# Plaza-2 gateway for OTC System of Derivatives Market

**version 1.0**

**Moscow 2022**

# Table of Contents

# History of changes

| Date | Changes |
|---|---|
| 19.08.2021 | Changes applied:<br>• Removed replication streams with the ‘IQS_’ prefix.<br>• The ‘AddQuote’, ‘DelQuote’, ‘DelUserQuotes’, ‘MoveQuote’, ‘ConfirmQDeal’ commands were removed. |
| 09.07.2021 | Changes applied:<br>• Stream 'IQS_REFDATA_REPL':<br>○ The 'enforce_ims_half_netting' field has been added to the 'fut_sess_contents' and 'fut_instruments' tables. |
| 14.04.2021 | Changes applied:<br>• Removed 'IQS_FUTCOMMON_REPL' and 'IQS_OPTCOMMON_REPL' streams. Use the 'IQS_COMMON_REPL' stream instead.<br>• Removed 'IQS_FUTINFO_REPL' and 'IQS_OPTINFO_REPL' streams. Use the 'IQS_REFDATA_REPL' stream instead.<br>• Removed the streams 'IQS_FUTAGGR5_REPL', 'IQS_FUTAGGR20_REPL', 'IQS_FUTAGGR50_REPL', 'IQS_OPTAGGR5_REPL', 'IQS_OPTAGGR20_REPL' and 'IQS_OPTAGGR50_REPL'. Use the 'IQS_AGGR5_REPL', 'IQS_AGGR20_REPL', and 'IQS_AGGR50_REPL' streams instead.<br>• Stream 'IQS_REFDATA_REPL':<br>○ Removed the 'd_start' field from the 'fut_sess_contents', 'fut_instruments' and 'opt_sess_contents' tables.<br>○ Table 'prohibition' now contain field 'xprohibition_id'.<br>○ Starting with version 6.9, the 'prohib_id' field in the 'prohibition' table is deprecated and will be removed in version 6.15.<br>○ Starting with version 6.9, the 'usd_online' table is deprecated and will be removed in version 6.15. |
| 22.01.2021 | Changes applied:<br>• Stream 'IQS_REFDATA_REPL':<br>○ Table 'dealer' now contain field 'order_allowed_in_morning_session'.<br>• The IQS_SRV (IQS Proxy) service for access to the IQS subsystem will not be used since version SPECTRA 6.7. Now the OTC_SRV service (OTC Proxy) will be used to send commands to the both subsystems (IQS and RFS). |
| 20.10.2020 | Changes applied:<br>• Stream 'IQS_REFDATA_REPL':<br>○ Table 'user' now contain field 'password_expiration_date'. |
| 17.08.2020 | Changes applied:<br>• Added new stream 'IQS_COMMON_REPL'. Combines 'IQS_FUTCOMMON_REPL' and 'IQS_OPTCOMMON_REPL' streams.<br>• The 'IQS_FUTCOMMON_REPL' and 'IQS_OPTCOMMON_REPL' streams in SPECTRA 6.5 are deprecated.<br>• Added new streams 'IQS_AGGR5_REPL', 'IQS_AGGR20_REPL', 'IQS_AGGR50_REPL'. Combining relevant futures and options streams.<br>• Added new stream 'IQS_REFDATA_REPL'. Combines 'IQS_FUTINFO_REPL' and 'IQS_OPTINFO_REPL' streams.<br>• The 'IQS_FUTINFO_REPL' and 'IQSS_OPTINFO_REPL' streams in SPECTRA 6.5 are deprecated, the changes should be found in the 'IQS_REFDATA_REPL' stream description.<br>• Stream 'IQS_REFDATA_REPL':<br>○ The 'code_vcb', 'old_kotir', 'd_pg', 'last_cl_quote' fields have been removed from the 'fut_sess_contents' table.<br>○ The 'code_vcb', 'is_foreign' fields have been removed from the 'fut_vcb' table.<br>○ The 'code_vcb', 'old_kotir', 'd_pg', 'd_exp', 'exec_name', 'last_cl_quote' fields have been removed from the 'fut_instruments' table. |

| Date | Changes |
|---|---|
| | ○ The 'go_ratio' field have been removed from the 'dealer' table.<br>○ The 'code_vcb' field have been removed from the 'prohibition' table.<br>○ Table 'fut_margin_type' now contain field 'operator_input'.<br>○ The 'code_vcb', 'old_kotir', 'd_pg', 'last_cl_quote', 'bgo_c', 'bgo_nc', 'bgo_buy' fields have been removed from the 'opt_sess_contents' table.<br>○ The 'code_vcb' field have been removed from the 'opt_vcb' table.<br>○ Table 'option_series' now contain field 'm_bach'. |
| 11.06.2020 | Changes applied:<br>• Stream 'IQS_FUTINFO_REPL':<br>○ Table 'fut_vcb' now contain fields 'negative_prices', 'option_model'.<br>• Stream 'IQS_OPTINFO_REPL':<br>○ Table 'opt_vcb' now contain fields 'negative_prices', 'option_model'.<br>○ Table 'option_series' now contain fields 'a-s_black', 'a-s_bach'. |

# 1. Introduction

## 1.1. Document purpose

The document describes various information, and aimed to help clients in architecting and developing software application to access the OTC System of the Derivatives Market via the Plaza-II gateway. The document contains a brief overview of the OTC System of the Derivatives Market, as well as detailed description of transmitted data (i.e. replication streams and tables) and directive commands.

For more information about configuration, operation, installation and setup of the Plaza-II software, please apply to **p2gate_en.pdf** [http://ftp.moex.com/pub/ClientsAPI/Spectra/Docs/]. Also, you can find CGate API operation details described in **cgate_en.pdf** [http://ftp.moex.com/pub/ClientsAPI/Spectra/Docs/].

## 1.2. Target group

This document is intended for business-analysts, system architects and developers, taking part in architecting and developing software for accessing the OTC System of the Derivatives market using the Plaza-2 gateway.

## 1.3. Terms and definitions

This document contains the following terms, definitions and acronyms:

| Term | Definition |
|---|---|
| LC | liquidity consumer |
| LP | liquidity provider |
| OTC | OTC System of Derivatives Market |
| RFS | Subsystem RFS |
| Order | A trading instruction added into the SPECTRA trading system |
| Indicative Quote (or simply quote) | A trading instruction added into the OTC system without collateral verification |
| Indicative trade | A trade performed as a result of matching indicative quotes within OTC system. |
| Trade | A trade performed as a result of matching orders within SPECTRA trading system. |

# 2. A brief overview of OTC System

OTC System is aimed for high-volume liquidity attraction and allows trading members to enter request for liquidity and take the liquidity in case of matching offer. Main features are:

- Minimal initial margin is required when requesting the liquidity and entering quotes. Full margin check is realized before trade execution.
- Minimizing execution risk: there is an option to reject the trade.
- There is an option to quote large volumes without affecting the market. Quotes are visible for interested participants only.
- Trades are made with the whole volume. There is no partial trade filling.

