PLAZA II gateway

version 9.12

Moscow 2026

Table of Contents

1. Introduction ... 41
1.1. Document purpose ... 41
1.2. Target audience ... 41
1.3. Abbreviations ... 41
2. SPECTRA system overview ... 42
2.1. Trading Membership ... 42
2.1.1. Clearing firms ... 42
2.1.2. Brokerage firms ... 42
2.1.3. Clients ... 42
2.1.4. System code pattern ... 42
2.1.5. Disclosure of data on participants ... 43
2.1.6. Users. How a user is linked to a trading participant ... 43
2.2. Instruments ... 43
2.2.1. Underlying assets ... 43
2.2.2. Futures ... 44
2.2.2.1. Daily futures contract with automatic prolongation ... 44
2.2.3. Options ... 45
2.2.4. Multi-leg instruments ... 45
2.2.5. Identification of instruments ... 46
2.2.6. Instruments with investor qualification restrictions ... 46
2.3. Trading operations ... 47
2.3.1. Orders — general information ... 47
2.3.2. Negotiated orders ... 48
2.3.2.1. Negotiated mode with matching by an unique code ... 48
2.3.3. Trades ... 48
2.3.4. Cross-trades ... 48
2.3.5. Specifics of trading multi-leg instruments ... 49
2.3.6. Iceberg orders ... 49
2.3.6.1. Iceberg orders in the system information streams ... 50
2.3.6.2. Iceberg order operations ... 52
2.3.6.3. Change of order ID during iceberg orders operations ... 52
2.4. Delivery of assets and exercise of options ... 53
2.4.1. Deliveries on futures ... 53
2.4.1.1. Settlement of futures contracts of derivatives market for stock market (T+1 mode) ... 53
2.4.2. Execution during the trading session (according to schedule) and at clearing ... 55
2.4.2.1. Fixation of indicative rates for expiration ... 55
2.4.3. Option exercise ... 55
2.4.3.1. Exercise of American deliverable futures options ... 55
2.4.3.2. Exercise of European cash-settled options ... 56
2.4.3.3. Option risk calculation before exercise ... 56
2.4.4. Termination of obligations under daily futures contracts with auto-prolongation (‘perpetual’ futures) ... 56
2.4.5. Trade Settlement TAS (*This service is not available in the current version*) ... 57
2.4.6. Flags applied to orders and trades ... 57
2.4.7. Trade types, created upon exercising and expiration of futures and options ... 59
2.5. Trading and clearing schedule ... 62
2.5.1. Trading schedule. Trading days. ... 62
2.5.2. Settlement price and indicative rates determination ... 62
2.5.2.1. Calculation of additional indicative margin and current operational risk values considering the settlement price ... 63
2.5.2.2. Fixation of indicative rates ... 63
2.5.3. Clearing Session (mtm) ... 63
2.5.4. Settlement Session ... 63
2.5.5. How different entities act on assigning a new Trading day ... 63
2.5.5.1. Reference data and session data ... 63
2.5.5.2. Funds and positions ... 63
2.5.5.3. Orders ... 64
2.5.5.4. Instruments ... 64
2.5.5.5. Replication streams ... 64
2.5.6. Event-sensitive scheme for data synchronizing ... 64
2.5.7. Instrument status broadcast service ... 66
2.5.7.1. Algorithm for determining instrument statuses by group statuses ... 66
2.5.8. Game and test mode trading schedule ... 67
2.5.9. Opening auction ... 68
2.5.10. Trading on weekends. ... 69
2.6. Risk management and limitation of trading operations ... 69
2.6.1. Collaterals ... 69
2.6.1.1. Margining of calendar spreads ... 70
2.6.2. Trading limits ... 70
2.6.2.1. Unified Collateral Pool ... 71

2.6.3. Limitations on trading operations and opening positions for clients ... 71
2.6.3.1. Prohibitions - general information ... 71
2.6.3.2. Automatic prohibitions ... 74
2.6.4. Position (obligation) transfer ... 75
2.6.5. Pausing trading session for extending limits of trading prices fluctuations ... 75
2.6.6. Risk parameters forecast information for trading participants ... 75
2.6.7. Blocking the brokerage part of the client fee ... 76
2.6.8. Negative prices support in SPECTRA ... 76
2.6.9. Exchange services under Directive 6681-U - Regulation of margin trading for brokerage clients ... 78
2.6.9.1. Initial margin considering the Client risk level category ... 78
2.6.9.2. Risk Coverage Standard calculation in the market method ... 78
2.6.9.3. Current operational risk per position ... 79
2.6.10. Model of participant funds accounting after transition to T+1 settlement cycle ... 79
2.6.10.1. Trading limits ... 79
2.6.10.2. Balances on accounts ... 80
2.6.10.3. Withdrawal of funds ... 81
2.6.10.4. Commissions ... 81
2.7. SMA Login (Sponsored Market Access) ... 81
2.8. Separate entities of Clearing Member and Trading Member ... 82
2.8.1. Who belongs to whom: a view inside SPECTRA ... 83
2.8.2. Separation of Members' rights ... 83
2.8.3. Management of a Trading Member obligations by a Clearing Member ... 85
2.9. Synthetic matching ... 85
2.9.1. Synthetic orders ... 86
2.9.2. Synthetic liquidity in aggregated order-books ... 87
2.10. Settlement trades ... 89
2.10.1. Reasons for settlement trades ... 89
2.10.2. Fines and fees ... 90
2.11. Options on spot assets ... 90
2.11.1. Calculation of free cash for options on spot assets ... 91
2.11.2. Amount of the premium payable/receivable at the nearest clearing session ... 91
2.11.3. Exercise of options on spot assets ... 91
2.11.4. Margining of options on spot assets ... 91
2.11.5. Prohibitions on options on spot assets ... 91
2.11.6. Margining of premium options without initial margin and NOV at the 7CC level ... 91
2.12. Trading on weekends ... 92
2.12.1. Session model. Trading periods ... 92
2.12.2. Access of brokerage firms to weekend trading ... 93
2.12.3. Instrument admission to additional weekend trading session ... 93
2.12.4. Opening auction crossing trading periods ... 94
2.12.5. Price ranges and risk rates in additional weekend trading session ... 94
2.12.6. Single calendar day data in public SPECTRA interfaces ... 95
2.12.6.1. Position snapshots ... 95
2.12.6.2. Order snapshots ... 95
2.12.6.3. Scenario for working with new data ... 95
2.12.7. Price range check for option orders during additional weekend trading session ... 96
2.13. Margin Calls ... 96
2.13.1. Margin Call fixation ... 96
2.13.2. Recalculation of Margin Calls ... 96
2.13.3. Values status during the trading day ... 96
3. Trading gate description ... 98
3.1. PLAZA II gateway. Components, installation and setup. ... 98
3.1.1. Components and architecture ... 98
3.1.2. Hardware and software requirements ... 99
3.1.2.1. Hardware requirements ... 99
3.1.2.2. Software requirements ... 99
3.1.3. Installation for Windows ... 99
3.1.4. Installation for Linux ... 104
3.1.4.1. Installation from zip archive ... 104
3.1.4.2. Installation from a deb package or rpm package ... 105
3.1.5. Developer guidelines ... 106
3.1.5.1. Usage of test examples ... 106
3.1.5.2. Distributed configurations ... 107
3.1.5.3. Recommendations for third-party companies on including the Moscow Exchange runtimes into user application when distributing the user software ... 107
3.2. Market data structure ... 107
3.2.1. Reference information ... 107
3.2.2. Trade information ... 108
3.2.3. Recovery information ... 108
3.2.4. Funds and limits information ... 109
3.2.5. Clearing information ... 109
3.2.6. Indices and rates information ... 109

3.2.7. Auxiliary information streams ... 109
3.3. Gateway usage specifics ... 109
3.3.1. Service replication fields ... 109
3.3.2. Commands ... 110
3.3.3. Flood control ... 110
3.3.4. Cancel On Disconnect ... 110
3.3.5. Replication stream sets for different login subtypes ... 111
3.3.6. Changing user password for the Trading System ... 113
3.3.6.1. Utility 'change_password' ... 114
3.3.6.2. One-time passwords ... 114
3.3.7. Partitioning of the matching ... 114
3.3.8. Stream types ... 115
3.3.9. Limiting the number of simultaneously open replication streams from one PLAZA II connection ... 116
3.3.10. Displaying information about treasurers' logins in trades/orders ... 116
3.4. Handling abnormal situations ... 116
3.4.1. Recovery on loss of connection with Exchange servers ... 116
3.4.1.1. Connection loss detection ... 117
3.4.1.2. Timeouts handling when commands/transactions are sent ... 117
3.4.1.3. Recovery algorithm ... 118
3.4.1.4. General recommendations ... 119
3.4.2. Recovery in case of the Exchange infrastructure failure ... 120
3.4.2.1. Data cleanup by streams ... 120
3.4.2.2. Possible data change in case of abnormal work of publishing services ... 120
4. Replication scheme FORTS_PUBLIC ... 121
4.1. Stream FORTS_TRADE_REPL - User's orders and trades (Type=R) ... 121
4.1.1. Data scheme ... 121
4.1.1.1. Table orders_log: Log of operations with orders ... 121
4.1.1.2. Table multileg_orders_log: Log of operations with multileg orders ... 123
4.1.1.3. Table user_deal: User trades ... 124
4.1.1.4. Table user_multileg_deal: User’s multileg orders trades ... 126
4.1.1.5. Table heartbeat: Server times table ... 127
4.1.1.6. Table sys_events: table of events ... 127
4.2. Stream FORTS_ORDLOG_REPL – anonymous orders (Type=R) ... 128
4.2.1. Data scheme ... 128
4.2.1.1. Table orders_log: Log of operations with orders ... 128
4.2.1.2. Table multileg_orders_log: Log of operations with multileg orders ... 129
4.2.1.3. Table heartbeat: Server times table ... 130
4.2.1.4. Table sys_events: table of events ... 130
4.3. Stream FORTS_DEALS_REPL – anonymous trades (Type=R) ... 131
4.3.1. Data scheme ... 131
4.3.1.1. Table deal: Trades ... 131
4.3.1.2. Table multileg_deal: Multileg trades ... 131
4.3.1.3. Table heartbeat: Server times table ... 132
4.3.1.4. Table sys_events: table of events ... 132
4.4. Stream FORTS_FEE_REPL - exchange fees and penalties (Type=AR) ... 133
4.4.1. Data scheme ... 133
4.4.1.1. Table adjusted_fee: exchange fees ... 133
4.4.1.2. Table penalty: penalties ... 134
4.4.1.3. Table FEE_info_currentday: Snapshot information ... 134
4.4.1.4. Table sys_events: table of events ... 134
4.5. Stream FORTS_FEERATE_REPL - Precise Exchange fee rates (Type=AR) ... 135
4.5.1. Data scheme ... 135
4.5.1.1. Table futures_rate: fee rates on futures and multi-leg instruments ... 135
4.5.1.2. Table option_rate: fee rates on option contracts ... 136
4.5.1.3. Table sys_events: table of events ... 136
4.6. Stream FORTS_BROKER_FEE_REPL - Brokerage fees (Type=I) ... 137
4.6.1. Data scheme ... 137
4.6.1.1. Table broker_fee: brokerage fee ... 137
4.6.1.2. Table BFEE_info_currentday: Snapshot information ... 137
4.6.1.3. Table sys_events: table of events ... 138
4.7. Stream FORTS_BROKER_FEE_PARAMS_REPL - Parameters for calculating the brokerage fee (Type=I) ... 138
4.7.1. Data scheme ... 138
4.7.1.1. Table broker_fee_params: Parameters for calculating the brokerage fee ... 139
4.7.1.2. Table sys_events: table of events ... 139
4.8. Stream FORTS_USERORDERBOOK_REPL - User orders: order-book snapshot (Type=R) ... 140
4.8.1. Data scheme ... 140
4.8.1.1. Table orders: Current futures and options order-book ... 140
4.8.1.2. Table multileg_orders: Table of active multileg orders ... 142
4.8.1.3. Table info: Order-book snapshots information ... 144
4.8.1.4. Table orders_currentday: Snapshot of active user orders at the start of the current day ... 144
4.8.1.5. Table multileg_orders_currentday: Snapshot of active user multileg orders at the start of the current day ... 146

