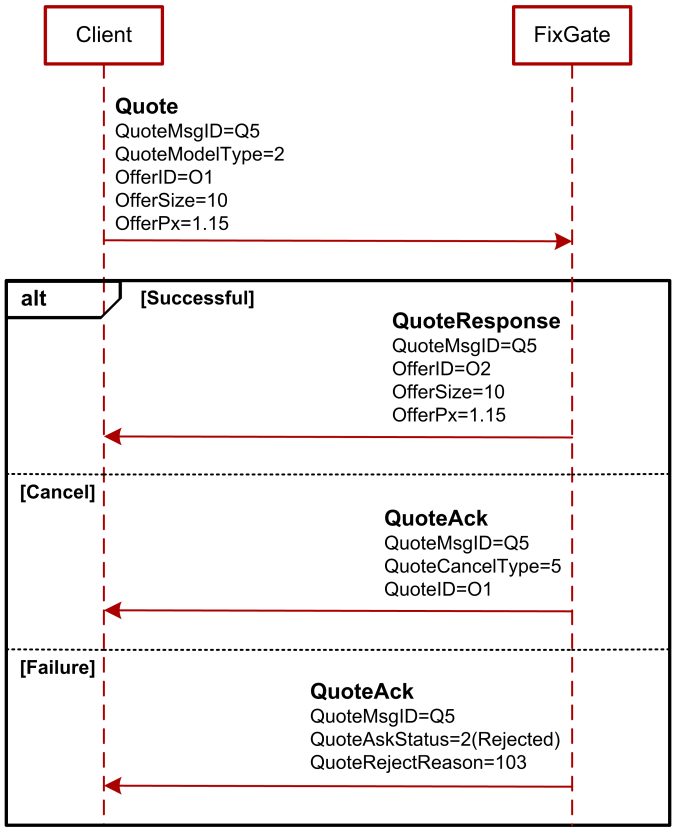

Figure 5. Diagram. Changing one quote

5.2.3.2. Changing two quotes via a single message

The client side sends message **Quote** with QuoteModelType=2 (change quote) containing some new quote parameters into the system. On successful change, the system either replies with two messages **QuoteResponse** (two new quotes added), or with two messages **QuoteAck** (two quotes deleted from order-book). Rejection of one quote will result in rejection of the second quote as well, and a single message **QuoteAck** with specified erroneous quote (1 - Bid, 2 - Offer) in field Side will be sent by FixGate in reply. When both quotes have been rejected due to the same reason, FixGate will reply with a single message **QuoteAck** with Side=3 (All quotes).