## 2.1. Trade counterparties and system user roles

There are two counterparties in the trade:

- active (taker) – trading member, whose quote executed the trade being added to order book;
- passive (maker) – trading members, whose quote was already in order book when trade was executed.

Trading members may fulfill the following roles when using RFS:

- **LC** (liquidity consumer) – active counterparty in trade execution. Consumer initiates request for liquidity and in case of matching offer takes the liquidity.
- **LP** (liquidity provider) – passive counterparty in trade execution. Liquidity providers see requests for liquidity and may respond with their quotes on requests.

All trading members and their clients may act as liquidity consumers and may place requests for liquidity.

Liquidity provider role (LP) is assigned to trading members and their clients based on trading member request. One trading member can have several clients with LP role. Client may be LP and LC simultaneously. Liquidity provider list is available in 'lp_users' table of RFS_INFO_REPL stream in gateway.

## 2.2. Requests for liquidity

Liquidity consumer may place request in OTC system to open liquidity stream. Liquidity providers will add their quotes in this stream. LC should specify the following parameters in the request:

| Parameter | Description |
|---|---|
| Instrument | Instrument identifier |
| Volume | Amount of instrument. Volume has minimal limit depending on base contract.. User cannot specify volume less than that minimal limit. List of minimal allowed values for base contracts is available in 'base_contract_min_amount' of RFS_INFO_REPL stream in gateway. |
| Quote direction | Specifies the direction of the quotes in the stream – buy, sell or bidirectional quotes. If liquidity stream is bidirectional, both bidirectional and unidirectional quotes are allowed. |
| Stream durability | The number of time when stream is available. Minimal durability is 30 seconds. Maximum durability is by the end of current trading session. |
| Minimal provider quotes life time (Speed bump) | Provider quotes change frequency limit in order for liquidity consumer to have enough time to respond. Amount of time when move or cancel operations are prohibited after entering the quote. Possible values are: 0.2, 0.5, 1, 3 seconds. This parameter doesn't restrict move transactions which make price better. |
| Automatic trade confirmation by providers. | Specifies that providers may place only automatically confirming quotes in the stream. Provider has no option to reject trade in this case. |

To create the stream RfsCreateStream command is used. When stream is created, base limit check of liquidity provider is performed. Limit sufficiency check is always less than initial margin in Spectra risk module. If limit is not sufficient, stream open request is rejected.

Open streams are visible for all liquidity providers. Stream information is available in 'streams_log' and 'streams' tables of RFS_USERMARKETDATA_REPL data stream in the gateway.

Consumer may delete his streams by RfsDeleteStream command.

Stream may be deleted by the system automatically:

- in case of trade execution;

- if trade was not executed because of liquidity consumer collateral insufficiency;
- stream life time finished;
- trading session finished.

## 2.3. Quotes entering by providers

After stream creation by liquidity provider consumers may enter their quotes in the stream. Quotes are placed on the whole volume specified in consumers' requests. Provider should specify the following parameters in their quotes:

| Parameter | Description |
|---|---|
| Stream number | Stream identifier in which provider enters his quote |
| Client code | Client on behalf of which provider enters the quote |
| Quote direction | Buy, sell or bidirectional quote. If liquidity stream is bidirectional, provider may enter both bidirectional and unidirectional quotes. |
| Buy price | Must be specified if direction is 'sell' or 'bidirectional quote' |
| Sell price | Must be specified if direction is 'buy' or 'bidirectional quote' |
| Quote life time | Quote life time cannot be less than minimal quote life time, and cannot be greater than stream duration. By default quotes live till liquidity stream is closed. |
| Trade automatic confirmation (auto_confirm) | Provider cannot reject trade if auto_confirm flag is on. This parameter is required if stream has corresponding automatic confirmation option set. Quotes with automatic confirmation have greater priority among quotes with the same price level for trade execution. |

To enter the quote RfsQuoteUpdate is used. One or two quotes may be placed in one transaction.

When entering the quote for futures or multi-leg instrument specified quote price is checked for price limits. Also cross-trade check and trading session activity check are performed for all instruments. When entering quote base limit check of liquidity provider is performed. Limit sufficiency check is always less than initial margin in Spectra risk module. If limit is not sufficient, stream open request is rejected.

Quotes placed in the stream are visible for consumer which created the stream and for providers who placed their quotes (filtered by owner). Only one quote of each direction is active (shown) in every time moment – quote with best buy/sell price. Quote with automatic confirmation flag is on becomes active in case of equal price among different quotes. In other cases quote with earliest registration time becomes active. Quote data is translated in 'quotes_log' (all) and 'quotes' (active) tables of RFS_USERMARKETDATA_REPL market data stream in the gateway.

Provider may change and delete quotes during liquidity stream life time, but not more often than it is specified in liquidity consumer request (speed bump). This rule does not restrict quotes which make price better. Only client who placed the quote can change and delete the quote.

RfsQuoteUpdate command changes the quote. The following parameters may be changed:

- quote price;
- quote life time (life time starts to count from time of change);
- automatic confirmation flag;
- external quote number.

RfsDelLPQuotes command deletes the quote.

Quote may be automatically deleted by the system:

- when quote life time is finished;
- if liquidity provider declined trade;
- if liquidity provider doesn't have enough money;
- stream was deleted;
- trading session finished.

## 2.4. Liquidity consumer's responce for quotes

Liquidity consumer always sees best (active) quote. To answer it liquidity consumer enters counter quote into the stream with FOK fulfillment type and with requested volume specified. RfsTakeQuote command creates responce for quote.

## 2.5. Indicative trade confirmation

In OTC system there is a step during trade execution when passive counterparty may reject or confirm trade for entered quote (last look). Penalty points are applied if trading member rejects trade (see Section 2.8, “Penalties for indicative trade refusal” for details). Trading member may skip quote confirmation step if he sets up “auto_confirm” quote flag, in this case quote is considered confirmed by default. In case of “last look” step passive counterparty should confirm indicative trade.

To confirm indicative trade when indicative trade status changes to '1' (passive counterparty confirmation pending) ConfirmQDeal (RfsConfirmQDeal)) command should be issued specifying indicative trade identifier. Answer time is limited, timeout is considered as trade rejection.

## 2.6. Prohibitions

System allows to specify additional restrictions for liquidity consumers’ operations – prohibitions. There are following prohibition types:

- 32 - Liquidity stream request without automatic confirmation prohibition.
- 64 - Liquidity stream request prohibition.
- 128 - Trade execution prohibition in case of less than n providers entered their quotes into the stream, where n is the parameter specified in RFS settings.
- 256 - Prohibition for liquidity stream request with quotes life time restrictions(speed bump).

Prohibitions are set in SPECTRA trading system via p2gate gateway with standard FutChangeClientProhibit and OptChangeClientProhibit commands. See **p2gate_ru.pdf** [http://ftp.moex.com/pub/ClientsAPI/Spectra/Docs/] for more details.