4.8.1.6. Table info_currentday: Snapshot information ... 148
4.9. Stream FORTS_ORDBOOK_REPL - Depersonalized order-book snapshot (Type=R) ... 148
4.9.1. Data scheme ... 148
4.9.1.1. Table orders: Current order-book ... 148
4.9.1.2. Table multileg_orders: Table of active multileg orders ... 149
4.9.1.3. Table info: Order-book snapshots information ... 150
4.9.1.4. Table orders_currentday: Snapshot of active orders at the start of the current day ... 150
4.9.1.5. Table multileg_orders_currentday: Snapshot of active multileg orders at the start of the current day ... 151
4.9.1.6. Table info_currentday: Snapshot information ... 152
4.10. Stream FORTS_COMMON_REPL - Market fundamentals (Type=I) ... 153
4.10.1. Data scheme ... 153
4.10.1.1. Table common: Market fundamentals ... 153
4.10.1.2. Table sys_events: table of events ... 154
4.11. Aggregated order-book streams (Type=I) ... 155
4.11.1. Data scheme ... 155
4.11.1.1. Table orders_aggr: Aggregated order-books ... 155
4.11.1.2. Table sys_events: table of events ... 158
4.12. Stream FORTS_POS_REPL - information on positions (Type=I) ... 159
4.12.1. Data scheme ... 159
4.12.1.1. Table position: Client positions ... 159
4.12.1.2. Table position_sa: Settlement Account positions ... 160
4.12.1.3. Table info: Snapshot information ... 160
4.12.1.4. Table sys_events: table of events ... 160
4.13. Stream FORTS_PART_REPL - Information about limits (Type=I) ... 161
4.13.1. Data scheme ... 161
4.13.1.1. Table part: Limits on clients and brokerage firms ... 161
4.13.1.2. Table part_sa: Limits by Settlement Account ... 162
4.13.1.3. Table sys_events: table of events ... 162
4.14. Stream FORTS_PROHIBITION_REPL - Prohibitions (Type=R) ... 163
4.14.1. Data scheme ... 163
4.14.1.1. Table prohibition: Prohibitions ... 163
4.14.1.2. Table sys_events: table of events ... 165
4.15. Stream FORTS_REFDATA_REPL - Reference and session information (Type=R) ... 166
4.15.1. Data scheme ... 166
4.15.1.1. Table rates: Currency rates dictionary ... 166
4.15.1.2. Table fut_sess_contents: Traded instruments directory (futures) ... 167
4.15.1.3. Table fut_vcb: Traded assets directory (futures) ... 169
4.15.1.4. Table fut_instruments: Instruments dictionary ... 170
4.15.1.5. Table fut_bond_registry: Spot asset parameters directory ... 172
4.15.1.6. Table dealer: Companies directory ... 172
4.15.1.7. Table sys_messages: Trading system messages ... 174
4.15.1.8. Table opt_sess_contents: Traded instruments directory (options) ... 175
4.15.1.9. Table opt_vcb: Traded assets directory (options) ... 177
4.15.1.10. Table multileg_dict: Multileg instruments dictionary ... 177
4.15.1.11. Table opt_exp_orders: Register of requests for exercise of option ... 178
4.15.1.12. Table fut_bond_nkd: Accrued interest as of the bond futures contract expiration date ... 178
4.15.1.13. Table fut_bond_nominal: Payment of bonds' face value ... 178
4.15.1.14. Table fut_bond_isin: Directory of compliance of instruments with spot assets ... 178
4.15.1.15. Table user: System users ... 179
4.15.1.16. Table sess_option_series: Option series ... 180
4.15.1.17. Table investor: Clients directory ... 181
4.15.1.18. Table fut_margin_type: Type of margining ... 182
4.15.1.19. Table fut_settlement_account: Settlement Account ... 183
4.15.1.20. Table sma_master: SMA login binding to MASTER login ... 183
4.15.1.21. Table sma_pre_trade_check: SMA login pre-trade verification settings ... 183
4.15.1.22. Table clearing_members: Clearing Members ... 184
4.15.1.23. Table instr2matching_map: Instrument binding to Matching ID ... 184
4.15.1.24. Table fut_exec_orders: Exercise requests of daily futures contracts with auto-prolongation ... 184
4.15.1.25. Table discrete_auction: Parameters of assigned opening auctions ... 185
4.15.1.26. Table discrete_auction_base_contract: Underlying contracts assigned to the opening auction ... 185
4.15.1.27. Table trade_periods: Trading period parameters ... 185
4.15.1.28. Table session: Information about a trading day ... 186
4.15.1.29. Table brokers_base_contracts_params: Individual coefficient of IM in the context of the underlying contract and BF ... 187
4.15.1.30. Table spectra_environment: Description of the current environment ... 187
4.15.1.31. Table sys_events: table of events ... 187
4.16. Stream FORTS_MM_REPL - information on MM's obligations (Type=I) ... 188
4.16.1. Data scheme ... 188
4.16.1.1. Table mm_agreement_filter: Table with numbers, types of contracts and codes of programs for the provision of market-making services ... 188
4.16.1.2. Table fut_MM_info: MM's obligations in futures ... 189
4.16.1.3. Table opt_MM_info: MM's obligations in options ... 189

4.16.1.4. Table cs_mm_rule: Instruments for recalculating the central strike price. ... 190
4.17. Stream FORTS_CLR_REPL - clearing information (Type=AR) ... 190
4.17.1. Data scheme ... 190
4.17.1.1. Table money_clearing: Balances for clients and brokerage firms ... 191
4.17.1.2. Table money_clearing_sa: Balances by Settlement Account ... 191
4.17.1.3. Table clr_rate: Currency and Index rates ... 192
4.17.1.4. Table fut_pos: Positional state in futures as a result of clearing session ... 192
4.17.1.5. Table opt_pos: Positional state in options as a result of clearing session ... 193
4.17.1.6. Table fut_sess_settl: Futures settlement prices ... 194
4.17.1.7. Table opt_sess_settl: Options settlement prices ... 194
4.17.1.8. Table pledge_details: Pledgs details table ... 194
4.17.1.9. Table fut_pos_sa: Positional state of SA on futures as a result of clearing session ... 195
4.17.1.10. Table opt_pos_sa: Positional state of SA on options as a result of clearing session ... 195
4.17.1.11. Table option_series_settl: Settlement prices for option series ... 194
4.17.1.12. Table sys_events: table of events ... 196
4.18. Stream RTS_INDEX_REPL - online indices (Type=R) ... 197
4.18.1. Data scheme ... 197
4.18.1.1. Table rts_index: Indices ... 197
4.19. Stream FORTS_VM_REPL - Variation margin and premium (Type=I) ... 197
4.19.1. Data scheme ... 197
4.19.1.1. Table fut_vm: Variation margin on futures by positions of clients ... 197
4.19.1.2. Table opt_vm: Variation margin and premium on options in the context of client positions ... 198
4.19.1.3. Table fut_vm_sa: Variation margin on futures in the context of SA positions ... 198
4.19.1.4. Table opt_vm_sa: Variation margin for options ... 199
4.19.1.5. Table sys_events: Table of events ... 199
4.20. Stream FORTS_VOLAT_REPL - online volatility information (Type=I) ... 200
4.20.1. Data scheme ... 200
4.20.1.1. Table volat: Volatility ... 200
4.20.1.2. Table sys_events: Table of events ... 201
4.21. Stream FORTS_RISKINFOBLACK_REPL - Risk parameters for Black-Scholes model (Type=I) ... 201
4.21.1. Data scheme ... 201
4.21.1.1. Table volat_coeff: Risk parameters for Black-Scholes model ... 202
4.22. Stream FORTS_RISKINFOBACH_REPL - Risk parameters for Bachelier model (Type=I) ... 202
4.22.1. Data scheme ... 202
4.22.1.1. Table volat_coeff: Risk parameters for Bachelier model ... 202
4.23. Stream FORTS_INFO_REPL - additional reference information (Type=R) ... 202
4.23.1. Data scheme ... 202
4.23.1.1. Table currency_params: FX parameters ... 203
4.23.1.2. Table base_contracts_params: Base contracts parameters ... 203
4.23.1.3. Table futures_params: Futures parameters ... 204
4.23.1.4. Table option_series_params: Parameters for series of options ... 205
4.23.1.5. Table options_params: Options parameters ... 206
4.23.1.6. Table investor: Clients directory ... 206
4.23.1.7. Table dealer: Companies directory ... 208
4.23.1.8. Table multileg_dictionary: Multileg instruments dictionary ... 209
4.23.1.9. Table common_params: Collateral calculation parameters ... 209
4.23.1.10. Table brokers_base_contracts_params: Individual coefficient of IM in the context of the underlying contract and BF ... 209
4.23.1.11. Table sys_events: Table of events ... 210
4.24. Stream FORTS_TNPENALTY_REPL - information about Transaction fees (Type=I) ... 210
4.24.1. Data scheme ... 210
4.24.1.1. Table fee_tn: Detailed information on the number of incorrect transaction ... 211
4.24.1.2. Table fee_all: Information on the number of points accrued ... 211
4.24.1.3. Table heartbeat: Server times table ... 211
4.25. Stream MOEX_RATES_REPL - online currency rates (Type=I) ... 211
4.25.1. Data scheme ... 211
4.25.1.1. Table curr_online: Currency rates values ... 211
4.26. Stream FORTS_FORECASTIM_REPL - Risk forecast after limits extension (Type=I) ... 212
4.26.1. Data scheme ... 212
4.26.1.1. Table part_sa_forecast: Free funds for SA volume forecast ... 212
4.26.1.2. Table sys_events: Table of events ... 212
4.27. Stream FORTS_USER_REPL - Users (Type=R) ... 213
4.27.1. Data scheme ... 213
4.27.1.1. Table user: System users ... 213
4.27.1.2. Table sma_master: SMA login binding to MASTER login ... 214
4.27.1.3. Table sma_pre_trade_check: SMA login pre-trade verification settings ... 215
4.27.1.4. Table sys_events: Table of events ... 215
4.28. Stream FORTS_REJECTEDORDERS_REPL - Register of orders rejected during the clearing (Type=R) ... 216
4.28.1. Data scheme ... 216
4.28.1.1. Table rejected_orders: Register of orders rejected during the clearing ... 216
4.29. Stream FORTS_RMT_REPL - Collateral without orders and current operational risk (Type=I) ... 216
4.29.1. Data scheme ... 216

4.29.1.1. Table rmt_im: Collateral without orders and current operational risk in the context of clients 217
4.29.1.2. Table sys_events: table of events 218
4.30. Stream FORTS_SESSIONSTATE_REPL - Current trading day status (Type=I) 218
4.30.1. Data scheme 218
4.30.1.1. Table session_state: Current trading day status 218
4.30.1.2. Table sys_events: table of events 219
4.31. Stream FORTS_INSTRUMENTSTATE_REPL - Statuses of instruments for the current trading day (Type=I) 220
4.31.1. Data scheme 220
4.31.1.1. Table instrument_state: Statuses of instruments for the current trading day 220
4.31.1.2. Table sys_events: table of events 220
4.32. Stream FORTS_SECURITYGROUPSTATE_REPL - Group status of instruments for the current trading day (Type=I)..... 221
4.32.1. Data scheme 221
4.32.1.1. Table security_group_state: Group status of instruments for the current trading day 221
4.32.1.2. Table sys_events: table of events 222
5. Commands description 224
5.1. Method AddOrder - Adding orders 224
5.2. Method DelOrder - Deletion of orders 225
5.3. Method DelUserOrders - Mass cancel orders 226
5.4. Method MoveOrder - Modify orders 227
5.5. Method IcebergAddOrder - Adding iceberg orders 228
5.6. Method IcebergDelOrder - Deletion of iceberg orders 230
5.7. Method IcebergMoveOrder - Modify iceberg orders 230
5.8. Method ChangeClientMoney - Change client limits 231
5.9. Method ChangeBFMoney - Change brokerage firm limits 232
5.10. Method OptChangeExpiration - Request for the exercise of options 233
5.11. Method FuturesExecutionRequest - Exercise requests of daily futures contracts with auto-prolongation 234
5.12. Method FutChangeClientProhibit - Modify client's restrictions for futures 235
5.13. Method OptChangeClientProhibit - Modify client's restrictions for options 236
5.14. Method ExchangeBFMoney - Transfer of funds between two BFs of the same SA 237
5.15. Method OptRecalcCS - Recalculate central strike request 237
5.16. Method TransferClientPosition - Transfer client positions 238
5.17. Method OptChangeRiskParametersNextSession - Risk parameters settings for options 238
5.18. Method ChangeBFParametersNextSession - Change BF's parameters by a clearing member 239
5.19. Method ChangeClientParameters - Change parameters of client account 240
5.20. Method ChangeClientParametersNextSession - Change parameters of client account in clearing session 241
5.21. Method ChangeBFClientDefaultParametersNextSession - Change default parameters of client sections 241
5.22. Method ChangeBFClientBaseContractParametersNextSession - Changing the parameters of BF clients for the basic contract 242
5.23. Method ChangeBFLimit - Change BF trading limits 242
5.24. Method CODHeartbeat - Heartbeat message for Cancel on Disconnect Service 243
5.25. Method SetSmaPreTradeCheck - Enable pre-trade verification mode for SMA login orders 244
5.26. Method DelSmaPreTradeCheck - Disable pre-trade verification mode for SMA login orders 245
5.27. Method UserKillSwitch - Disable transactions for trading member login 245
5.28. Method SetBrokerFeeParamNextSession - Setting parameters for calculating the brokerage fee 246
5.29. Method DelOrdersByBFLimit - Request to NCC for collateral sufficiency verification of Brokerage Firm 247
5.30. Method ChangeClientNotUseNOVMode - Changing the special margining mode for premium options for a client 247
5.31. Method ChangeBFClientDefaultNotUseNOVMode - Changing the special margining mode for premium options for Brokerage Firm clients 248
5.32. Method ChangePassword - Change user password for the Trading System 249
PLAZA II data types 249
List of return codes 250

History of changes

Date	Changes
21.10.2026	Changes applied: • Stream 'FORTS_FEE_REPL': ○ Added table 'FEE_info_currentday' - snapshot information • Stream 'FORTS_BROKER_FEE_REPL': ○ Added table 'BFEE_info_currentday' - snapshot information • Stream 'FORTS_COMMON_REPL': ○ Table 'common' now contains field 'settlement_price'. ○ Removed deprecated 'settlement_price_open' field from 'common' table. • Stream 'FORTS_REFDATA_REPL': ○ Added table 'spectra_environment' - Description of the current environment. ○ The 'xlot_volume' field has been added to the 'fut_sess_contents' and 'fut_instruments' tables. ○ Starting from version 9.12, the 'lot_volume' field in the 'fut_sess_contents' table is deprecated and will be removed in version 10.3. ○ Starting from version 9.12, the 'lot_volume' field in the 'fut_instruments' table is deprecated and will be removed in version 10.3. ○ The deprecated field 'settlement_price_open' has been removed from the 'fut_sess_contents', 'fut_instruments', and 'opt_sess_contents' tables. • Stream 'FORTS_INFO_REPL': ○ Table 'futures_params' now contains field 'xlot_volume'. ○ Starting from version 9.12, the 'lot' field in the 'futures_params' table is deprecated and will be removed in version 10.3. • Added new error code: 20000-20019, 20101-20117, 20129, 20130, 30000-30002., 40000-40007.
17.09.2026	Changes applied: • Changed description for error codes: 3, 14, 26, 31 32, 35, 50, 77, 314, 315, 316, 332, 4005, 4098, 10006.
20.08.2026	Changes applied: • Changes have been made to section '2.5.1. Trading schedule. Trading days'. • Added section '2.11.6. Margining of premium options without Initial margin and NOV at the 7CC level'. • Stream 'FORTS_TRADE_REPL': ○ The 'prevorder_id' field has been added to the 'orders_log' table. • Stream 'FORTS_USERORDERBOOK_REPL': ○ The 'prevorder_id' field has been added to the 'orders' and 'orders_currentday' tables. • Stream 'FORTS_COMMON_REPL': ○ Table 'common' now contains field 'swap_rate_curr'. • Stream 'FORTS_PART_REPL': ○ Table 'part' now contains field 'premium_blocked'. ○ The deprecated fields 'vm_intercl' and 'premium_intercl' have been removed from the 'part' and 'part_sa' tables. • Stream 'FORTS_REFDATA_REPL': ○ Table 'dealer' now contains field 'not_use_nov_client_default'. ○ Table 'investor' now contains field 'not_use_nov'.