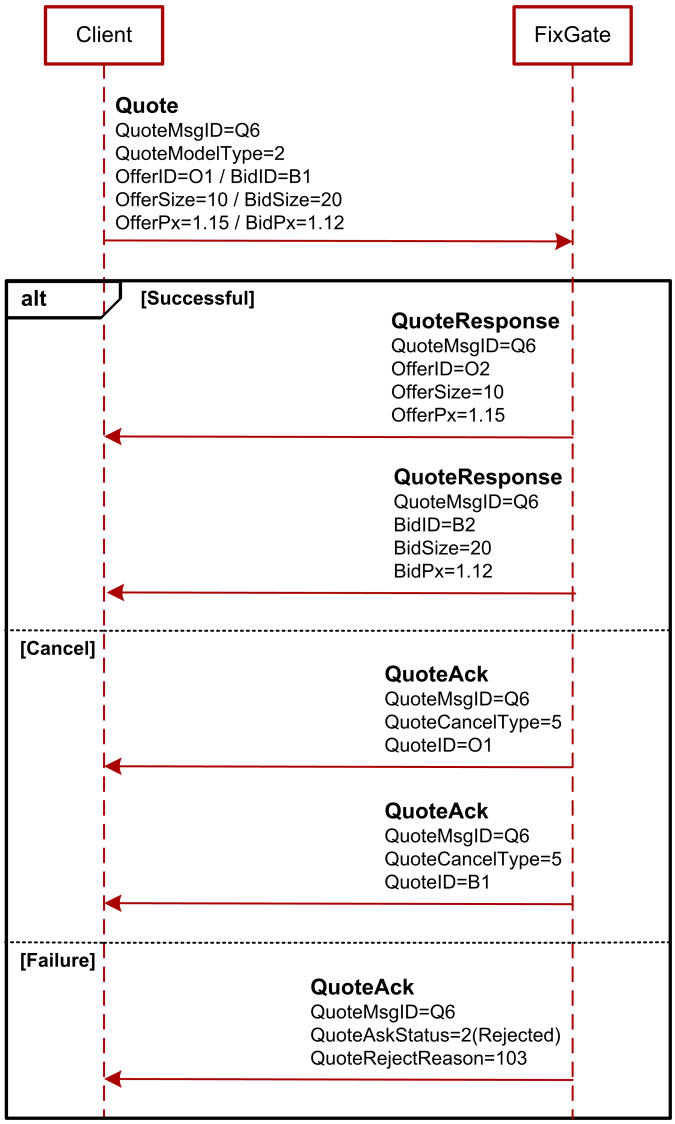

Figure 6. Diagram. Changing two quotes with single message

5.2.4. Matching quotes into a trade

Within IQS, indicative quotes are matched in the Order Driven Market mode. When two indicative quotes of opposite direction match, they perform an indicative trade. There are two sides behind each indicative trade, i.e. Initiator (initially added quote), and Contractor (the matching quote, which was added later and performs the indicative trade).

Every stage of matching indicative quotes is indicated by indicative trade status change.

Two negotiated orders are added into SPECTRA, one by one, that perform a trade. Finally, the process of matching two quotes into trade is considered to be completed.

Every stage of matching indicative quotes is indicated by indicative trade status change.

Below is the quote matching algorithm, step by step:

- After two indicative quotes are matched, an indicative trade performs within IQS, and the message **ExecutionReport** with Status=0 (being processed) will be sent to the Client side.
- A negotiated order from the Contractor side based on the indicative trade parameters is added into SPECTRA. Upon adding the order, all necessary checks and verifications are applied, including the one for collateral sufficiency. After the order is added, the message **ExecutionReport** with Status=1 (wait for confirmation from Initiator) will be sent to the Client side.
- Within a certain time frame (specified by administrator), IQS is awaiting for trade confirmation from the Initiator (see Section 5.1.3, “ExecutionReport (output)” for details). After confirmation is received, the message **ExecutionReport** with Status=2 (confirmation from Initiator received) will be sent to the Client side. A negotiated order from the Initiator side based on the indicative trade parameters is added into SPECTRA.

- After two orders are matched into trade within SPECTRA, the message **ExecutionReport** with Status=4 (trade performed) will be sent to the Client side.
- Once there any error occurs on matching the quotes, or, for any reason, the trade cannot be confirmed, the message **ExecutionReport** with Status=3 (trade not performed due to an error) will be sent to the Client side, containing the appropriate IQS error code in field 'RejectReason'.

5.2.4.1. Trade confirmation

To confirm a trade, the Initiator side is to send message **ExecutionReport** containing the indicative trade ID in reply to message **ExecutionReport** with indicative trade status '1' (wait for confirmation from Initiator). On successful confirmation, the system will send message **ExecutionAck** with ExecAckStatus=1(Accepted), and the indicative trade status will be set to 2 (confirmation from Initiator received). On unsuccessful confirmation, the system will send **ExecutionAck** with ExecAckStatus=2(Rejected), and the indicative trade status will be set to 3 (trade not performed due to an error).