## 2.7. Consumer and provider quotes matching

Only quotes of consumers and providers could be matched into trades in liquidity stream. Providers’ quotes matching is not possible. Indicative trade is created when quotes are matched in OTC System. Then two orders on behalf of liquidity consumer and liquidity provider are placed in SPECTRA trading system. After matching of these two orders final trade is executed.

Matching algorithm is the following:

- After liquidity provider and liquidity consumer quotes matching indicative trade is created in the OTC system. Indicative trade status is '0' (indicative trade is in processing).
- Order is place on behalf of liquidity consumer in SPECTRA trading system with order parameters taken from indicative trade. If consumer’s trade is rejected by SPECTRA trading system, indicative trade in OTC System finishes with '3' status (trade was not executed because of error), and liquidity stream is closed with '2' result (trade was not executed because of not enough money on LC side). In case of successful order placement indicative trade changes status to '1' (passive counterparty confirmation pending).
- OTC System during certain amount of time (specified by administrator) waits for providers’ confirmation. This step may be skipped when providers’ quote has 'auto_confirm' option set. If provider rejects trade (haven’t confirmed indicative trade), then indicative trade in OTC System changes status to '3' (trade was not executed because of error).
- If confirmation is received (or quote has 'auto_confirm' option set), indicative trade changes its status to '2' (passive counterparty confirmed), and order on behalf of liquidity provider is placed in SPECTRA trading system. If provider’s trade is rejected by SPECTRA trading system, indicative trade in OTC System finishes with '3' status (trade was not executed because of error).
- In case of successful provider’s order placement and matching of orders in SPECTRA trading system, indicative trade finishes with '4' status (trade executed) in OTC System, and liquidity stream is closed with '1' reason (trade executed).
- If providers’ order is successfully placed in SPECTRA trading system, but orders were not matched (timeout or other reason), indicative trade in OTC System finishes with '3' status (trade was not executed because of error).

Excuted indicative trades data is translated in user_qdeal table of RFS_USERMARKETDATA_REPL stream in the gateway. Information about indicative trades’ states is translated in user_qdeal_state table of RFS_USERMARKETDATA_REPL stream in the gateway.

Information about trades executed in SPECTRA trading system is translated in user_deal and user_multileg_deal tables of RFS_USERMARKETDATA_REPL stream in the gateway.

Basically process of liquidity request is shown on the following picture:

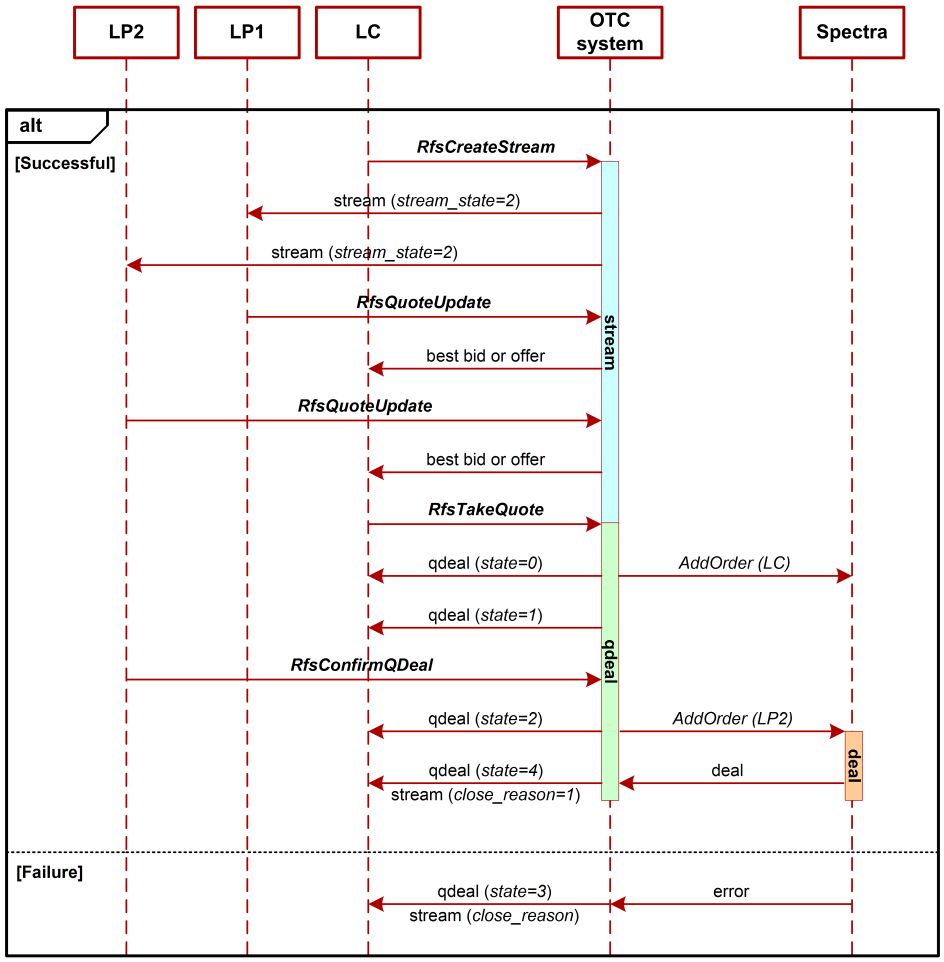


**Figure 1. Diagram. Requests for liquidity**

Liquidity consumer (LC) creates request (RfsCreateStream command), and liquidity stream is created in OTC System. Liquidity providers LP1 and LP2 subsequently enter their quotes (RfsQuoteUpdate). Quote LP2 is better, it fully matches consumers' request, and consumer answers with his quote (RfsTakeQuote). Then in OTC System indicative trade is executed, which after its confirmation by LP2 (RfsConfirmQDeal)) finishes with trade execution in SPECTRA trading system. Indicative trade finishes in OTC System and liquidity stream is closed.

In case of any error indicative trade and liquidity stream finish with error code.

## 2.8. Penalties for indicative trade refusal

Penalty is applied on trading member for indicative trades refusal and insufficient collateral when placing quotes. When penalty points exceed certain level, sanctions are applied on trading member in form of quotes cancellation and prohibition on new quotes entering during certain time. Penalties are calculated and logged by client code, underlying asset, instrument type (futures/option/multileg). Penalty level changes from 0 to 10, accurate to 5th decimal place.

There are following penalty types:

- Penalty for indicative trade confirmation rejection. By default, one reject increases penalty level by 1 in case of bidirectional quote and by 0,1 in case of unidirectional quote.
- Penalty in case of unsuccessful quote entering by active counterparty. By default one unsuccessful quote placement increases penalty level by 1 in case of bidirectional quote and by 0,1 in case of unidirectional quote.
- Penalty in case of unsuccessful quote entering by passive counterparty. By default one unsuccessful quote placement increases penalty level by 1 in case of bidirectional quote and by 0,1 in case of unidirectional quote.