Date	Changes
	○ The deprecated fields 'inter_cl_begin', 'inter_cl_end', and 'inter_cl_state' have been removed from the 'session' table. ○ The deprecated field 'step_price_interclr' has been removed from the 'fut_sess_contents', 'fut_instruments', and 'sess_option_series' tables. ○ The deprecated 'xamount_apply' field has been removed from the 'opt_exp_orders' and 'fut_exec_orders' tables. ○ The deprecated 'fut_intercl_info' and 'opt_intercl_info' tables have been removed. • Stream 'FORTS_CLR_REPL': ○ The 'vm_settl_price', 'vm_swap_rate', 'vm_index_div' fields have been added to the 'fut_pos' and 'fut_pos_sa' tables. ○ Table 'fut_sess_settl' now contains the field 'swap_rate_curr'. ○ The 'date_money_clearing' and 'settl_datetime' fields has been added to the 'fut_pos', 'opt_pos', 'fut_pos_sa' and 'opt_pos_sa' tables. ○ The deprecated 'asset_type' and 'pay' fields have been removed from the 'money_clearing' and 'money_clearing_sa' tables. ○ The deprecated 'pos_exec' field has been removed from the 'fut_pos', 'fut_pos_sa', 'opt_pos', and 'opt_pos_sa' tables. Use the 'xpos_exec' field instead. • Stream 'FORTS_VM_REPL': ○ The 'swap_rate_curr' field has been added to the 'fut_vm' and 'fut_vm_sa' tables. • Stream 'FORTS_INFO_REPL': ○ Table 'dealer' now contains field 'not_use_nov_client_default'. ○ Table 'investor' now contains field 'not_use_nov'. • Changes applied to command scheme repository: ○ Added command 'ChangeClientNotUseNOVMode' (msgid=479) - Changing the special margining mode for premium options for a client. ○ Added command 'ChangeBFClientDefaultNotUseNOVMode' (msgid=480) - Changing the special margining mode for premium options for Brokerage Firm clients. • Added new error code: 4233, 4234.
28.04.2026	Changes applied: • Added section '2.12.7. Price range check for option orders during Additional Weekend Trading Session'. • Stream 'FORTS_COMMON_REPL': ○ Starting from version 9.6, the 'settlement_price_open' field in the 'common' table is deprecated and will be removed in version 9.12. • Stream 'FORTS_PART_REPL': ○ The 'moment' field has been added to the 'part' and 'part_sa' tables. • Stream 'FORTS_REFDATA_REPL': ○ The 'd_sess_assign' field has been added to the 'fut_sess_contents' and 'opt_sess_contents' tables. ○ In the 'sess_option_series' table, the semantics of the 'd_sess_assign' field have changed. It now represents the trading start date (calendar date from which orders for the instrument may be submitted). ○ Table 'sess_option_series' now contains field 'fixed_spot_discount_clr'. ○ Table 'fut_margin_type' now contains field 'spf'. ○ Starting from version 9.6, the 'settlement_price_open' field in the 'fut_sess_contents', 'opt_sess_contents' and 'fut_instruments' tables is deprecated and will be removed in version 9.12. ○ Starting from version 9.6, the 'ticker' field in the 'fut_sess_contents' table is deprecated and will be removed in version 9.12.

Date	Changes
	• Stream 'FORTS_MM_REPL': ○ Table 'mm_agreement_filter' now contains field 'program_code'. • Stream 'FORTS_VOLAT_REPL': ○ The 'up_prem_weekend' and 'down_prem_weekend' fields have been added to the 'volat' table. • Stream 'FORTS_INFO_REPL': ○ Table 'option_series_params' now contains field 'fixed_spot_discount_clr'. • Stream 'FORTS_RMT_REPL': ○ Table 'rmt_im' now contains field 'rmt_premium'. • Added new error code: 3003.
04.02.2026	Changes applied: • Added section '2.13. Margin Calls. • Stream 'FORTS_TRADE_REPL': ○ In the tables of own orders 'orders_log' and 'multileg_orders_log' (in the 'reason' field) and trades 'user_deal' and 'user_multileg_deal' (in the 'reason_buy' and 'reason_sell' fields), transmission of a new value has been added: ■ 11 - Balancing trade for the obligations of the Defaulting Clearing Member. • Stream 'FORTS_PART_REPL': ○ The 'margin_call_moment', 'margin_call_fix', 'margin_call_current' fields have been added to the 'part_sa' table. • Stream 'FORTS_REFDATA_REPL': ○ Table 'session' now contains field 'margin_call_fix_schedule'. • The 'server_time' field has been added to the 'sys_events' table in streams: ○ 'FORTS_TRADE_REPL' ○ 'FORTS_ORDLOG_REPL' ○ 'FORTS_DEALS_REPL' ○ 'FORTS_FEE_REPL' ○ 'FORTS_FEERATE_REPL' ○ 'FORTS_BROKER_FEE_REPL' ○ 'FORTS_BROKER_FEE_PARAMS_REPL' ○ 'FORTS_COMMON_REPL' ○ 'FORTS_POS_REPL' ○ 'FORTS_PART_REPL' ○ 'FORTS_PROHIBITION_REPL' ○ 'FORTS_REFDATA_REPL' ○ 'FORTS_CLR_REPL' ○ 'FORTS_VM_REPL' ○ 'FORTS_VOLAT_REPL' ○ 'FORTS_USER_REPL'
10.11.2025	In version 9.3 of the Spectra Trading System, the functionality of the Unified Trading Session on the Derivatives market has been implemented. Key changes: • Cancellation of intraday clearing session: ○ No trading halts during high-liquidity periods;

Date	Changes
	○ No risks associated with restarting the trading system after intraday clearing. • Trading session is conducted within one calendar day: ○ Main clearing moved to the market close time; ○ Evening additional trading session period belongs to the current trading session; ○ Changes in parameters for determining settlement prices, with settlement prices determined at 07:00 PM. • Transition to T+1 settlement cycle on the Derivatives market: ○ Changes in participant funds accounting model (planned and actual balances); ○ Settlement session at 08:00 PM on day T+1 without trading interruptions. Reports on execution results are formed; ○ Increased deadlines for Margin Calls. • Contract execution online (during trading without halts). Schedule of the Unified Trading Session: • Opening auction (if held) conducted before the Morning additional trading session (8:50 AM - 9:00 AM) • Morning additional trading session (if held) (9:00 AM - 10:00 AM) • Opening auction (if held) conducted before the Main trading session (09:50 AM - 10:00 AM) • Main trading session (10:00 AM - 07:00 PM) • Evening additional trading session (if held) (07:00 PM - 11:50 PM) • Settlement session (08:00 PM, without halts) • M-t-m clearing session (11:50 PM - 12:30 AM)
	Changes applied: • Added section '2.4.2. Execution during the trading session (according to schedule) and at clearing'. • Added section '2.5.2. Settlement price and indicative rates determination'. • Updated section '2.5.2. Intermediate clearing session'. This section is now called '2.5.3. Clearing session (mtm)'. • Updated section '2.5.3. Main clearing session'. This section is now called '2.5.4. Settlement session'. • Added section '2.6.10. Model of Participant funds Accounting after Transition to T+1 Settlement Cycle'. • Stream 'FORTS_USERORDERBOOK_REPL': ○ Removed deprecated 'logRev' and 'lifeNum' fields from 'info' table. Instead of these fields, use 'trades_rev' and 'trades_lifenum', respectively. ○ The 'publication_state' field has been added to the 'info_currentday' table. • Stream 'FORTS_ORDBOOK_REPL': ○ Removed deprecated 'logRev' and 'lifeNum' fields from 'info' table. Instead of these fields, use 'trades_rev' and 'trades_lifenum', respectively. ○ The 'publication_state' field has been added to the 'info_currentday' table. • Stream 'FORTS_REFDATA_REPL': ○ The 'expiration_time' field has been added to the 'fut_vcb' and 'sess_option_series' tables. ○ The 'settlement_price_clr' field has been added to the 'fut_sess_contents' and 'opt_sess_contents' tables. ○ New fields 'settl_price_calc_time', 'settl_sess_t1_begin', 'settl_sess_begin' and 'clr_sess_begin' have been added to the 'session' table. ○ Starting from version 9.3, the 'inter_cl_begin', 'inter_cl_end', and 'inter_cl_state' fields in the 'session' table are deprecated and will be removed in version 9.9. ○ Starting from version 9.3, the 'step_price_interclr' field in the 'sess_option_series', 'fut_sess_contents', and 'fut_instruments' tables is deprecated and will be removed in version 9.9.

Date	Changes
	○ Starting from version 9.3, the 'xamount_apply' field in the 'fut_exec_orders' and 'opt_exp_orders' tables is deprecated and will be removed in version 9.9. ○ Starting from version 9.3, the 'fut_intercl_info' and 'opt_intercl_info' tables are deprecated and will be removed in version 9.9. ○ In the 'fut_vcb' table of the 'signs' field, the purpose of the flag '0x1' was redefined. It now indicates: Scheduled execution or Execution in Clearing session: ■ '0' - Clearing session ■ '1' - Scheduled execution ○ In the 'fut_sess_contents' table of the 'signs' field, the purpose of the flag '0x80000' was redefined. It now indicates: Scheduled execution or Execution in Clearing session: ■ '0' - Clearing session ■ '1' - Scheduled execution ○ In the 'sess_option_series' table of the 'signs' field, the purpose of the flag '0x1' was redefined. It now indicates: Scheduled execution or Execution in Clearing session: ■ '0' - Clearing session ■ '1' - Scheduled execution • Stream 'FORTS_PART_REPL': ○ The 'actual_amount_of_base_currency' field has been added to the 'part' and 'part_sa' tables. ○ Starting from version 9.3, the 'vm_intercl' and 'premium_intercl' fields in the 'part' and 'part_sa' tables is deprecated and will be removed in version 9.9. • Stream 'FORTS_CLR_REPL': ○ The 'date_money_clearing', 'settl_datetime', 'in_out_netto', 'vm_mtm', 'premium_mtm', 'charges', 'charges_mtm', 'fee_fut_mtm', 'fee_opt_mtm' fields have been added to the 'money_clearing' and 'money_clearing_sa' tables. ○ Starting from version 9.3, the 'asset_type' and 'pay' fields in the 'money_clearing' and 'money_clearing_sa' tables is deprecated and will be removed in version 9.9. ○ The 'xpos_exec' field has been added to the 'fut_pos', 'opt_pos', 'fut_pos_sa' and 'opt_pos_sa' tables. ○ Starting from version 9.3, the 'pos_exec' field in the 'fut_pos', 'opt_pos', 'fut_pos_sa' and 'opt_pos_sa' tables is deprecated and will be removed in version 9.9. Use the 'xpos_exec' field instead. • Stream 'FORTS_VM_REPL': ○ The 'vm_at_settl' field has been added to the 'fut_vm', 'opt_vm', 'fut_vm_sa' and 'opt_vm_sa' tables. • Stream 'FORTS_INFO_REPL': ○ In the 'base_contracts_params' table of the 'signs' field, the purpose of the flag '0x1' was redefined. It now indicates: Scheduled execution or Execution in Clearing session: ■ '0' - Clearing session ■ '1' - Scheduled execution • Stream 'FORTS_RMT_REPL': ○ New fields 'rmt_posrisk_at_settl' and 'rmt_vm_at_settl' have been added to the 'rmt_im' table. • Removed descriptions of deprecated fields from the 'sys_events' table.
14.07.2025	Changes applied: • In the section '2.4.5. Flags applied to orders and trades' a description of new flags is added: ○ 'PerpetualFuturesExecutionPartialVoluntary (0x2000000000000000)' - a technical transaction for exiting perpetual futures with forced closure of one party's position. • Stream 'FORTS_AGGRXX_REPL': ○ Added the 'sys_events' event table.

Date	Changes
	• Stream 'FORTS_REFDATA_REPL': ○ Removed deprecated 'is_trade_evening' field from 'fut_sess_contents' table. Use the 'trade_period_access' field instead. ○ Removed deprecated 'is_trade_evening' field from 'opt_sess_contents' table. Use the 'trade_period_access' field instead. ○ Removed deprecated 'pos_transfer_begin' and 'pos_transfer_end' fields from 'session' table. ○ In the 'fut_sess_contents' and 'opt_sess_contents' tables, the 'signs' field has stopped broadcasting values: ■ 0x1 - The instrument is traded in the additional trading session ■ 0x40 - Sign of trading in the main session • Stream 'FORTS_CLR_REPL': ○ Table 'fut_pos' now has fields 'pos_enforcement_exec' and 'pay_enforcement_exec'. ○ Table 'fut_sess_settl' now contains the field 'exp_enforcement_pay'. ○ Tables 'fut_sess_settl', 'opt_sess_settl', and 'option_series_settl' now have fields 'step_price' and 'step_price_curr'.
04.07.2025	Changes applied: • Stream 'FORTS_USERORDERBOOK_REPL': ○ Added table 'multileg_orders' - Table of active multileg orders. ○ Added table 'multileg_orders_currentday' - Snapshot of active user multileg orders at the start of the current day. ○ Table 'orders_currentday' now contains the field 'match_ref'. • Stream 'FORTS_ORDBOOK_REPL': ○ Added table 'multileg_orders' - Table of active multileg orders. ○ Added table 'multileg_orders_currentday' - Snapshot of active multileg orders at the start of the current day.
17.06.2025	Changes applied: • Added section '2.2.6. Instruments with investor qualification restrictions'. • Added section '2.5.6. Instrument status broadcast service'. • Added section '2.12. Trading on weekends'. • Stream 'FORTS_USERORDERBOOK_REPL': ○ Added table 'orders_currentday' - Snapshot of active user orders at the start of the current day. ○ Added table 'info_currentday' - Snapshot information. ○ Table 'info' now has fields 'trades_rev', 'trades_lifenum'. ○ Starting from version 8.3, the 'logRev' and 'lifeNum' fields in the 'info' table are deprecated and will be removed in version 8.9. Instead of these fields, you must use the 'trades_rev' and 'trades_lifenum' fields, respectively. • Stream 'FORTS_ORDBOOK_REPL': ○ Added table 'orders_currentday' - Snapshot of active orders at the start of the current day. ○ Added table 'info_currentday' - Snapshot information. ○ Table 'info' now has fields 'trades_rev', 'trades_lifenum'. ○ Starting from version 8.3, the 'logRev' and 'lifeNum' fields in the 'info' table are deprecated and will be removed in version 8.9. Instead of these fields, you must use the 'trades_rev' and 'trades_lifenum' fields, respectively. • Stream 'FORTS_COMMON_REPL': ○ Table 'common' now contains field 'index_div'. • Stream 'FORTS_POS_REPL':