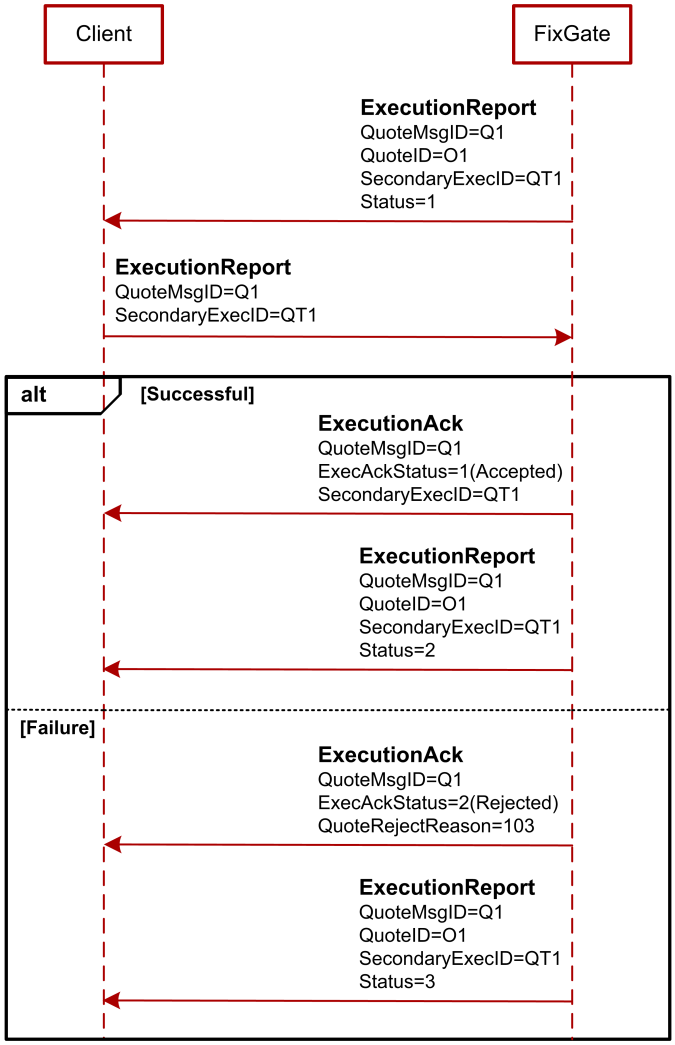

Figure 7. Diagram. Indicative trade confirmation

5.2.4.2. Matching quote type 'Day'

Quote type 'Day' - remains in order-book after being partially matched.

The client side sends message **Quote** with TimeInForce=0 into the system. Later, the quote will be fully matched into indicative trade (message **ExecutionReport** with Status=0). Below is the diagram displaying consequence of the messages sent, where the Client side is Contractor, with automatic confirmation of indicative trades.

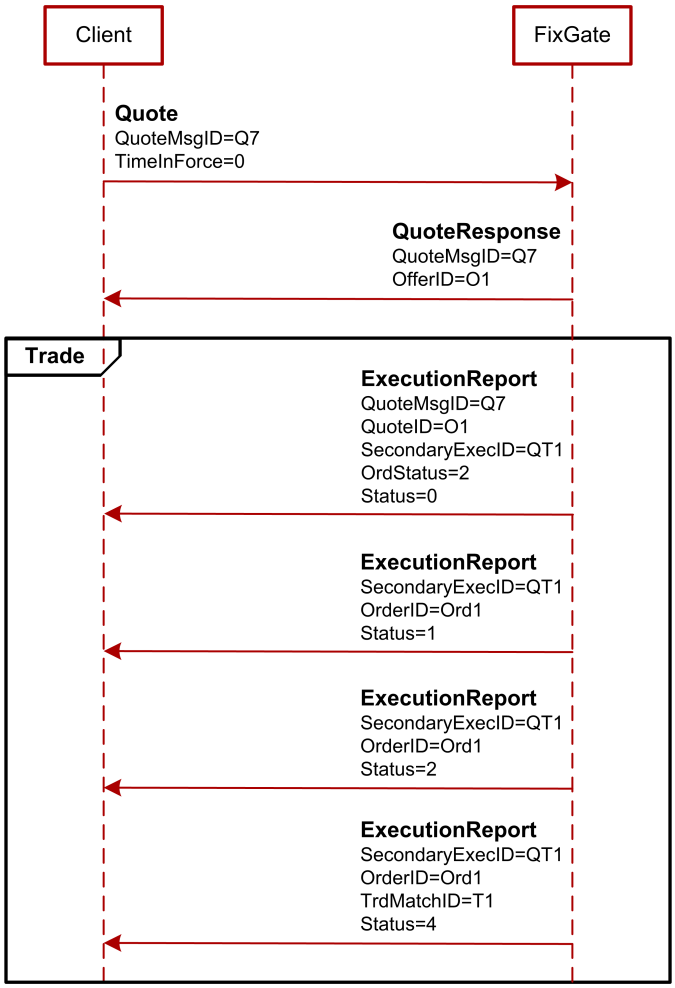

Figure 8. Diagram. Matching quote type 'Day'

5.2.4.3. Matching quote type IOC

Quote type IOC - cancels after auction end.