There is a time interval setting in the system during which several indicative trade rejects are considered as one reject and therefore penalties are applied as for one reject.

As time passes penalty levels decrease, and finally penalty sanctions are cancelled. By default penalty level decreases 1 level per minute.

Penalty levels, rules of penalty application and cancellation are set by system administrator. Trading member penalty information is translated in RFS_FINESLEVEL_REPL and RFS_PENALTY_REPL streams of the gateway.

## 2.9. Flood control

The control system of clients' application flood control is a part of the OTC system. It restricts client's application to send more transactions per time unit (for single login on SPECTRA) than it is stated in the connection agreement. At present moment, you can acquire login with 30, 60, 90, etc. (but not more than 3000) trading transactions per second. If you exceed the limit of messages, the control system does not transmit a message into the trading system core, and sends the user a reply message (msgid=199) with the notification of denial of service of the following structure:

| Field | Type | Description |
|---|---|---|
| queue_size | u4 | Number of messages for a single user |
| penalty_remain | u4 | Time in milliseconds after which the next message from this user will be successfully received. |
| message | c128 | Error message text |

Please pay attention to the two details:

- 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.
- A reject message with can be sent in a reply to any user's message.

## 2.10. Sending commands

For sending commands, you should create a publisher with parameters 'NAME=OTC_SRV', 'category=IQS_MSG'. See the URL example below:

```
p2mq://OTC_SRV;category=IQS_MSG;name=srvlink;timeout=5000;scheme=|FILE|otc_messages.ini|message
```

where otc_messages.ini — an INI file which contains scheme of the commands from CGate distribution pack.

If you need to receive replies to the messages sent, you should specify the flag 'CG_PUB_NEEDREPLY ' within the message sending function, and create a listener of type 'p2mqreply'.

In case of the message delivery and handling errors, the client receives either sending message function error or the 'system error' (msgid=200) message in return.

| Field | Type | Description |
|---|---|---|
| code | u4 | Return code |
| message | c255 | Message body |

Please note that the 'system error' message can be received in reply to any business-logic command.

# 3. Replication scheme

## 3.1. Stream RFS_INFO_REPL - RFS References

Tables:

- lp_users - Liquidity provider catalog
- base_contract_min_amount - Minimum allowable volume

### 3.1.1. Table lp_users: Liquidity provider catalog

**Table 1. Fields of table lp_users**

| Field | Type | Description |
|---|---|---|
| replID | i8 | Service field of the replication subsystem |
| replRev | i8 | Service field of the replication subsystem |
| replAct | i8 | Service field of the replication subsystem |
| lp_code | c20 | Liquidity provider login |
| client_code | c7 | Client code |

### 3.1.2. Table base_contract_min_amount: Minimum allowable volume

The table contains restrictions on the minimum allowable quotes.

**Table 2. Fields of table base_contract_min_amount**

| Field | Type | Description |
|---|---|---|
| replID | i8 | Service field of the replication subsystem |
| replRev | i8 | Service field of the replication subsystem |
| replAct | i8 | Service field of the replication subsystem |
| base_contract_code | c25 | Underlying asset code |
| min_amount | i4 | Minimum volume in contracts |

Notes:

- The value '*' in the 'base_contract_code' field means that this restriction is used for all underlying contract not specified in this table.

## 3.2. Stream RFS_FINESLEVEL_REPL - Member's penalty levels

Tables:

- fines_level - Participant's penalty levels

### 3.2.1. Table fines_level: Member's penalty levels

The table contains data on penalties applied to the participant as a result of their refusal to confirm indicative trades.The penalty levels are calculated for each possible parameter, i.e. client ID, underlying asset, instrument type (futures/option/calendar spread). The penalty level is a subject to change from 0 to 10, accurate to 5th decimal place.

**Table 3. Fields of table fines_level**

| Field | Type | Description |
|---|---|---|
| replID | i8 | Service field of the replication subsystem |
| replRev | i8 | Service field of the replication subsystem |
| replAct | i8 | Service field of the replication subsystem |
| client_code | c7 | Client ID |
| code_vcb | c25 | Underlying asset code |
| isin_type | u1 | Instrument type |
| fine_level | u8 | Penalty level multiplied by $10^5$ |
| moment_ns | u8 | Calculation time, in nanoseconds |

Notes:

- Field 'isin_type' may contain the following values:

  0 Futures

  1 Option

  2 Multi-leg instrument

- In order to obtain the correct penalty value, one should multiply the value of field 'fine_level' by $10^{-5}$.

# 3.3. Stream RFS_PENALTY_REPL - Penalties log

Tables:

- penalty_log - Penalties log

## 3.3.1. Table penalty_log: Penalties log

The table contains a log of penalties applied to the participant as a result of their refusal to confirm/perform indicative trades.

**Table 4. Fields of table penalty_log**

| Field | Type | Description |
|---|---|---|
| replID | i8 | Service field of the replication subsystem |
| replRev | i8 | Service field of the replication subsystem |
| replAct | i8 | Service field of the replication subsystem |
| client_code | c7 | Client code |
| code_vcb | c25 | Underlying asset code |
| isin_type | u1 | Instrument type |
| quotes_deleted | u1 | Penalty 'Quotes deletion': 0 - none, 1 - applied |
| prohibition_applied | u1 | Penalty 'Restriction of adding quotes': 0 - none, 1 - applied |
| level_after_penalty | u8 | Penalty level (after applying penalty) multiplied by $10^5$ |
| penalty_start_ns | u8 | Applied penalty start time, in nanoseconds |
| penalty_end_ns | u8 | Applied penalty end time, in nanoseconds |
| sess_id | i4 | Trading session ID |

Notes:

- Field 'isin_type' may contain the following values:

  0 Futures

  1 Option

  2 Multi-leg instrument

- In order to obtain the correct penalty level, one should multiply the value of field 'level_after_penalty' by $10^{-5}$.

# 3.4. Stream RFS_USERMARKETDATA_REPL - Liquidity streams, indicative quotes, indicative trades and trades

Tables:

- streams_log - Liquidity streams log
- streams - Active liquidity streams
- quotes_log - Indicative quotes log
- quotes - Active indicative quotes
- user_qdeal - User's indicative trades log
- user_qdeal_state - User's indicative trades status
- user_deal - User's trades log
- user_multileg_deal - User's multi-leg trades log

- heartbeat - Server times table
- sys_events - System events table

## 3.4.1. Table streams_log: Liquidity streams log

Table contains liquidity streams changes log. Records are filtered: liquidity consumers can see only his streams, liquidity providers can see all streams, consumer information is not displayed.