Date	Changes
	○ The 'xday_open_qty', 'xday_open_buys_qty', 'xday_open_sells_qty' fields have been added to the 'position' and 'position_sa' tables. ○ Added table 'info' - Snapshot information. • Stream 'FORTS_REFDATA_REPL': ○ Added table 'trade_periods' - Trading period parameters. ○ The 'section_id' field has been added to the 'fut_vcb' and 'opt_vcb' tables. ○ Table 'dealer' now contains field 'order_allowed_in_weekend_session'. ○ Table 'discrete_auction' now has fields 'trade_period_id', 'trade_period_type'. ○ Table 'fut_instruments' now contains field 'trade_period_access'. • Stream 'FORTS_VM_REPL': ○ The 'swap_rate' and 'index_div' fields have been added to the 'fut_vm' and 'fut_vm_sa' tables. • Stream 'FORTS_INFO_REPL': ○ Table 'base_contracts_params' now contains field 'signs'. ○ Table 'investor' now contains field 'xstatus'. ○ Table 'dealer' now contains field 'order_allowed_in_weekend_session'. ○ Table 'option_series_params' now contains field 'lot_volume'. • Added a new stream 'FORTS_SESSIONSTATE_REPL' - Status of active session. Tables: ○ session_state - Status of active session ○ sys_events - Table of events • Added a new stream 'FORTS_INSTRUMENTSTATE_REPL' - Instrument statuses for active session. Tables: ○ instrument_state - Instrument statuses for active session ○ sys_events - Table of events • Added a new stream 'FORTS_SECURITYGROUPSTATE_REPL' - Group status of instruments for active session. Tables: ○ security_group_state - Group status of instruments for active session ○ sys_events - Table of events • Added new error codes: 4285, 4227.
17.03.2025	Changes applied: • Added section '2.6.9. Exchange services under Directive 6681-U - Regulation of margin trading for brokerage clients'. • Added section '3.3.10. Displaying information about treasurers' logins in trades/orders'. • Stream 'FORTS_REFDATA_REPL': ○ Table 'fut_sess_contents' now has fields 'trade_period_access', 'buy_deposit_erc', 'sell_deposit_erc', 'buy_deposit_hrc', 'sell_deposit_hrc', 'buy_deposit_lrc', 'sell_deposit_lrc', 'buy_deposit_mrc', 'sell_deposit_mrc'. ○ Starting with version 8.0, the 'is_trade_evening' field from the 'fut_sess_contents' table is deprecated and will be removed in version 8.6. Use the 'trade_period_access' field instead. ○ Table 'opt_sess_contents' now has fields 'trade_period_access', 'base_im_covered_sell_erc', 'base_im_covered_sell_hrc', 'base_im_covered_sell_lrc', 'base_im_covered_sell_mrc', 'base_im_sell_erc', 'base_im_buy_erc', 'base_im_sell_hrc', 'base_im_buy_hrc', 'base_im_sell_lrc', 'base_im_buy_lrc', 'base_im_sell_mrc', 'base_im_buy_mrc'. ○ Starting with version 8.0, the 'is_trade_evening' field from the 'opt_sess_contents' table is deprecated and will be removed in version 8.6. Use the 'trade_period_access' field instead. ○ Table 'sess_option_series' now contains field 'd_sess_assign'.

Date	Changes
	○ Starting from version 8.0, the 'pos_transfer_begin' and 'pos_transfer_end' fields in the 'session' table are deprecated and will be removed in version 8.6. • Stream 'FORTS_INFO_REPL': ○ Table 'investor' now contains field 'client_risk_level'. ○ Table 'base_contracts_params' now has fields 'mr1_lrc', 'mr2_lrc', 'mr3_lrc', 'mr1_mrc', 'mr2_mrc', 'mr3_mrc', 'mr1_hrc', 'mr2_hrc', 'mr3_hrc', 'mr1_erc', 'mr2_erc', 'mr3_erc'. • Added a new stream 'FORTS_RMT_REPL' - Collateral without orders and current operational risk. Tables: ○ rmt_im - Collateral without orders and current operational risk in the context of clients ○ sys_events - Table of events
11.12.2024	Changes applied: • Stream 'FORTS_REFDATA_REPL': ○ Table 'investor' now contains field 'client_risk_level'.
09.10.2024	Changes applied: • Added section '3.3.6.1. Utility 'change_password'. • Added section '3.3.6.2. One-time passwords'. • Stream 'FORTS_REFDATA_REPL': ○ The 'trade_mode_id' and 'group_mask' fields have been added to the 'fut_instruments' and 'fut_sess_contents' tables. ○ Table 'opt_sess_contents' now has fields 'trade_mode_id', 'group_mask', 'is_trade_evening'. ○ A new bit has appeared in the 'fut_sess_contents' table for the 'signs' field: 0x200 - Futures. • Changes applied to command scheme repository: ○ Added a restriction to the 'AddOrder' and 'IcebergAddOrder' commands for the 'comment' field: The field must not contain control characters (0x00-0x1F and 0x7F) and must be converted from cp1251 to utf-8. • Added new error codes: 4180.
08.07.2024	Changes applied: • Stream 'FORTS_TRADE_REPL': ○ The 'match_ref' field has been added to the 'orders_log' and 'multileg_orders_log' tables. • Removed deprecated 'FORTS_MISCINFO_REPL' stream. Use the 'FORTS_RISKINFOBLACK_REPL' and 'FORTS_RISKINFOBACH_REPL' streams instead.
12.04.2024	Changes applied: • Stream 'FORTS_REFDATA_REPL': ○ In the 'fut_sess_contents' table, new values in the 'state' field: ■ 8 - Session for this instrument is running. You can put orders to close position for this instrument. ■ 9 - The opening auction for this instrument is running. You can put orders to close position for this instrument. ○ In the 'opt_sess_contents' table, new values in the 'state' field: ■ 6 - The opening auction for this instrument started. You can put and delete orders for this instrument. ■ 7 - The opening auction for this instrument is completed. ■ 8 - Session for this instrument is running. You can put orders to close position for this instrument. ■ 9 - The opening auction for this instrument is running. You can put orders to close position for this instrument. ○ The 'tas_base_fut_isin_id' field has been added to the 'fut_sess_contents' and 'fut_instruments' tables. • Added new error codes: 140.
18.03.2024	Changes applied:

Date	Changes
	• Added section '2.4.4. Trade Settlement TAS'. • In the section '2.4.5. Flags applied to orders and trades' a description of new flags is added: ○ 'TASSettlement (0x10000)' - Trade Settlement TAS. • Added section '3.4.1.2. Timeouts handling when commands/transactions are sent'. • Stream 'FORTS_TRADE_REPL': ○ The 'compliance_id' field has been added to the 'orders_log' and 'multileg_orders_log' tables. • Stream 'FORTS_USERORDERBOOK_REPL': ○ The 'compliance_id' field has been added to the 'orders' table. • Stream 'FORTS_REFDATA_REPL': ○ A new bit has appeared in the 'fut_sess_contents' table for the 'signs' field: 0x100000 - TAS futures. ○ A new bit has appeared in the 'fut_vcb' table for the 'signs' field: 0x4 - TAS futures. ○ A new bit has appeared in the 'dealer' table for the 'status' field: 0x10000 - NCC. ○ Table 'sys_messages' now contains field 'type_id'. ○ Removed deprecated 'fut_rejected_orders' and 'opt_rejected_orders' tables. Instead of these tables, you must use the 'rejected_orders' table of the 'FORTS_REJECTEDORDERS_REPL' stream. • Starting from version 7.21, the 'FORTS_MISCINFO_REPL' stream is deprecated and will be deleted in version 7.27. Instead, the 'FORTS_RISKINFOBLACK_REPL' and 'FORTS_RISKINFOBACH_REPL' streams should be used. • Stream 'FORTS_INFO_REPL': ○ Table 'futures_params' now contains field 'tas_base_fut_isin_id'. ○ A new bit has appeared in the 'futures_params' table for the 'attribute' field: 0x100000 - TAS futures. • Changes applied to command scheme repository: ○ The 'compliance_id' (c1) field has been added to the 'AddOrder' message. Added new message type 'AddOrder (msgid=474)'. ○ The 'compliance_id' (c1) field has been added to the 'IcebergAddOrder' message. Added new message type 'IcebergAddOrder (msgid=475)'. ○ The 'compliance_id' (c1) field has been added to the 'MoveOrder' message. Added new message type 'AddOrder (msgid=476)'. ○ The 'compliance_id' (c1) field has been added to the 'IcebergMoveOrder' message. Added new message type 'IcebergAddOrder (msgid=477)'. ○ Modified section '5.25. Method SetSmaPreTradeCheck - Enable pre-trade verification mode for SMA login orders'.
01.12.2023	Changes applied: • The name of the document has been changed: instead of 'SPECTRA Plaza-2 gate' it became 'PLAZA II gateway'. • Added section '2.3.2.1. Negotiated mode with matching by a unique code'. • In the section '2.4.4. Flags applied to orders and trades' a description of new flags is added: ○ The obsolete 'InternalHalfTrade (0x80000000)' bit has been renamed to 'NegotiatedMatchByRef (0x80000000)' - Negotiated order or trade matched by reference. • Added a new stream 'FORTS_REJECTEDORDERS_REPL' - Register of orders rejected during the clearing. Tables: ○ rejected_orders - Register of orders rejected during the clearing. The table is used instead of the deprecated 'fut_rejected_orders' and 'opt_rejected_orders' tables of the 'FORTS_REFDATA_REPL' stream. • Stream 'FORTS_COMMON_REPL': ○ Removed deprecated 'type' field from 'sys_events' table. Use the 'event_type' field instead. • Stream 'FORTS_REFDATA_REPL':

Date	Changes
	○ Removed deprecated 'opt_sess_id' field from 'session' table. Use the 'sess_id' field instead. ○ Starting with version 7.18, the 'fut_rejected_orders' and 'opt_rejected_orders' tables are deprecated and will be removed in version 7.21. Instead of these tables, you must use the 'rejected_orders' table of the 'FORTS_REJECTEDORDERS_REPL' stream. • Stream 'RTS_INDEX_REPL': ○ Removed deprecated 'value' field from 'rts_index' table. Use the 'value_highprec' field instead. • Changes applied to command scheme repository: ○ Removed slow versions of the 'ChangeClientMoney' command (command versions with ID in = 4, 60, 63, 67, 409, 425 / Reply ID out = 104), previously deprecated. Use the faster version of the 'ChangeClientMoney' command with ID in = 458 / Reply ID out = 187.
04.09.2023	Changes applied: • Section '3.1.4. Installation for Linux' was split into two sections: ○ '3.1.4.1. Installation from zip archive'. ○ '3.1.4.2. Installation from a deb package or rpm package'. • Added a new stream 'FORTS_USER_REPL' - Users. Tables: ○ user - System users ○ sma_master - SMA login binding to MASTER login ○ sma_pre_trade_check - SMA login pre-trade verification settings. ○ sys_events - Table of events • Stream 'FORTS_PROHIBITION_REPL': ○ Removed value '0x20000000 (Spots)' for 'group_mask' field in 'prohibition' table. • Stream 'FORTS_REFDATA_REPL': ○ The deprecated 'option_series' table has been deleted. Instead of this table, use the 'sess_option_series' table. ○ Table 'sess_option_series' now has fields 'interest_rate_risk_up', 'interest_rate_risk_down', 'r2', 'interest_rate2_risk_up', 'interest_rate2_risk_down'. ○ Removed deprecated 'state' field from 'sess_option_series' table. Use the 'state' field from the 'opt_sess_contents' table instead. ○ Removed deprecated 'status' field from 'investor' table. Use the 'xstatus' field instead. • Stream 'FORTS_CLR_REPL': ○ The 'sbor_nosys' field has been added to the 'fut_pos', 'opt_pos', 'fut_pos_sa' and 'opt_pos_sa' tables. ○ Table 'fut_sess_settl' now contain field 'index_div'. • Stream 'FORTS_VM_REPL': ○ Added the 'sys_events' event table. • Stream 'FORTS_VOLAT_REPL': ○ Added the 'sys_events' event table. • Stream 'FORTS_INFO_REPL': ○ Table 'option_series_params' now has fields 'r2', 'interest_rate2_risk_up', 'interest_rate2_risk_down'. • Stream 'FORTS_TNPENALTY_REPL': ○ Table 'fee_tn' now contain field 'num_orders'. • Removed error codes: 9999.
02.06.2023	Changes applied:

Date	Changes
	• Stream 'FORTS_COMMON_REPL': ○ Table 'sys_events' now contain field 'event_type'. ○ Starting with version 7.12, the 'type' field from the 'sys_events' table is deprecated and will be removed in version 7.18. Use the 'event_type' field instead. • Stream 'FORTS_REFDATA_REPL': ○ Removed deprecated 'd_exec_beg' and 'd_exec_end' fields from 'opt_sess_contents' table. The 'expiration_date' field from the 'sess_option_series' table should be used instead. ○ Removed deprecated 'step_price' field from 'opt_sess_contents' table. Use the 'step_price' field from the 'sess_option_series' table instead. ○ Starting with version 7.12, the 'opt_sess_id' field from the 'session' table is deprecated and will be removed in version 7.18. Use the 'sess_id' field instead. ○ Table 'sess_option_series' now has fields 'step_price_clr', 'step_price_interclr'. • Stream 'FORTS_CLR_REPL': ○ Removed 'vat_ex' and 'vat_cc' fields from 'fut_pos', 'opt_pos', 'fut_pos_sa' and 'opt_pos_sa' tables. • Stream 'RTS_INDEX_REPL': ○ Starting with version 7.12, the 'value' field from the 'rts_index' table is deprecated and will be removed in version 7.18. Use the 'value_highprec' field instead. • Changes applied to command scheme repository: ○ Slow versions of the 'ChangeClientMoney' command (versions of commands with ID in = 4, 60, 63, 67, 409, 425 / Reply ID out = 104) starting from version 7.12 are deprecated and will be removed in version 7.18. Use the faster version of the 'ChangeClientMoney' command with ID in = 458 / Reply ID out = 187. • In version 7.12, the system introduced a limit on the number of simultaneous subscriptions to one Plaza2 (Cgate) stream from one gateway login - no more than 20 (for more details, see the section "3.3.9. Limiting the number of simultaneously open replication streams from one Plaza2 connection").
27.03.2023	Changes applied: • In the section '2.4.4. Trade types, created upon exercising and expiration of futures and options' a description of new flags is added: ○ The obsolete 'eREPOCCStatus (0x2000)' bit has been renamed to 'DueToCrossCancel (0x2000)' - Sign of canceling a passive order in a cross trade. • Starting from version 7.9 in the SPECTRA system, the service for informing participants about the forecast values of risk parameters (ForecastIM) is deprecated with the subsequent removal of the service in version 7.15. • Stream 'FORTS_COMMON_REPL': ○ Table 'common' now contain field 'swap_rate'. ○ Added the 'sys_events' event table. • Stream 'FORTS_PROHIBITION_REPL': ○ Table 'prohibition' now has fields 'section_id', 'base_contract_id'. ○ As of version 7.9, the value '0x20000000 (Spots)' for the 'group_mask' field in the 'prohibition' table is deprecated and will be removed in version 7.15. • Stream 'FORTS_REFDATA_REPL': ○ Removed deprecated 'is_percent' field from 'fut_sess_contents' and 'fut_instruments' tables.Use the 'asset_class' field from the 'fut_vcb' table instead. ○ Table 'investor' now contain field 'xstatus'. The 'xstatus' field differs from the existing 'status' field by the extended type i8 and the transmitted of two additional flags in it: ■ 0x10000000000 - Qualified investor ■ 0x40000000000 - Cancel a passive order in a cross trade

Date	Changes
	○ Starting with version 7.9, the 'status' field from the 'investor' table is deprecated and will be removed in version 7.15. Use the 'xstatus' field instead. ○ Added table 'brokers_base_contracts_params'. ○ Table 'opt_sess_contents' now contain field 'state'. ○ Starting with version 7.9, the 'state' field from the 'option_series' and 'sess_option_series' tables is deprecated and will be removed in version 7.15. Use the 'state' field from the 'opt_sess_contents' table instead. • Stream 'FORTS_INFO_REPL': ○ Removed deprecated 'is_percent' field from 'base_contracts_params' table. The 'asset_class' field should be used instead. ○ Added table 'brokers_base_contracts_params'. • Changes applied to command scheme repository: ○ Added command 'ChangeBFClientBaseContractParametersNextSession ' (msgid=1057) - Changing the parameters of BF clients for the basic contract.
09.11.2022	Changes applied: • New order type Book-or-Cancel (BOC) was added to section "2.3.1. Orders — general information". • Section '2.4.4. Trade types, created upon exercising and expiration of futures and options' was split into two sections: ○ '2.4.4. Flags applied to orders and trades'. ○ '2.4.5. Trade types, created upon exercising and expiration of futures and options'. • In the section '2.4.4. Flags applied to orders and trades' a description of new flags is added: ○ DuringDiscreteAuction (0x4000000000000000) - Sign of an order or trade in the opening auction. ○ BOC (0x1000000000000000) - Book-or-Cancel order. • Added section '2.5.7. Opening auction'. • Stream 'FORTS_TRADE_REPL': ○ The 'xstatus2' field has been added to the 'orders_log' and 'multileg_orders_log' tables. ○ The 'xstatus2_buy' and 'xstatus2_sell' fields have been added to the 'user_deal' and 'user_multileg_deal' tables. • Stream 'FORTS_ORDLOG_REPL': ○ The 'xstatus2' field has been added to the 'orders_log' and 'multileg_orders_log' tables. • Stream 'FORTS_DEALS_REPL': ○ The 'xstatus2_buy' and 'xstatus2_sell' fields have been added to the 'deal' and 'multileg_deal' tables. • Stream 'FORTS_USERORDERBOOK_REPL': ○ The 'xstatus2' field has been added to the 'orders' table. • Stream 'FORTS_ORDBOOK_REPL': ○ The 'xstatus2' field has been added to the 'orders' table. • Stream 'FORTS_COMMON_REPL': ○ Table 'common' now contain field 'opening_auction_price'. • Stream 'FORTS_REFDATA_REPL': ○ Added 'sess_option_series', 'discrete_auction' and 'discrete_auction_base_contract' tables. ○ The fields 'europe', 'min_step', 'lot_volume' were removed from the 'opt_sess_contents' table. ○ Table 'user' now contain field 'user_level'.

Date	Changes
	○ Starting with version 7.6, the 'option_series' table is deprecated and will be removed in future versions. Instead of this table, use the 'sess_option_series' table. • Stream 'FORTS_CLR_REPL': ○ The 'pos_exec' and 'charge_exec' fields were added to the 'fut_pos', 'opt_pos', 'fut_pos_sa' and 'opt_pos_sa' tables. ○ Starting with version 7.6, the 'vat_ex' and 'vat_cc' fields in the 'fut_pos', 'opt_pos', 'fut_pos_sa' and 'opt_pos_sa' tables are deprecated and will be removed in version 7.12. • Stream 'FORTS_INFO_REPL': ○ The 'subrisk_step' field was removed from the 'base_contracts_params' table. ○ The 'subrisk' field was removed from the 'futures_params' table. • Changes applied to command scheme repository: ○ The 'type' (i4) field has been added to the 'IcebergAddOrder' message. Added new message type 'IcebergAddOrder (msgid=472)'. • Added new error codes: 82, 85, 90, 95, 100, 105, 110, 115, 120, 125, 130, 135, 3002, 4300-4305 • Changed texts of error codes: 62
15.09.2022	Changes applied: • Added section '2.3.4. Cross-trades'. • Added section '2.4.3. Termination of obligations under daily futures contracts with auto-prolongation (‘perpetual’ futures)'. • In the section '2.4.4. Trade types, created upon exercising and expiration of futures and options' a description of new flags is added: ○ ePerpetualFuturesExecutionVoluntary (0x10000000000000) - The technical trade as a result of exiting a perpetual futures (based on the submitted requests). ○ ePerpetualFuturesExecutionForced (0x400000000000000) - The technical trade as a result of forced exiting a perpetual futures (realization of unsatisfied demand). ○ ePerpetualFuturesExecution (0x800000000000000) - The technical trade with linked instrument as a result of exiting a perpetual futures. • Stream 'FORTS_FEE_REPL': ○ Table 'adjusted_fee' now contains fields 'adjusted_fee_trade_buy', 'adjusted_fee_clearing_buy', 'adjusted_fee_trade_sell', 'adjusted_fee_clearing_sell'. • Stream 'FORTS_PROHIBITION_REPL': ○ Added the 'sys_events' event table. • Stream 'FORTS_REFDATA_REPL': ○ The deprecated 'prohibition' table has been deleted. Instead of this table, use the 'prohibition' table of the 'FORTS_PROHIBITION_REPL' stream. ○ Added table 'fut_exec_orders'. ○ Starting with version 7.3, the 'is_percent' field from the 'fut_sess_contents' and 'fut_instruments' tables is deprecated and will be removed in version 7.9. Use the 'asset_class' field from the 'fut_vcb' table instead. ○ Starting from version 7.3, the 'd_exec_beg' and 'd_exec_end' fields in the 'opt_sess_contents' table are deprecated and will be removed in version 7.9. The 'expiration_date' field from the 'option_series' table should be used instead. • Stream 'FORTS_INFO_REPL': ○ The 'signs' field was removed from the 'currency_params' table. ○ Starting with version 7.3, the 'is_percent' field in the 'base_contracts_params' table is deprecated and will be removed in version 7.9. The 'asset_class' field should be used instead.

Date	Changes
	• Changes applied to command scheme repository: ○ The 'client_priority' (i4) field has been added to the 'FutChangeClientProhibit' and 'OptChangeClientProhibit' messages. New types of 'FutChangeClientProhibit' (msgid=469) and 'OptChangeClientProhibit' (msgid=468) messages have been added. ○ Added command 'FuturesExecutionRequest ' (msgid=470) - Exercise requests for daily futures contracts with auto-prolongation. ○ In the 'OptChangeExpiration' message, the 'order_id' field type has been changed to 'i8'. Added new message type 'OptChangeExpiration (msgid=471)'. • Added new error codes: 4283, 5052-5055, 5061-5065, 5069, 5071-5073. • Changed texts of error codes: 4050-4055, 4060-4067, 4069, 4070. • Removed error codes: 300-307, 4068.
17.05.2022	Changes applied: • Added section '2.11. Equity Options'. • Changed section '2.2.3. Options'. • Changed section '2.4.2. Option exercise'. • In the section '2.4.3. Trade types, created upon exercising and expiration of futures and options' a description of new flags is added: ○ eSyntheticPassive (0x200000000000000) - 'Sign of a passive synthetic order'. • Changed section '2.5. Trading and clearing schedule'. • Changed section '2.6.3. Limitations on trading operations and opening positions for clients'. • Stream 'FORTS_TRADE_REPL': ○ Removed 'id_ord', 'xamount', 'xamount_rest', 'action' fields from 'orders_log' and 'multileg_orders_log' tables. ○ Removed 'id_ord_buy' and 'id_ord_sell' fields from 'user_deal' and 'user_multileg_deal' tables. • Stream 'FORTS_USERORDERBOOK_REPL': ○ The fields 'id_ord', 'xamount', 'xamount_rest', 'action', 'init_moment', 'xinit_amount' were removed from the 'orders' table. • Stream 'FORTS_POS_REPL': ○ The 'last_quantity' field was added to the 'position' and 'position_sa' tables. • Stream 'FORTS_PART_REPL': ○ The 'premium_intercl' and 'net_option_value' fields were added to the 'part' and 'part_sa' tables. • Stream 'FORTS_PROHIBITION_REPL': ○ The 'prohib_id' field was removed from the 'prohibition' table. • Stream 'FORTS_REFDATA_REPL': ○ Table 'option_series' now contains fields 'margin_style', 'settlement_type', 'exercise_style', 'min_step', 'step_price', 'lot_coefficient', 'r', 'interest_rate_risk_up', 'interest_rate_risk_down', 'fixed_spot_discount', 'projected_spot_discount', 'step_price_curr', 'underlying_price', 'lot_volume','state'. ○ The 'base_isin_id' field was removed from the 'opt_sess_contents' table. ○ Starting with version 7.0, the 'europe' field in the 'opt_sess_contents' table is deprecated and will be removed in version 7.6. Use the 'exercise_style' field from the 'option_series' table instead. ○ Starting with version 7.0, the 'min_step' field in the 'opt_sess_contents' table is deprecated and will be removed in version 7.6. Use the 'min_step' field from the 'option_series' table instead. ○ Starting with version 7.0, the 'lot_volume' field in the 'opt_sess_contents' table is deprecated and will be removed in version 7.6. Use the 'lot_volume' field from the 'option_series' table instead.

Date	Changes
	○ Starting with version 7.0, the 'step_price' field in the 'opt_sess_contents' table is deprecated and will be removed in version 7.6. Use the 'step_price' field from the 'option_series' table instead. ○ Table 'fut_vcb' now has fields 'asset_class', 'board_md'. ○ In the 'fut_vcb' table, the value for the 'asset_class' field has been removed: '7' - Precious metal. ○ Table 'opt_intercl_info' now has fields 'premium', 'premium_in_settl_currency'. ○ Table 'opt_vcb' now contains field 'settlement_currency'. ○ The 'prohib_id' field was removed from the 'prohibition' table. ○ Removed deprecated table 'usd_online'. ○ The fields 'A', 'B', 'C', 'D', 'E', 'S' were removed from the 'option_series' table. • Stream 'FORTS_CLR_REPL': ○ The 'premium' and 'premium_in_settl_currency' fields were added to the 'opt_pos' and 'opt_pos_sa' tables. ○ Added table 'option_series_settl'. • Stream 'FORTS_VM_REPL': ○ The 'premium' and 'premium_in_settl_currency' fields were added to the 'opt_vm' and 'opt_vm_sa' tables. • Stream 'FORTS_INFO_REPL': ○ Starting with version 7.0, the 'subrisk_step' field in the 'base_contracts_params' table is deprecated and will be removed in version 7.6. Use the 'strike_step' field from the 'option_series_params' table instead. ○ Table 'base_contracts_params' now has fields 'asset_class', 'cf_risk'. ○ In the 'base_contracts_params' table, the value for the 'asset_class' field has been removed: '7' - Precious metal. ○ Starting with version 7.0, the 'subrisk' field in the 'futures_params' table is deprecated and will be removed in version 7.6. Use the 'sub_risk' field from the 'option_series_params' table instead. ○ Table 'option_series_params' now contains fields 'sub_risk', 'spread_aspect', 'enforce_half_netting', 'min_step', 'step_price', 'lot_coefficient', 'r', 'interest_rate_risk_up', 'interest_rate_risk_down', 'fixed_spot_discount', 'projected_spot_discount'. ○ The fields 'a', 'b', 'c', 'd', 'e', 's' were removed from the 'option_series_params' table. • Changed texts of error codes: 4067.
05.04.2022	Changes applied: • Added section '2.2.2.1. Daily futures contract with automatic prolongation'. • The section '3.3.4. Latency monitoring by the client side', which is outdated, has been removed. • Stream 'FORTS_REFDATA_REPL': ○ A new bit has appeared in the 'fut_sess_contents' table for the 'signs' field: 0x4000 - Daily futures contract with automatic prolongation (CFD - Contract for difference). • Stream 'FORTS_CLR_REPL': ○ Table 'fut_sess_settl' now contain field 'swap_rate'.
20.10.2021	Changes applied: • Updating the utility for changing the password (change_password.exe): ○ The 'app_name' parameter (application name) has been added to the command string. ○ The 'local_pass' parameter (password for the local connection to the router) has been added to the command string. ○ The 'key' parameter has been removed from the valid command string parameters. • Added a new stream 'FORTS_PROHIBITION_REPL' - Prohibitions. The 'prohibition' table is broadcast in a separate stream.