The client side sends message **Quote** with TimeInForce=3 into the system. Later, the quote will be partially matched into indicative trade (message **ExecutionReport** with Status=0), and its residual will be cancelled (messages **QuoteAck**with QuoteCancelType=5). Below is the diagram displaying consequence of the messages sent, where the Client side is Contractor, with automatic confirmation of indicative trades.

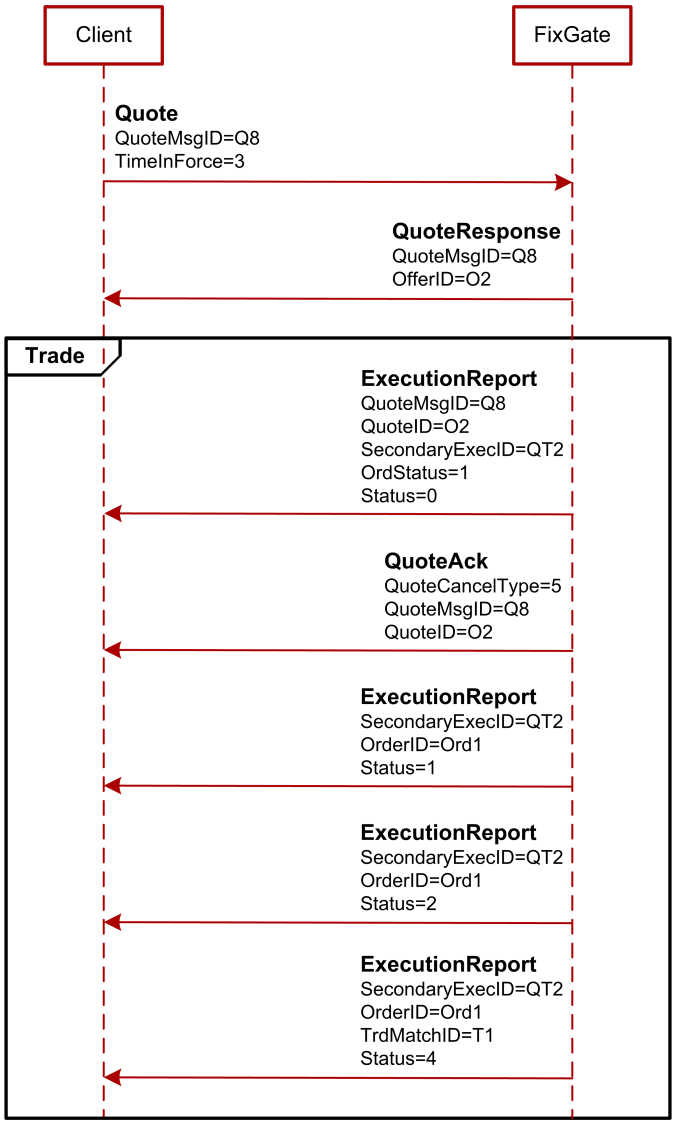

Figure 9. Diagram. Matching quote type IOC

5.3. Flood Control

The control system of clients' application flood control is a part of the FixGate. It restricts client's application to send more transactions per time unit (for single fix session) than it is stated in the connection agreement. At present moment, you can acquire login with 30, 60, 90, etc. messages per second. If you exceed the limit of messages, the control system sends the user a reply message (Reject) of the following structure:

Tag	Field	Manda-tory	Type	Details
<Group Header>		Y		Message type '3'.
373	SessionRejectReason	N	Int32	=7100 (Flood control)
58	Text	N	String255	Rejection reason details. Message text format: "penalty_remain=%d;queue_size=%d;message=%s", where • Time in milliseconds after which the next message from this user will be successfully received. • queue_size - Number of messages for a single user. • message - error message text.
<Group Trailer>		Y		

The number of messages for the elapsed second is estimated while receiving every single message. Thus, if a user constantly sends requests with the frequency greater than it is allowed, then his messages will not be processed at all.

5.4. General System Error

In case of a system-level error in delivering and processing a message, the Client side receives message **Reject** containing the error details:

Tag	Field	Mandatory	Type	Details
<Group Header>		Y		Message type '3'.
373	SessionRejectReason	N	Int32	=7101 (System error)
58	Text	N	String255	Rejection reason details. Message text format: "code=%d;message=%s".
<Group Trailer>		Y		