**Table 5. Fields of table streams_log**

| Field | Type | Description |
|---|---|---|
| replID | i8 | Service field of the replication subsystem |
| replRev | i8 | Service field of the replication subsystem |
| replAct | i8 | Service field of the replication subsystem |
| sess_id | i4 | Trading session ID |
| stream_id | u8 | Stream unique ID |
| isin_id | u4 | Instrument unique ID |
| client_code | c7 | Client code who created the stream |
| lc_login | c20 | Client login who created the stream |
| lc_tag | c64 | Liquidity consumer tag |
| amount | u8 | Instrument amount |
| dir | u1 | Direction: 1 - Buy, 2 - Sell, 3 - Buy and Sell |
| duration_type | u8 | Stream duration |
| signs | i8 | Flags field |
| speed_bump_type | u1 | Limitations on the frequency of quotes updates by liquidity providers. It is set as the minimum lifetime of the provider quotes in seconds. |
| close_reason | u1 | Reason for closed |
| moment | t | Stream status changing time |
| moment_ns | u8 | Stream status changing time, nanoseconds since Unix epoch, UTC |
| qdeal_id | u8 | Indicative trade ID. The field is filled in when the stream is closed. |
| spectra_deal_id | u8 | Trade ID in the TS Spectra. The field is filled in when the stream is closed. |
| comment | c20 | Comment LC |
| comment_to_lp | c20 | Comment for LP |
| local_stamp | t | User's local time stamp |

Notes:

- Field 'signs' is a bit mask:

  0x1 Automatic confirmation of indicative trades by providers (0 - any quotes can be set, 1 - auto confirmation quotes only)

  0x2 Closed stream sign.

- Field 'duration_type' may contain the following values:

  0 no limit

  1 30 seconds

  2 60 seconds

  3 90 seconds

  4 120 seconds

- Field 'speed_bump_type' may contain the following values:

  0 no limit

  1 0,2 seconds

  2 0,5 seconds

  3 1 seconds

4 3 seconds

- Field 'close_reason' may contain the following values:

  1 Trade performed

  2 Trade not performed due to lack of funds at 'LC'

  3 On request LC

  4 Stream timeout

  5 Closed by the Administrator of trading

  6 Trading session closed

## 3.4.2. Table streams: Active liquidity streams

Table with active records from streams_log. In deleted records all fields are changed to 0 except replication fields. Deleted records are recycled for repeated use when new streams are added. Therefore total records in the table amount equals to maximum number of simultaneously opened streams. Filtering is the same as in streams_log table.

**Table 6. Flelds of table streams**

| Field | Type | Description |
|---|---|---|
| replID | i8 | Service field of the replication subsystem |
| replRev | i8 | Service field of the replication subsystem |
| replAct | i8 | Service field of the replication subsystem |
| sess_id | i4 | Trading session ID |
| stream_id | u8 | Stream unique ID |
| isin_id | u4 | Instrument unique ID |
| client_code | c7 | Client code who created the stream |
| lc_login | c20 | Client login who created the stream |
| lc_tag | c64 | Liquidity consumer tag |
| amount | u8 | Instrument amount |
| dir | u1 | Direction: 1 - Buy, 2 - Sell, 3 - Buy and Sell |
| duration_type | u8 | Stream duration |
| signs | i8 | Flags field |
| speed_bump_type | u1 | Limitations on the frequency of quotes updates by liquidity providers. It is set as the minimum lifetime of the provider quotes in seconds. |
| close_reason | u1 | Reason for closed |
| moment | t | Stream status changing time |
| moment_ns | u8 | Stream status changing time, nanoseconds since Unix epoch, UTC |
| qdeal_id | u8 | Indicative trade ID. The field is filled in when the stream is closed. |
| spectra_deal_id | u8 | Trade ID in the TS Spectra. The field is filled in when the stream is closed. |
| comment | c20 | Comment LC |
| comment_to_lp | c20 | Comment for LP |
| local_stamp | t | User's local time stamp |

Notes:

- Field 'signs' is a bit mask:

  0x1 Automatic confirmation of indicative trades by providers (0 - any quotes can be set, 1 - auto confirmation quotes only)

  0x2 Closed stream sign.

- Field 'duration_type' may contain the following values:

  0 no limit

  1 30 seconds

  2 60 seconds

3 90 seconds

4 120 seconds

- Field 'speed_bump_type' may contain the following values:

  0 no limit

  1 0,2 seconds

  2 0,5 seconds

  3 1 seconds

  4 3 seconds

- Field 'close_reason' may contain the following values:

  1 Trade performed

  2 Trade not performed due to lack of funds at 'LC'

  3 On request LC

  4 Stream timeout

  5 Closed by the Administrator of trading

  6 Trading session closed

## 3.4.3. Table quotes_log: Indicative quotes log

Table contains quotes log. Records are filtered: liquidity consumer (LC) can see his quotes (TakeQuote) and quotes of LPs in his streams. Information about LP is deleted from quotes. Liquidity provider can see his quotes from all streams, data about consumer is deleted from records.

**Table 7. FIelds of table quotes_log**

| FIeld | Type | Description |
|---|---|---|
| replID | i8 | Service field of the replication subsystem |
| replRev | i8 | Service field of the replication subsystem |
| replAct | i8 | Service field of the replication subsystem |
| sess_id | i4 | Trading session ID |
| isin_id | i4 | Instrument unique ID |
| isin_type | i1 | Instrument type |
| stream_id | u8 | Stream unique ID |
| quote_id | u8 | Indicative quote ID |
| amount | u8 | Quote amount, in lots |
| price | i8 | Quote price/swap price multiplied by $10^5$ |
| moment | t | Quote update time |
| moment_ns | u8 | Quote update time, nanoseconds since Unix epoch, UTC |
| dir | u1 | Direction: 1 - Buy, 2 - Sell, 3 - Buy and Sell |
| lp_code | c20 | LP code |
| client_code | c7 | Client code, who added the quote |
| client_code_lc | c7 | LC code |
| login_from | c20 | Login ID, who added the quote |
| comment | c20 | LP/LC comment |
| ext_id | u8 | External ID |
| duration_us | u8 | Quote lifetime in microseconds. 0 - no limit. |
| local_stamp | t | User's local time stamp |
| qdeal_id | u8 | Indicative trade ID |
| qdeal_price | i8 | Price of the performed indicative trade multiplied by $10^5$ |
| status | u8 | Flags field |

| FIeld | Type | Description |
|---|---|---|
| action | i1 | Operation with the quote |

Notes:

- Field 'status' is a bit mask:

  | | |
  |---|---|
  | 0x1 | Day |
  | 0x2 | Counter |
  | 0x1000 | Last record in transaction |
  | 0x100000 | Quote deleted as a result of 'move' operation |
  | 0x200000 | Quote deleted as a result of 'delete' operation |
  | 0x400000 | Quote deleted as a result of 'mass cancel' operation |
  | 0x8000000 | Multi-leg instrument quote |
  | 0x1000000000 | Quote deleted due to penalties |
  | 0x2000000000000 | Quote deleted by TimeOut |
  | 0x4000000000000 | Automatic confirmation quotes |

- Field 'isin_type' may contain the following values:

  0 Fitures

  1 Option

  2 Multi-leg instrument

- Field 'action' contains an action with the quote:

  0 Quote deleted

  1 Quote added

  2 Quote matched into an indicative trade

## 3.4.4. Table quotes: Active indicative quotes

Table contains active quotes by every liquidity stream. Active quote is the quote with best price (TOP-1). In case of equal price quote with automatic confirmation becomes active. Otherwise quote with earliest registration time becomes active..