Date	Changes
	• Stream 'FORTS_REFDATA_REPL': ○ Starting with version 6.15, the 'prohibition' table is deprecated and will be removed in version 7.3. Instead of this table, use the 'prohibition' table of the 'FORTS_PROHIBITION_REPL' stream. • Stream 'FORTS_INFO_REPL': ○ Starting with version 6.15, the 'signs' field in the 'currency_params' table is deprecated and will be removed in version 7.3. • Added new error codes: 81, 4280-4282. • Changed texts of error codes: 4160. • Removed error codes: 4168.
23.07.2021	Changes applied: • Added section '2.10. Settlement trades'. • Stream 'FORTS_PART_REPL': ○ The 'balance_money' field was removed from the 'part' table. • Stream 'FORTS_REFDATA_REPL': ○ The 'enforce_ims_half_netting' field was added to the 'fut_sess_contents' and 'fut_instruments' tables. • Stream 'FORTS_INFO_REPL': ○ Table 'futures_params' now contains field 'enforce_ims_half_netting'. ○ Table 'option_series_params' now contain fields 'margin_style', 'settlement_type', 'exercise_style'. • Stream FORTS_FEERATE_REPL: ○ Tables 'futures_rate', 'option_rate' now contain 'exp_clearing_fee' field. In version 6.12, the field will always contain "0.0". In version 6.15, this field will be filled with rate values. • Removed error codes: 4120, 4121. • Added new error codes: 80.
14.05.2021	Changes applied: • In the section '2.4.3. Trade types, created upon exercising and expiration of futures and options' a description of new flags is added: ○ eDontFineRF (0x80000000000000) - 'No penalty for settlement transactions'. • Removed 'FORTS_FUTTRADE_REPL' and 'FORTS_OPTTRADE_REPL' streams. Use the 'FORTS_TRADE_REPL' stream instead. • Removed 'FORTS_FUTORDERBOOK_REPL' and 'FORTS_OPTORDERBOOK_REPL' streams. Use the 'FORTS_USERORDERBOOK_REPL' stream instead. • Removed 'FORTS_FUTCOMMON_REPL' and 'FORTS_OPTCOMMON_REPL' streams. Use the 'FORTS_COMMON_REPL' stream instead. • Removed 'FORTS_FUTINFO_REPL' and 'FORTS_OPTINFO_REPL' streams. Use the 'FORTS_REFDATA_REPL' stream instead. • Removed the streams 'FORTS_FUTAGGR5_REPL', 'FORTS_FUTAGGR20_REPL', 'FORTS_FUTAGGR50_REPL', 'FORTS_OPTAGGR5_REPL', 'FORTS_OPTAGGR20_REPL' and 'FORTS_OPTAGGR50_REPL'. Use the 'FORTS_AGGR5_REPL', 'FORTS_AGGR20_REPL', and 'FORTS_AGGR50_REPL' streams instead. • Stream 'FORTS_TRADE_REPL': ○ Removed 'local_stamp' field from 'orders_log' and 'multileg_orders'_log tables. ○ The 'reason' field has been added to the 'orders_log' and 'multileg_orders_log' tables. ○ The 'reason_buy' and 'reason_sell' fields have been added to the 'user_deal' and 'user_multileg_deal' tables. • Stream 'FORTS_FEE_REPL':

Date	Changes
	○ Added table 'penalty'. • Stream 'FORTS_USERORDERBOOK_REPL': ○ Table 'orders' now contain field 'reason'. • Stream 'FORTS_PART_REPL': ○ Table 'part' now contain field 'penalty'. ○ Table 'part_sa' now contain field 'blocked_tax'. • Stream 'FORTS_REFDATA_REPL': ○ Removed the 'd_start' field from the 'fut_sess_contents', 'fut_instruments' and 'opt_sess_contents' tables. ○ Table 'prohibition' now contain field 'xprohibition_id'. ○ Starting with version 6.9, the 'prohib_id' field in the 'prohibition' table is deprecated and will be removed in version 6.15. ○ Starting with version 6.9, the 'usd_online' table is deprecated and will be removed in version 6.15. Instead of this table, use the 'curr_online' table of the 'MOEX_RATES_REPL' stream. • Stream 'FORTS_CLR_REPL': ○ Table 'money_clearing_sa' now contain field 'blocked_tax'. • Changes applied to command scheme repository: ○ The 'local_stamp' field has been removed from the 'AddOrder', 'MoveOrder', 'DelOrder', 'DelUserOrders', 'IcebergAddOrder','IcebergMoveOrder', and 'IcebergDelOrder' messages. Added new message types: 'AddOrder' (msgid = 465), 'MoveOrder' (msgid = 460), 'DelOrder' (msgid = 461), 'DelUserOrders' (msgid = 466), 'IcebergAddOrder' (msgid = 462), 'IcebergMoveOrder' (msgid = 463) and 'IcebergDelOrder' (msgid = 464). ○ The deprecated message 'FutTransferRisk.' was removed. • Added new error codes: 3001. • Changed texts of error codes: 4017.
25.02.2021	Changes applied: • Stream 'FORTS_PART_REPL': ○ Starting from version 6.8, the 'balance_money' field in the 'part' table is deprecated and will be removed in version 6.12.
12.01.2021	Changes applied: • Stream 'FORTS_REFDATA_REPL': ○ Table 'dealer' now contain field 'order_allowed_in_morning_session'. • Stream 'FORTS_INFO_REPL': ○ Table 'dealer' now contain field 'order_allowed_in_morning_session'. • Changes applied to command scheme repository: ○ Since version 6.7, the 'FutTransferRisk' procedure is deprecated and will be removed in future versions. • Added new error codes: 4226.
19.10.2020	Changes applied: • Changed section '2.5. Trading and clearing schedule'. • Changed section '2.4.3. Trade types, created upon exercising and expiration of futures and options'. • Stream 'FORTS_TRADE_REPL': ○ In the 'orders_log' and 'multileg_orders_log' tables, the 'local_stamp' field is deprecated and will be removed in version 6.9. • Stream 'FORTS_DEALS_REPL':

Date	Changes
	○ Table 'deal' now has fields 'xstatus_buy', 'xstatus_sell'. ○ Table 'multileg_deal' now has fields 'xstatus_buy', 'xstatus_sell'. • Stream 'FORTS_USERORDERBOOK_REPL': ○ Table 'info' now contain field 'publication_state'. • Stream 'FORTS_ORDBOOK_REPL': ○ Table 'info' now contain field 'publication_state'. • Stream 'FORTS_REFDATA_REPL': ○ Table 'user' now contain field 'password_expiration_date'. • Changes applied to command scheme repository: ○ Updated command for setting client limits 'ChangeClientMoney' (msgid=458). ○ Removed the following deprecated commands: 'FutAddOrder', 'OptAddOrder', 'FutAddMultilegOrder', 'FutDelOrder', 'OptDelOrder', 'FutMoveOrder', 'OptMoveOrder', 'FutDelUserOrders', 'OptDelUserOrders', 'FutChangeClientMoney', 'FutChangeBFMoney', 'FutExchangeBFMoney', 'FutTransferClientPosition', 'OptTransferClientPosition', 'FutChangeBFLimit'. ○ The command 'OptChangeRiskParameters' renamed to 'OptChangeRiskParametersNextSession'. ○ In commands, the 'local_stamp' field is deprecated and will be removed in version 6.9. • Added new error codes: 300-307, 4175. • Changed texts of error codes: 4006-4011, 4017.
17.08.2020	Changes applied: • Added section '2.3.5. Iceberg orders'. • In the section '2.4.3. Trade types, created upon exercising and expiration of futures and options' a description of new flags is added: ○ eIceberg (0x800000000000) - sign of an iceberg order, trade on an iceberg order ○ eSynthetic (0x200000000000) - sign of synthetic order ○ eOperatorInputSA (0x1000000000000) - blocking by Settlement Account • Changed section '2.6.9. Negative prices support in SPECTRA'. • Added section '2.9. Synthetic matching '. • Added new stream 'FORTS_TRADE_REPL'. Combines 'FORTS_FUTTRADE_REPL' and 'FORTS_OPTTRADE_REPL' streams. • The 'FORTS_FUTTRADE_REPL' and 'FORTS_OPTTRADE_REPL' streams in SPECTRA 6.5 are deprecated, the changes should be found in the 'FORTS_TRADE_REPL' stream description. • Stream 'FORTS_TRADE_REPL': ○ Table 'orders_log' now contain fields 'public_order_id', 'public_amount', 'public_amount_rest', 'public_action', 'private_order_id', 'private_amount', 'private_amount_rest', 'variance_amount', 'private_action', 'disclose_const_amount'. ○ Table 'multileg_orders_log' now contain fields 'public_order_id', 'public_amount', 'public_amount_rest', 'public_action', 'private_order_id', 'private_amount', 'private_amount_rest', 'variance_amount', 'private_action', 'disclose_const_amount'. ○ Table 'user_deal' now contain fields 'public_order_id_buy', 'public_order_id_sell', 'private_order_id_buy', 'private_order_id_sell'. ○ Table 'user_multileg_deal' now contain fields 'public_order_id_buy', 'public_order_id_sell', 'private_order_id_buy', 'private_order_id_sell'. • Added new stream 'FORTS_USERORDERBOOK_REPL'. Combines 'FORTS_FUTORDERBOOK_REPL' and 'FORTS_OPTORDERBOOK_REPL' streams.

Date	Changes
	• The 'FORTS_FUTORDERBOOK_REPL' and 'FORTS_OPTORDERBOOK_REPL' streams in SPECTRA 6.5 are deprecated, the changes should be found in the 'FORTS_USERORDERBOOK_REPL' stream description. • Stream 'FORTS_USERORDERBOOK_REPL': ○ Table 'orders' now contain fields 'public_order_id', 'public_amount', 'public_amount_rest', 'public_action', 'private_order_id', 'private_amount', 'private_amount_rest', 'variance_amount', 'private_action', 'disclose_const_amount', 'public_init_moment', 'public_init_amount', 'private_init_moment', 'private_init_amount'. • In the 'orders_log' and 'multileg_orders_log' tables of the 'FORTS_ORDLOG_REPL' stream: ○ The 'id_ord' field is renamed to 'public_order_id'. ○ The 'xamount' field is renamed to 'public_amount'. ○ The 'xamount_rest' field is renamed to 'public_amount_rest'. ○ The 'action' field is renamed to 'public_action'. • In the 'deal' and 'multileg_deal' tables of the 'FORTS_DEALS_REPL' stream: ○ The 'id_ord_buy' field is renamed to 'public_order_id_buy'. ○ The 'id_ord_sell' field is renamed to 'public_order_id_sell'. • In the 'orders' table of the 'FORTS_ORDBOOK_REPL' stream: ○ The 'id_ord' field is renamed to 'public_order_id'. ○ The 'xamount' field is renamed to 'public_amount'. ○ The 'xamount_rest' field is renamed to 'public_amount_rest'. ○ The 'action' field is renamed to 'public_action'. ○ The 'init_moment' field is renamed to 'public_init_moment'. ○ The 'xinit_amount' field is renamed to 'public_init_amount'. • Added new stream 'FORTS_COMMON_REPL'. Combines 'FORTS_FUTCOMMON_REPL' and 'FORTS_OPTCOMMON_REPL' streams. • The 'FORTS_FUTCOMMON_REPL' and 'FORTS_OPTCOMMON_REPL' streams in SPECTRA 6.5 are deprecated, the changes should be found in the 'FORTS_COMMON_REPL' stream description. • Stream 'FORTS_COMMON_REPL': ○ Description of the 'best_buy', 'xamount_buy', 'orders_buy_qty', 'xorders_buy_amount', 'best_sell', 'xamount_sell', 'orders_sell_qty', 'xorders_sell_amount' fields has been changed in the 'common' table. ○ Table 'common' now contain fields 'best_buy_native', 'xamount_buy_native', 'xorders_buy_amount_native', 'best_sell_native', 'xamount_sell_native', 'xorders_sell_amount_native'. ○ The 'old_kotir' and 'cur_kotir' fields have been removed from the 'common' table. • Aggregated order-book stream: ○ Added new streams 'FORTS_AGGR5_REPL', 'FORTS_AGGR20_REPL', 'FORTS_AGGR50_REPL'. Combining relevant futures and options streams. ○ Table 'orders_aggr' now contain field 'synth_volume'. ○ Description of the 'volume' field has been changed in the 'orders_aggr' table. • Stream 'FORTS_CLR_REPL': ○ The 'share' field have been removed from the 'money_clearing' table. ○ The 'account' field have been removed from the 'fut_pos' table. ○ The 'account' field have been removed from the 'opt_pos' table. • Added new stream 'FORTS_REFDATA_REPL'. Combines 'FORTS_FUTINFO_REPL' and 'FORTS_OPTINFO_REPL' streams.

Date	Changes
	• The 'FORTS_FUTINFO_REPL' and 'FORTS_OPTINFO_REPL' streams in SPECTRA 6.5 are deprecated, the changes should be found in the 'FORTS_REFDATA_REPL' stream description. • Stream 'FORTS_REFDATA_REPL': ○ The 'code_vcb', 'old_kotir', 'd_pg', 'last_cl_quote' fields have been removed from the 'fut_sess_contents' table. ○ The 'code_vcb', 'is_foreign' fields have been removed from the 'fut_vcb' table. ○ The 'code_vcb', 'old_kotir', 'd_pg', 'd_exp', 'exec_name', 'last_cl_quote' fields have been removed from the 'fut_instruments' table. ○ The 'go_ratio' field have been removed from the 'dealer' table. ○ The 'code_vcb' field have been removed from the 'prohibition' table. ○ Table 'fut_margin_type' now contain field 'operator_input'. ○ 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. ○ The 'code_vcb' field have been removed from the 'opt_vcb' table. ○ Table 'option_series' now contain field 'm_bach'. • Stream 'FORTS_INFO_REPL': ○ The 'code_vcb' field have been removed from the 'base_contracts_params' table. ○ The 'code_vcb', 'settl_price', 'settl_price_real' fields have been removed from the 'futures_params' table. ○ The 'go_ratio' field have been removed from the 'investor' table. ○ The 'go_ratio' field have been removed from the 'dealer' table. ○ Table 'option_series_params' now contain field 'm_bach'. • Stream 'FORTS_PART_REPL': ○ The 'coeff_go', 'no_fut_discount', 'num_clr_2delivery' fields have been removed from the 'part' table. • Stream 'FORTS_RISKINFOBACH_REPL': ○ Table 'volat_coeff' now contain field 'm'. • Changes applied to command scheme repository: ○ Added command 'AddOrder'. Combines 'FutAddOrder', 'OptAddOrder' and 'FutAddMultilegOrder'. ○ Added command 'DelUserOrders'. Combines 'FutDelUserOrders' and 'OptDelUserOrders'. ○ Added commands 'IcebergAddOrder', 'IcebergDelOrder', 'IcebergMoveOrder'. Commands to manage iceberg orders. ○ The command 'ChangeBFParameters' renamed to 'ChangeBFParametersNextSession'. ○ The command 'ChangeBFClientDefaultParameters' renamed to 'ChangeBFClientDefaultParametersNextSession'. • Added new error codes: 4174, 4258, 4259, 4260, 4261, 4262, 4264, 4266, 4268.
19.06.2020	Changes applied: • Added section '2.6.9. Negative prices support in SPECTRA'. • Added new stream 'FORTS_RISKINFOBLACK_REPL - Risk parameters for Black-Scholes model'. • Added new stream 'FORTS_RISKINFOBACH_REPL - Risk parameters for Bachelier model'. In release 6.4.20, '0' is translated in the parameters. • Stream 'FORTS_FUTCOMMON_REPL': ○ Table 'common' now contain field 'price_assigned_by_admin'. • Stream 'FORTS_FUTINFO_REPL': ○ Table 'fut_vcb' now contain fields 'negative_prices', 'option_model'.

Date	Changes
	• Stream 'FORTS_OPTINFO_REPL': ○ Table 'opt_vcb' now contain fields 'negative_prices', 'option_model'. ○ Table 'option_series' now contain fields 'a-s_black', 'a-s_bach'. • Stream 'FORTS_INFO_REPL': ○ Table 'base_contracts_params' now contain field 'option_model'. The 'has_options' and 'spot_price' fields are deprecated and will be removed in version 6.6. ○ Table 'futures_params' now contain fields 'attribute', 'mr_addon_up', 'mr_addon_down'. ○ Table 'option_series_params' now contain fields 'a-s_black', 'a-s_bach'. • Stream 'FORTS_VOLAT_REPL': ○ Table 'volat' now contain field 'option_model'.
15.01.2020	Changes applied: • Added section '2.6.8. Blocking the brokerage part of the client fee'. • Added new stream 'FORTS_BROKER_FEE_REPL - Brokerage fees'. • Added new stream 'FORTS_BROKER_FEE_PARAMS_REPL - Parameters for calculating the brokerage fee' . • Stream 'FORTS_PART_REPL': ○ Table 'part' now contain field 'broker_fee'. • Stream 'FORTS_OPTCOMMON_REPL': ○ Table 'common' now contain field 'total_premium_volume'. • Changes applied to command scheme repository: ○ Removed 'FutChangeBFParameters' command. ○ Added command 'SetBrokerFeeParamNextSession - Setting parameters for calculating the brokerage fee'. • Added new error codes: 3000.
10.12.2019	Changes applied: • Modified section '3.1.3. Installation for Windows'.
12.09.2019	Changes applied: • Added section '3.3.10. Stream types'. Type added to stream descriptions.
31.08.2019	Changes applied: • Added section '3.3.9. Partitioning of the matching'. • The description of two new flags is added to section '2.4.3. Trade types, created upon exercising and expiration of futures and options': eActiveSide (0x20000000000), ePassiveSide (0x40000000000). They are put down in orders and trades. • Stream 'FORTS_FUTINFO_REPL': ○ Added table 'instr2matching_map'. • Stream 'RTS_INDEX_REPL': ○ Table 'rts_index': deleted fields 'prev_close_value', 'open_value', 'max_value', 'min_value'. • Changes applied to command scheme repository: ○ Added commands 'DelOrder', 'MoveOrder', 'ChangeClientParameters'. ○ Field type changed from i4 to i1 for the fields 'calendar_spread_margin_type' and 'ics_margin_type' in the 'ChangeBFParameters' and 'ChangeClientParametersNextSession' commands. • Added new error codes: 78, 79, 4269.
20.06.2019	Changes applied: • Stream 'FORTS_FUTTRADE_REPL':

Date	Changes
	○ Table 'orders_log': deleted fields 'hedge', 'trust'. ○ Table 'multileg_orders_log': deleted fields 'hedge', 'trust'. ○ Table 'user_deal': deleted fields 'trust_buy', 'trust_sell', 'hedge_buy', 'hedge_sell'. ○ Table 'user_multileg_deal': deleted fields 'isin_id_repo', 'buyback_amount', 'trust_buy', 'trust_sell', 'hedge_buy', 'hedge_sell'. • Stream 'FORTS_OPTTRADE_REPL': ○ Table 'orders_log': deleted fields 'hedge', 'trust'. ○ Table 'user_deal': deleted fields 'trust_buy', 'trust_sell', 'hedge_buy', 'hedge_sell'. • Field 'buyback_amount ' deleted from table 'multileg_deal' of stream 'FORTS_DEALS_REPL'. • Field 'id_repo' deleted from table 'adjusted_fee' of stream 'FORTS_FEE_REPL'. • Stream 'FORTS_FUTORDERBOOK_REPL' / 'FORTS_OPTORDERBOOK_REPL': ○ Table 'orders': deleted fields 'hedge', 'trust'. • Stream 'FORTS_FUTCOMMON_REPL': ○ Table 'common': deleted field 'cur_kotir_real'. Added fields 'settlement_price_open', 'market_price'. • Stream 'FORTS_OPTCOMMON_REPL': ○ Table 'common' now contain field 'settlement_price_open'. • Stream 'FORTS_FUTINFO_REPL': ○ Added table 'clearing_members'. ○ Table 'fut_sess_contents': deleted fields 'd_exp', 'price_dir'. Added fields 'base_contract_code', 'settlement_price_open', 'settlement_price', 'last_trade_date'. ○ Table 'fut_vcb' now contain fields 'base_contract_code', 'signs'. ○ Table 'fut_instruments': deleted field 'price_dir'. Added fields 'base_contract_code', 'settlement_price_open', 'settlement_price', 'last_trade_date', 'd_exp_start', 'series_type'. ○ Table 'prohibition' now contain field 'base_contract_code'. ○ Table 'dealer' now contain fields 'coeff_im', 'short_option_minimum_charge_ratio', 'ics_margin_type'. ○ Table 'investor' now contain fields 'is_blank', 'short_option_minimum_charge_ratio', 'ics_margin_type', 'coeff_im', 'no_fut_discount', 'num_clr_2delivery', 'exp_weight'. • Stream 'FORTS_OPTINFO_REPL': ○ Table 'opt_sess_contents' now contain fields 'base_contract_code', 'settlement_price_open', 'base_im_covered_sell', 'base_im_sell', 'last_trade_date', 'base_im_buy', 'settlement_price'. ○ Table 'opt_vcb' now contain field 'base_contract_code'. ○ Table 'option_series' now contain field 'signs'. • Stream 'FORTS_CLR_REPL': ○ Field 'asset_type' added into tables 'money_clearing', 'money_clearing_sa'. ○ Table 'money_clearing' now contain field 'asset_type'. ○ Table 'fut_pos' now contain field 'account_type'. ○ Table 'opt_pos' now contain field 'account_type'. ○ Table 'pledge_details': deleted field 'com_ensure'. • Stream 'FORTS_VM_REPL': ○ Feld 'vm_real' deleted from tables 'fut_vm', 'opt_vm', 'fut_vm_sa', 'opt_vm_sa' • Stream 'FORTS_INFO_REPL':

Date	Changes
	○ Added table 'multileg_dictionary'. ○ Table 'base_contracts_params': deleted field 'is_usd'. Added fields 'base_contract_code', 'window_size'. ○ Table 'futures_params' now contain fields 'base_contract_code', 'settlement_price', 'risk_range_center'. ○ Deleted table 'virtual_futures_params'. ○ Table 'investor' now contain fields 'is_blank', 'coeff_im', 'short_option_minimum_charge_ratio', 'ics_margin_type'. ○ Table 'investor': field 'n_clr_2delivery' renamed to 'num_clr_2delivery'. ○ Table 'dealer' now contain fields 'coeff_im', 'short_option_minimum_charge_ratio', 'ics_margin_type'. • Changes applied to command scheme repository: ○ The fields 'du', 'hedge' removed in commands 'FutAddOrder', 'OptAddOrder'. ○ The field 'limit_pledge' removed in command 'FutChangeBFMoney'. ○ The field 'amount_pledge' removed in command 'FutExchangeBFMoney'. ○ The fields ‘price’, ‘hedge’, ‘trust’, ‘trade_mode’ removed in command ‘FutAddMultiLegOrder'. The field 'rate_price' renamed to 'swap_price'. ○ The field 'code_vcb' renamed to 'base_contract_code' in commands 'FutDelUserOrders', 'OptDelUserOrders', 'FutChangeClientProhibit', 'OptChangeClientProhibit'. ○ Added commands 'ChangeClientMoney', 'ChangeBFMoney', 'ExchangeBFMoney', 'ChangeBFLimit', 'ChangeBFParameters', 'ChangeClientParametersNextSession'. ○ The command 'FutChangeBFClientDefaultParameters' renamed to 'ChangeBFClientDefaultParameters'. ○ Command 'OptChangeRiskParameters' now have field 'short_option_minimum_charge_ratio' - Individual coefficient of SOMC scenario weight. • Added new error codes: 77, 4225.
14.01.2019	Changes applied: • Field 'base_contract_id' added into table 'fut_vcb' of stream 'FORTS_FUTINFO_REPL'. • Field 'base_contract_id' added into table 'opt_vcb' of stream 'FORTS_OPTINFO_REPL'.
05.12.2018	Changes applied: • Added sections '2.7. Separate entities of Clearing Member and Trading Member', '3.3.8. Changing user password for the Trading System'. • Commands 'FutAddOrder', 'FutAddMultiLegOrder', 'FutDelOrder', 'FutMoveOrder', 'OptAddOrder', 'OptDelOrder', 'OptMoveOrder' now have field ncc_request - flag 'Request to NCC'. The commands now have new IDs. • Added commands 'DelOrdersByBFLimit', 'ChangePassword'. • Command 'OptRecalcCS': field 'isin_id' replaced with 'option_series_id', the command now has a new ID. • Added new error codes: 682, 4168, 4169, 4170, 4171, 4172, 4173, 4221, 4222, 4223, 4224, 4230. • Stream 'FORTS_OPTINFO_REPL': ○ added table 'option_series'. ○ table 'opt_sess_contents': deleted fields 'is_limited', 'limit_up', 'limit_down', 'exch_pay', added field 'option_series_id'. • Stream 'FORTS_FUTINFO_REPL': ○ deleted tables 'diler', 'investr'. ○ table 'dealer' now has fields 'firm_id', 'tm_name'. ○ table 'fut_sess_contents': deleted fields 'is_limited', 'exch_pay'. Added fields 'd_exp_start', 'd_exp_end'. ○ table 'fut_instruments': deleted fields 'is_limited', 'volat_min', 'volat_max', 'is_limit_opt', 'limit_up_opt', 'limit_down_opt', 'adm_lim', 'adm_lim_offmoney', 'apply_adm_limit'.

Date	Changes
	• Stream 'FORTS_MM_REPL': ○ table 'cs_mm_rule': field 'isin_id' renamed to 'option_series_id'. • Stream 'FORTS_MISCINFO_REPL': ○ Table 'volat_coeff': field 'isin_id' renamed to 'option_series_id'. • Stream 'FORTS_INFO_REPL': ○ table 'option_series_params': field 'isin' renamed to 'small_name', field 'exp_date' renamed to 'expiration_date'. Added fields 'option_series_id', 'underlying_id'. ○ table 'base_contracts_params': deleted field 'currency_volat'.
26.09.2018	New error code added (4208).
25.09.2018	Added section '3.3.7. SMA Login (Sponsored Market Access)'.
03.08.2018	Field 'aspref' deleted from table 'orders' of stream 'FORTS_ORDBOOK_REPL'.
01.08.2018	Tables 'sma_master', 'sma_pre_trade_check' deleted from stream FORTS_INFO_REP
31.07.2018	Stream 'FORTS_INFO_REPL' now contains table 'option_series_params'.
30.07.2018	Field 'coeff_out' deleted from table 'opt_vcb' of stream 'FORTS_OPTINFO_REPL'.
27.07.2018	Changes applied: • Changed descriptions for fields 'is_cupon' of tables 'fut_bond_nkd', 'fut_bond_nominal' • Renamed table 'fut_bond_nkd'.
26.07.2018	New error code added (4220).
26.07.2018	Changes applied: • Stream 'FORTS_CLR_REPL': ○ fields 'pos_beg', 'pos_end' deleted from tables 'fut_pos', 'opt_pos', 'fut_pos_sa', 'opt_pos_sa' ○ fields 'amount_beg', 'pay, amount', 'amount_beg_money', 'pay_money', 'amount_money' deleted from table 'pledge_details'. • Stream FORTS_DEALS_REPL: ○ fields 'pos', 'amount' deleted from table 'deal'. ○ field 'amount' deleted from table 'multileg_deal'. • Stream 'FORTS_FUTCOMMON_REPL': fields 'amount_buy', 'orders_buy_amount', 'amount_sell', 'orders_sell_amount', 'amount', 'contr_count', 'pos' deleted from table 'common'. • Stream FORTS_OPTCOMMON_REPL: fields 'amount_buy', 'orders_buy_amount', 'amount_sell', 'orders_sell_amount', 'amount', 'contr_count', 'pos' deleted from table 'common'. • Stream 'FORTS_MM_REPL': fields 'amount_sells', 'amount_buys', 'mm_amount' deleted from tables 'fut_MM_info', 'opt_MM_info'. • Stream 'FORTS_OPTINFO_REPL': ○ field 'amount' deleted from table 'opt_rejected_orders' ○ fields 'amount', 'amount_apply' deleted from table 'opt_exp_orders'. • Stream 'FORTS_ORDLOG_REPL': fields 'amount', 'amount_rest', 'status' deleted from tables 'orders_log', 'multileg_orders_log'. • Stream 'FORTS_POS_REPL': fields 'pos', 'buys_qty', 'sells_qty', 'open_qty' deleted from tables 'position', 'position_sa'. • Stream 'FORTS_FUTINFO_REPL': ○ field 'amount' deleted from table 'fut_rejected_orders'. ○ added tables 'user', 'sma_master', 'sma_pre_trade_check' ○ deleted table 'fut_sess_settl' ○ table 'fut_margin_type' now contains field 'type'