Maximum two records are shown for every stream – best quotes for every direction. Table records are visible only for liquidity consumers (LC) and only from their streams. Update mode in this table is the same as in streams table: all data is replaced by 0 except replication field for deleted records. Deleted records are recycled for repeated usage when new quotes are added.

**Table 8. FIelds of table quotes**

| FIeld | Type | Description |
|---|---|---|
| replID | i8 | Service field of the replication subsystem |
| replRev | i8 | Service field of the replication subsystem |
| replAct | i8 | Service field of the replication subsystem |
| sess_id | i4 | Trading session ID |
| isin_id | i4 | Instrument unique ID |
| isin_type | i1 | Instrument type |
| stream_id | u8 | Stream unique ID |
| quote_id | u8 | Indicative quote ID |
| amount | u8 | Quote amount, in lots |
| price | i8 | Quote price/swap price multiplied by $10^5$ |
| moment | t | Quote update time |
| moment_ns | u8 | Quote update time, nanoseconds since Unix epoch, UTC |
| dir | u1 | Direction: 1 - Buy, 2 - Sell, 3 - Buy and Sell |
| lp_code | c20 | LP code |

| FIeld | Type | Description |
|---|---|---|
| client_code | c7 | Client code, who added the quote |
| client_code_lc | c7 | LC code |
| login_from | c20 | Login ID, who added the quote |
| comment | c20 | LP/LC comment |
| ext_id | u8 | External ID |
| duration_us | u8 | Quote lifetime in microseconds. 0 - no limit. |
| local_stamp | t | User's local time stamp |
| qdeal_id | u8 | Indicative trade ID |
| qdeal_price | i8 | Price of the performed indicative trade multiplied by $10^5$ |
| status | u8 | Flags field |
| action | i1 | Operation with the quote |

Notes:

- Field 'status' is a bit mask:

  | | |
  |---|---|
  | 0x1 | Day |
  | 0x2 | Counter |
  | 0x1000 | Last record in transaction |
  | 0x100000 | Quote deleted as a result of 'move' operation |
  | 0x200000 | Quote deleted as a result of 'delete' operation |
  | 0x400000 | Quote deleted as a result of 'mass cancel' operation |
  | 0x8000000 | Multi-leg instrument quote |
  | 0x2000000000000 | Quote deleted by TimeOut |
  | 0x4000000000000 | Automatic confirmation quotes |

- Field 'isin_type' may contain the following values:

  0 Fitures

  1 Option

  2 Multi-leg instrument

- Field 'action' contains an action with the quote:

  0 Quote deleted

  1 Quote added

  2 Quote matched into an indicative trade

## 3.4.5. Table user_qdeal: User's indicative trades log

**Table 9. Fields of table user_qdeal**

| Field | Type | Description |
|---|---|---|
| replID | i8 | Service field of the replication subsystem |
| replRev | i8 | Service field of the replication subsystem |
| replAct | i8 | Service field of the replication subsystem |
| sess_id | i4 | Trading session ID |
| stream_id | u8 | Stream unique ID |
| isin_id | i4 | Instrument unique ID |
| qdeal_id | u8 | Indicative trade ID |
| amount | u8 | Volume, number of units of the instrument in indicative trade |
| quote_id_buy | u8 | Buyer side quote ID |
| quote_id_sell | u8 | Seller side quote ID |

| Field | Type | Description |
|---|---|---|
| price | i8 | Price of the performed indicative trade multiplied by $10^5$ |
| moment | t | Time of indicative trade |
| moment_ns | u8 | Time of indicative trade, in nanoseconds |
| status_buy | u8 | Buyer side flag |
| status_sell | u8 | Seller side flag |
| ext_id_buy | u8 | Quote external ID (Buyer side) |
| ext_id_sell | u8 | Quote external ID (Seller side) |
| code_buy | c7 | Buyer side code |
| code_sell | c7 | Buyer side code |
| code_rts_buy | c7 | RTS code of the buyer company |
| code_rts_sell | c7 | RTS code of the seller company |
| active_side | u1 | Contractor side |
| login_buy | c20 | Buyer side login |
| login_sell | c20 | Seller side login |
| comment_buy | c20 | Buyer side quote comment |
| comment_sell | c20 | Seller side quote comment |
| isin_type | i1 | Instrument type |

Notes:

- Fields 'status_buy', 'status_sell' are bit masks:

  0x1 Day

  0x2 Counter

  0x1000 Last record in transaction

  0x8000000 Multi-leg instrument quote

  0x20000000000 The active side in the trade

  0x40000000000 The passive side in the trade

  0x4000000000000 Automatic confirmation quotes

- Field 'active_side' defines the Contractor side. The Initiator side is required to confirm the indicative trade.

  1 Buyer side.

  2 Seller side.

- Field isin_type may contain the following values:

  0 Futures

  1 Option

  2 Multi-leg instrument

## 3.4.6. Table user_qdeal_state: User's indicative trades status

**Table 10. Fields of table user_qdeal_state**

| Field | Type | Description |
|---|---|---|
| replID | i8 | Service field of the replication subsystem |
| replRev | i8 | Service field of the replication subsystem |
| replAct | i8 | Service field of the replication subsystem |
| sess_id | i4 | RFS trading session ID |
| isin_id | i4 | Instrument unique ID |
| stream_id | u8 | Stream unique ID |
| qdeal_id | u8 | Indicative trade ID |

| Field | Type | Description |
|---|---|---|
| quote_id | u8 | Quote ID |
| id_ord | u8 | Order ID |
| state | i4 | Indicative trade status |
| error_code | i4 | RFS error code |
| spectra_error_code | i4 | SPECTRA error code |
| dir | u1 | Transaction direction: 1 - Buy, 2 - Sell. |
| client_code | c7 | Client code |

Notes:

- Field 'state' may contain the following values:

  0 Indicative trade is being processed.

  1 Wait for confirmation from Initiator side.

  2 Confirmation from Initiator side received

  3 Indicative trade not performed due to an error.

  4 Indicative trade performed.

- Field 'error_code' may contain the following values:

  1 Indicative trade not confirmed.

  2 Error adding Contractor side order

  3 Error adding Initiator side order

  4 Order not found on SPECTRA

  5 Order timeout on SPECTRA

  6 SPECTRA is unavailable

- The complete list of error codes transmitted in field 'spectra_error_code' can be found in **p2gate_en.pdf** [ftp://ftp.moex.com/pub/ClientsAPI/Spectra/CGate/docs/].

## 3.4.7. Table user_deal: User's trades log

**Table 11. Fields of table user_deal**

| Field | Type | Description |
|---|---|---|
| replID | i8 | Service field of the replication subsystem |
| replRev | i8 | Service field of the replication subsystem |
| replAct | i8 | Service field of the replication subsystem |
| sess_id | i4 | Trading session ID |
| stream_id | u8 | Stream unique ID |
| isin_id | i4 | Instrument unique ID |
| id_deal | u8 | Trade ID |
| qdeal_id | u8 | Indicative trade ID |
| amount | u8 | Volume, number of units of the instrument |
| id_ord_buy | u8 | Buyer side order ID |
| id_ord_sell | u8 | Seller side order ID |
| quote_id_buy | u8 | Buyer side quote ID |
| quote_id_sell | u8 | Seller side quote ID |
| price | i8 | Price |
| moment | t | Time of indicative trade |
| moment_ns | u8 | Time of indicative trade, in nanoseconds |
| status_buy | u8 | Buyer side flag |
| status_sell | u8 | Seller side flag |

| Field | Type | Description |
|---|---|---|
| ext_id_buy | u8 | Order external ID (Buyer side) |
| ext_id_sell | u8 | Order external ID (Seller side) |
| code_buy | c7 | Buyer side code |
| code_sell | c7 | Seller side code |
| comment_buy | c20 | Buyer side order comment |
| comment_sell | c20 | Seller side order comment |
| fee_buy | i8 | Buyer side fee on trade |
| fee_sell | i8 | Seller side fee on trade |
| login_buy | c20 | Buyer side users login |
| login_sell | c20 | Seller side users login |
| code_rts_buy | c7 | RTS code of the buyer company |
| code_rts_sell | c7 | RTS code of the seller company |
| opt_type | i1 | Option contract type |

Notes:

- Fields 'xstatus_buy', 'xstatus_sell' are bit masks:

  0x1 Day

  0x2 Counter

  0x1000 Last record in transaction

  0x8000000 Multi-leg instrument quote

- Field 'opt_type' may contain the following values:

  1 PUT.

  2 CALL.

## 3.4.8. Table user_multileg_deal: User's multi leg trades log

**Table 12. Fields of table user_multileg_deal**

| Field | Type | Description |
|---|---|---|
| replID | i8 | Service field of the replication subsystem |
| replRev | i8 | Service field of the replication subsystem |
| replAct | i8 | Service field of the replication subsystem |
| sess_id | i4 | Trading session ID |
| stream_id | u8 | Stream unique ID |
| isin_id | i4 | Multi-leg instrument ID |
| isin_id_rd | i4 | Instrument ID of the first leg |
| isin_id_rb | i4 | Instrument ID of the second leg |
| duration_days | i4 | The difference in calendar days between the dates of execution of two futures |
| id_deal | u8 | Deal ID number for multileg deals |
| id_deal_rd | u8 | Deal ID of the first leg |
| id_deal_rb | u8 | Deal ID of the second leg |
| id_ord_buy | u8 | Order ID (Buyer side) |
| id_ord_sell | u8 | Order ID (Seller side) |
| quote_id_buy | u8 | Quote ID (Buyer side) |
| quote_id_sell | u8 | Quote ID (Seller side) |
| amount | u8 | Volume, number of units of the instrument |
| price | i8 | Price of the near leg of multi-leg trade |
| rate_price | i8 | Trade rate |

| Field | Type | Description |
|---|---|---|
| swap_price | i8 | Trade swap price |
| moment | t | Trade time |
| moment_ns | u8 | Trade time, in nanoseconds |
| status_buy | u8 | Buyer side flag |
| status_sell | u8 | Seller side flag |
| ext_id_buy | u8 | External ID (Buyer side order) |
| ext_id_sell | u8 | External ID (Seller side order) |
| code_buy | c7 | Buyer side code |
| code_sell | c7 | Seller side code |
| comment_buy | c20 | Buyer side order comment |
| comment_sell | c20 | Seller side order comment |
| login_buy | c20 | Buyer side user login |
| login_sell | c20 | Seller side user login |
| code_rts_buy | c7 | RTS code of the buyer company |
| code_rts_sell | c7 | RTS code of the seller company |

Notes:

- Fields 'status_buy', 'status_sell' are bit masks:

  0x1 Day

  0x2 Counter

  0x1000 Last record in transaction

  0x8000000 Multi-leg instrument quote

## 3.4.9. Table heartbeat: Server times

In this table, data are accrued from the trading system core in a specified period of time, so that the data can be used to obtain a timing reference (for example, to verify that all trades are received on time). All records added into the table will be later deleted during the non-trading time (night time).

**Table 13. Fields of table heartbeat**

| Field | Type | Description |
|---|---|---|
| replID | i8 | Service field of the replication subsystem |
| replRev | i8 | Service field of the replication subsystem |
| replAct | i8 | Service field of the replication subsystem |
| server_time | t | Server date and time |

## 3.4.10. Table sys_events: Table of system events

**Table 14. Fields of table sys_events**

| Field | Type | Description |
|---|---|---|
| replID | i8 | Service field of the replication subsystem |
| replRev | i8 | Service field of the replication subsystem |
| replAct | i8 | Service field of the replication subsystem |
| event_id | i8 | Event unique ID |
| sess_id | i4 | Trading session number |
| event_type | i4 | Event type |
| message | c64 | Event description |

Notes:

- Possible event types:

  event_type = 1
  message = "session_data_ready"

All data from the clearing system have been loaded into the trading system

event_type = 2
message = "intraday_clearing_finished"
All clearing procedures have been finished in the intraday clearing session

event_type = 4
message = "intraday_clearing_started"
Intraday clearing session has started

event_type = 5
message = "clearing_started"
Main clearing session has started

event_type = 6
message = "extension_of_limits_finished"
Limits have been extended

event_type = 8
message = "broker_recalc_finished"
Funds have been recalculated after intraday clearing session

event_type =10000
message = "rfs_session_inited"
RFS trading session has been scheduled

event_type =10001
message = "rfs_session_started"
RFS trading session is in progress

event_type =10002
message = "rfs_session_suspended"
RFS trading session has been suspended (due to a critical internal RFS error)

# 4. Commands description

## 4.1. RfsCreateStream - Create liquidity stream

Message type: 400

Reply message type: 250

**Table 15. Input data**

| Parameter name | Type | Default value | Description |
|---|---|---|---|
| broker_code | c4 | "" | BF code |
| client_code | c3 | | Client code |
| isin_id | u4 | | Instrument ID |
| amount | u8 | | Instrument amount |
| dir | u1 | | Direction: 1 - Buy, 2 - Sell, 3 - Buy and Sell |
| duration_type | u1 | | Stream duration |
| auto_confirm | u1 | | Automatically confirmed quotes only |
| speed_bump_type | u1 | | Limitations on the frequency of quotes updates by liquidity providers. It is set as the minimum lifetime of the provider quotes in seconds (selected from the list of predefined values). |
| comment_to_lp | c20 | | Comment for LP |
| comment | c20 | | Comment LC |
| ext_id | u8 | | Stream external ID |
| local_stamp | t | | Time stamp |

**Table 16. Output data**

| Parameter name | Type | Default value | Description |
|---|---|---|---|
| reply_code | i4 | "" | Return code |
| message | c255 | | Message text |
| stream_id | u8 | "" | Liquidity stream ID |

Return codes:

0 success

Other value error

Notes:

- The 'duration_type' parameter can take the following values:

  0 no limit

  1 30 seconds

  2 60 seconds

  3 90 seconds

  4 120 seconds

- The 'auto_confirm' parameter can take the following values:

  0 any quotes

  1 automatically confirmed quotes only

- The 'speed_bump_typ' parameter can take the following values:

  0 no limit

  1 0,2 seconds

  2 0,5 seconds

3 1 seconds

4 3 seconds

## 4.2. RfsDeleteStream - Delete liquidity stream

Message type: 401

Reply message type: 251

**Table 17. Input data**

| Parameter name | Type | Default value | Description |
|---|---|---|---|
| broker_code | c4 | "" | BF code |
| stream_id | u8 | "" | Liquidity stream ID |
| local_stamp | t | | Time stamp |

**Table 18. Output data**

| Parameter name | Type | Default value | Description |
|---|---|---|---|
| reply_code | i4 | "" | Return code |
| message | c255 | | Message text |

Return codes:

0 success

Other value error

## 4.3. RfsQuoteUpdate - Add/Move quote by provider

Message type: 402

Reply message type: 252

The command is intended for adding (moving) quotes to the stream by liquidity providers. The command is available only for LP.

**Table 19. Input data**

| Parameter name | Type | Default value | Description |
|---|---|---|---|
| broker_code | c4 | "" | BF code |
| client_code | c3 | | Client code |
| stream_id | u8 | | Liquidity stream ID |
| auto_confirm | u1 | | Automatic confirmation of indicative trade |
| dir | u1 | | Direction |
| price_sell | i8 | | Sell quote price/swap price multiplied by $10^5$ |
| ext_id_sell | u8 | | Sell quote external ID |
| price_buy | i8 | | Buy quote price/swap price multiplied by $10^5$ |
| ext_id_buy | u8 | | Buy quote external ID |
| duration_us | u8 | | Quote lifetime in microseconds. 0 - no limit |
| comment_buy | c20 | | Comment field, reserved for developers' needs. |
| comment_sell | c20 | | Comment field, reserved for developers' needs. |
| local_stamp | t | | Time stamp |

**Table 20. Output data**

| Parameter name | Type | Default value | Description |
|---|---|---|---|
| reply_code | i4 | "" | Return code |
| message | c255 | | Message text |
| reply_code_dir | u1 | | Erroneous quote:<br>• 0 – Undefined (reply_code=0). |

| Parameter name | Type | Default value | Description |
|---|---|---|---|
| | | | • 1 – Buy quote.<br>• 2 – Sell quote.<br>• 3 – Both quotes. |
| canceled_quote_id_sell | u8 | 0 | ID of the cancelled sell quote |
| canceled_quote_id_buy | u8 | 0 | ID of the cancelled buy quote |
| quote_id_sell | u8 | | Sell quote ID |
| quote_id_buy | u8 | | Buy quote ID |

Return codes:

0 success

Other value error

Notes:

- The 'auto_confirm' field may contain the following values:

  0 do not confirm automatically

  1 confirm automatically

  2 do not change (keep the previous flag value, applicable only for move quote)

- The 'dir' field may contain the following values:

  1 buy

  2 sell

  3 buy and sell

## 4.4. RfsDelLPQuotes - Delete quote by provider

Message type: 404

Reply message type: 254

**Table 21. Input data**

| Parameter name | Type | Default value | Description |
|---|---|---|---|
| broker_code | c4 | "" | BF code |
| client_code | c3 | "" | Client code |
| stream_id | u8 | 0 | Liquidity stream ID |
| dir | u1 | 3 | Direction: 1 - Buy, 2 - Sell, 3 - Buy and Sell |
| ext_id | u8 | 0 | External ID |
| isin_id | i4 | 0 | Instrument ID |
| local_stamp | t | | Time stamp |

**Table 22. Output data**

| Parameter name | Type | Default value | Description |
|---|---|---|---|
| reply_code | i4 | "" | Return code |
| message | c255 | | Message text |
| num_quotes_canceled | u4 | "" | Number of canceled quotes |
| num_quotes_speedbumped | u4 | "" | Number of quotes not canceled due to 'Speed bump' |

Return codes:

0 success

Other value error

Notes:

- If the 'client_code' parameter not specified, then quotes will be deleted for all client accounts.
- If the 'stream_id' parameter is not equal to zero, then 'ext_id' and 'isin_id' should be equal to zero.
- If the 'ext_id' parameter is not equal to zero, then 'stream_id' and 'isin_id' should be equal to zero.
- If the 'isin_id' parameter is not equal to zero, then 'stream_id' and 'ext_id' should be equal to zero.
- The 'dir' parameter is ignored if 'ext_id' is not equal to zero.
- If the 'num_quotes_speedbumped' field is not null in the response message, then you must repeat the operation with the same parameters.

# 4.5. RfsTakeQuote - Response to quote

Message type: 405

Reply message type: 255

In response to the quotes provided by the providers, the liquidity consumer (LC) can send 'RfsTakeQuote' command (set the matching quote), thereby confirming the desire to complete a trade at the proposed price. The command is available only for LC.

**Table 23. Input data**

| Parameter name | Type | Default values | Description |
|---|---|---|---|
| stream_id | u8 | | Liquidity stream ID |
| dir | u1 | | Direction: 1 - Buy, 2 - Sell |
| price | i8 | | Price/swap price value multiplied by $10^5$ |
| ext_id | u8 | | External ID |
| comment | c20 | | Comment field, reserved for developers' needs. |
| local_stamp | t | | Time stamp |

**Table 24. Output data**

| Parameter name | Type | Default values | Description |
|---|---|---|---|
| reply_code | i4 | "" | Return code |
| message | c255 | | Message text |
| quote_id | u8 | | Quote ID |

Return codes:

0 success

Other value error

# 4.6. RfsConfirmQDeal - Confirm indicative trade

Message type: 406

Reply message type: 256

**Table 25. Input data**

| Parameter name | Type | Default values | Description |
|---|---|---|---|
| qdeal_id | u8 | | Indicative trade ID |
| local_stamp | t | | Time stamp |

**Table 26. Output data**

| Parameter name | Type | Default values | Description |
|---|---|---|---|
| reply_code | i4 | "" | Return code |
| message | c255 | | Message text |

Return codes:

0 Success

Other value error