Date	Changes
	○ changed descriptions on fields 'UCP_type', 'prohibit_coeff'. • Stream 'FORTS_INFO_REPL': ○ deleted table 'opt_sess_settl' ○ field 'min_vol' deleted from table 'opt_vcb'. • Stream 'FORTS_FUTTRADE_REPL': ○ tables 'orders_log', 'multileg_orders_log' now contain field 'aspref' ○ fields 'amount', 'amount_rest', 'status' deleted from tables 'orders_log', 'multileg_orders_log'. ○ fields 'pos', 'amount', 'status_buy', 'status_sell' deleted from table 'user_deal'. ○ fields 'amount', 'status_buy', 'status_sell' deleted from table 'user_multileg_deal'. • Stream 'FORTS_OPTTRADE_REPL': ○ table 'orders_log' now contain field 'aspref' ○ fields 'amount', 'amount_rest', 'status' deleted from table 'orders_log' ○ fields 'pos', 'amount', 'status_buy', 'status_sell' deleted from table 'user_deal'. • Stream 'FORTS_ORDBOOK_REPL': ○ table 'orders' now contain field 'aspref' ○ fields 'status', 'amount', 'amount_rest', 'init_amount' deleted from table 'orders'. • Streams 'FORTS_FUTORDERBOOK_REPL'/'FORTS_OPTORDERBOOK_REPL': ○ table 'orders' now contain field 'aspref' ○ fields s'tatus', 'amount', 'amount_rest', 'init_amount' deleted from table 'orders'. • Stream 'RTS_INDEX_REPL': table 'rts_index' now contains fields 'value_highprec', 'prev_close_value_highprec', 'open_value_highprec', 'max_value_highprec', 'min_value_highprec'.
18.07.2018	Add commands 'SetSmaPreTradeCheck, 'DelSmaPreTradeCheck', 'UserKillSwitch'.
25.06.2018	New error codes added (76, 4167, 4200 - 4207).
21.06.2018	Added section '3.3.6. Replication stream sets for different login subtypes'.
19.06.2018	Removed unused fields ’limit_pledge’, ‘coeff_liquidity’ of command FutChangeClientMoney.
21.05.2018	Field 'signs' added into table 'diler' of stream 'FORTS_FUTINFO_REPL'.
11.04.2018	Changed message type for messages 'OptChangeExpiration', 'FutTransferClientPosition', 'OptTransferClientPosition'. Also, changed type of the field 'amount' in these messages.
30.03.2018	Table 'part_sa' of stream 'FORTS_PART_REPL' now contains a new field 'money_old'.
22.03.2018	Changes applied: • Field 'signs' added into table 'dealer' of stream 'FORTS_FUTINFO_REPL'. • Fields 'strike_step', 'exp_clearings_bf' and 'exp_clearings_cc' added into table 'virtual_futures_params' of stream 'FORTS_INFO_REPL'. • Field 'lot' added into table 'futures_params' of stream 'FORTS_INFO_REPL'. • Fields 'has_options', 'msp_type' and 'currency_id' added into table 'base_contracts_params' of stream 'FORTS_INFO_REPL'. • Tables ‘currency_params' and 'common_params' added into stream 'FORTS_INFO_REPL'.
28.02.2018	Table 'part_forecast' deleted from stream 'FORTS_FORECASTIM_REPL'.
26.02.2018	Changes applied: • Fields 'client_code', 'exch_pay', 'exch_pay_scalped', 'clear_pay', 'clear_pay_scalped', 'exch_pay_spot', 'exch_pay_spot_repo', 'sell_fee' and 'buy_fee' deleted from table 'fut_vcb' of stream 'FORTS_FUTINFO_REPL'. • Fields 'client_code', 'exch_pay', 'exch_pay_scalped', 'clear_pay', 'clear_pay_scalped', 'is_spec', 'spec_spread', 'sell_fee' and 'buy_fee' deleted from table 'opt_vcb' of stream 'FORTS_OPTINFO_REPL'.

Date	Changes
21.02.2018	Added new error codes: 4148, 4149.
20.02.2018	Changes applied: • Added description of command 'FutChangeBFLimit'. • Fields 'money_blocked' and 'vm_reserve' added into table 'part_sa' of stream 'FORTS_PART_REPL'.
31.01.2018	Changes applied: • Field 'ext_reserve' deleted from table 'money_clearing' of stream 'FORTS_CLR_REPL' . • Field 'coeff' deleted from tables 'fut_sess_contents' and 'fut_instruments' of stream 'FORTS_FUTINFO_REPL'. • Table 'fut_bond_registry' of stream 'FORTS_FUTINFO_REPL': field 'bond_type' type changed to i4. • Tables 'deal' and 'multileg_deal deleted from stream 'FORTS_FUTTRADE_REPL. • Table 'deal' deleted from stream 'FORTS_OPTTRADE_REPL'. • Field 'points_num' deleted from table 'base_contracts_params' of stream 'FORTS_INFO_REPL'. Fields 'spot_price', 'mr1', 'mr2', 'mr3', 'lk1', 'lk2', 'risk_points_n' added into table 'base_contracts_params' of stream 'FORTS_INFO_REPL'. • Fields 'limit' and 'spot_signs' deleted from table 'futures_params' of stream 'FORTS_INFO_REPL'. Fields 'interest_rate_risk_up', 'interest_rate_risk_down', 'time_to_expiration', 'normalized_spot' added into table 'futures_params' of stream 'FORTS_INFO_REPL'. • Fields 'is_net_positive', 'volat_range', 't_squared' and 'max_addrisk' deleted from table 'virtual_futures_params' of stream 'FORTS_INFO_REPL'. Fields 'exp_clearings_sa', 'volatility_risk', 'volatility_risk_mismatch', 'time_to_expiration' added into table 'virtual_futures_params' of stream 'FORTS_INFO_REPL'. • Field 'server_time' added into table 'sys_events' of stream 'FORTS_INFO_REPL'. • Field 'isin_is_spec' deleted from table 'common' of stream 'FORTS_OPTCOMMON_REPL'. • Fields 'pledge_free', 'pledge_blocked', 'coeff_liquidity', 'pledge_old', 'pledge_amount' deleted from table 'part' of stream 'FORTS_PART_REPL'. • Fields 'pledge_amount' and 'liquidity_ratio' deleted from table 'part_sa' of stream 'FORTS_PART_REPL' . Fields 'vm_intercl' and 'fee' added into table 'part_sa' of stream 'FORTS_PART_REPL'. • Added description of stream 'FORTS_FEERATE_REPL - Precise Exchange fee rates'. • Added description of commands 'FutChangeBFParameters', 'FutChangeClientParameters' and 'FutChangeBF-ClientDefaultParameters'. • Fields 'exp_weight', 'num_clr_2delivery', 'margin_type', 'calendar_spread_margin_type', 'num_clr_2delivery_client_default', 'exp_weight_client_default', 'go_ratio', 'check_limit_on_withdrawal', 'limit_tied_money', 'limits_set', 'no_fut_discount', 'no_fut_discount_client_default' added into table 'diler' of stream 'FORTS_FUTINFO_REPL'. • Field 'calendar_spread_margin_type' added into table 'investr' of stream 'FORTS_FUTINFO_REPL'. • Tables 'dealer' and 'investor' added into stream 'FORTS_FUTINFO_REPL'. • Tables 'dealer' and 'investor' added into stream 'FORTS_INFO_REPL'.
26.12.2017	Changes applied: • Table 'position' of stream 'FORTS_POS_REPL' now contains a new field 'account_type' • Stream 'FORTS_POS_REPL' now contains a new table 'position_sa'.
21.12.2017	Added new error codes (4160 - 4166).
16.11.2017	Description change for parameter 'code_vcb' of method 'FutDelUserOrders'.
25.10.2017	Changes applied: • Table 'delivery_report' removed from stream FORTS_FUTINFO_REPL • Table 'fut_rejected_orders' of stream 'FORTS_FUTINFO_REPL' now contains a new field 'xamount' • Tables 'opt_rejected_orders' and 'opt_exp_orders' of stream 'FORTS_OPTINFO_REPL' now contain a new field 'xamount' • Table 'opt_exp_orders' of stream 'FORTS_OPTINFO_REPL' now contains a new field 'xamount_apply'.

Date	Changes
24.10.2017	Changes applied: • Table 'fut_MM_info' of stream 'FORTS_MM_REPL' now contain fields 'xamount_sells', 'xamount_buys', 'xmm_amount' • Table 'opt_MM_info' of stream 'FORTS_MM_REPL' now contain fields 'xamount_sells', 'xamount_buys', 'xmm_amount'.
28.08.2017	Changed message type for messages 'OptChangeExpiration', 'FutTransferClientPosition', 'OptTransferClientPosition'. Also, changed type of the field 'amount' in these messages.
23.06.2017	Deletion of stream RTS_INDEXLOG_REPL.
02.06.2017	Changes applied: • Table 'multileg_dict' of stream 'FORTS_FUTINFO_REPL' now contain field 'leg_order_no'. • Table 'fut_margin_type' of stream 'FORTS_FUTINFO_REPL' now contain fields 'UCP_type', 'prohibit_coeff', 'prohibit_type'.
18.05.2017	Changes applied: • Tables 'fut_pos', 'opt_pos', 'fut_pos_sa' and 'opt_pos_sa' of stream 'FORTS_CLR_REPL' now contain fields 'xpos_beg' and 'xpos_end'. • Table 'pledge_details' of stream 'FORTS_CLR_REPL' now contains fields 'xamount_beg', 'xpay', 'xamount', 'xamount_beg_money', 'xpay_money', 'xamount_money'.
15.05.2017	Changes applied: • Tables 'common' of streams 'FORTS_FUTCOMMON_REPL' and 'FORTS_OPTCOMMON_REPL' now contain fields 'xamount_buy', 'xorders_buy_amount', 'xamount_sel'l, 'xorders_sell_amount', 'xamount', 'xcontr_count', 'xpos'. • Tables 'orders' of streams 'FORTS_ORDBOOK_REPL', 'FORTS_FUTORDERBOOK_REPL' and 'FORTS_OPTORDERBOOK_REPL' now contain fields 'xamount', 'xamount_rest', 'xinit_amount'. • Table 'position' of stream 'FORTS_POS_REPL' now contains fields 'xpos', 'xbuys_qty', 'xsells_qty', 'xopen_qty'.
05.05.2017	Changes applied: • Table 'deal' of stream 'FORTS_DEALS_REPL' now contains fields 'xpos' and 'xamount'. • Table 'multileg_deal' of stream 'FORTS_DEALS_REPL' now contains field 'xamount'. • Table 'orders_log' of stream 'FORTS_FUTTRADE_REPL' now contains fields 'xamount' and 'xamount_rest'. • Table 'multileg_orders_log' of stream 'FORTS_FUTTRADE_REPL' now contains fields 'xamount' and 'xamount_rest'. • Table 'deal' of stream 'FORTS_FUTTRADE_REPL' now contains fields 'xpos' and 'xamount'. • Table 'multileg_deal' of stream 'FORTS_FUTTRADE_REPL' now contains field 'xamount'. • Table 'user_deal' of stream 'FORTS_FUTTRADE_REPL' now contains fields 'xpos' and 'xamount'. • Table 'user_multileg_deal' of stream 'FORTS_FUTTRADE_REPL' now contains field 'xamount'. • Table 'orders_log' of stream 'FORTS_OPTTRADE_REPL' now contains fields 'xamount' and 'xamount_rest'. • Table 'deal' of stream 'FORTS_OPTTRADE_REPL' now contains fields 'xpos' and 'xamount'. • Table 'user_deal' of stream 'FORTS_OPTTRADE_REPL' now contains fields 'xpos' and 'xamount'. • Table 'orders_log' of stream 'FORTS_ORDLOG_REPL' now contains fields 'xamount' and 'xamount_rest'. • Table 'multileg_orders_log' of stream 'FORTS_ORDLOG_REPL' now contains fields 'xamount' and 'xamount_rest'.
24.03.2017	Changes applied: • Fields exch_pay, exch_pay_scalped, clear_pay, clear_pay_scalped, sell_fee, buy_fee, exch_pay_spot, exch_pay_spot_repo, client_code of table fut_vcb of stream FORTS_FUTIINFO_REPL contain default values (nulls, empty strings). • Fields exch_pay, exch_pay_scalped, clear_pay, clear_pay_scalped, sell_fee, buy_fee, is_spec, spec_spread, client_code of table opt_vcb of stream FORTS_OPTINFO_REPL contain default values (nulls, empty strings).
28.12.2016	Changes applied:

Date	Changes
	• Added section 'Stream FORTS_FORECASTIM_REPL - Risk forecast after limits extension'. • Field 'exp_weight' deleted from table 'part' of stream 'FORTS_PART_REPL'.
21.12.2016	In accordance with decommission policy, starting from December 5, 2016, P2ClientGate API and Plaza2 libraries v.198 or below are no longer supported. The client software using Plaza2 libraries version 198 and below, or P2ClientGate API will no longer be able to connect to the trading system.
30.08.2016	Changed list of synchroevents in table sys_events of streams FORTS_PART_REPL, FORTS_CLR_REPL, FORTS_INFO_REPL.
18.05.2016	Changes applied: • Deleted description of methods for working with Spots: ○ 'FutChangeBrokerVcb - Changing BF parameters on Underlying'; ○ 'FutChangeClientVcb - Changing client parameters on Underlying'; ○ 'FutChangeMoney - Changing limit for bying spots on BF'. • Table 'fut_instruments' of stream 'FORTS_FUTINFO_REPL ' now contains field 'exec_name' (Flag of option maturity). • Added description of method 'OptChangeRiskParameters - Risk-parameter management for options'. • Field 'num_clr_2delivery' deleted from message 'FutChangeClientMoney - Changing client limits'. If filled in, this field will be ignored in all previous versions of the message. • Added description of method 'FutTransferRisk - Risk balancing'. • Added return codes: 75, 331, 339, 383, 4127, 4138, 4139, 4150-4155, 9999, 10000, 10001, 10004-10006. • Stream 'FORTS_FUTINFO_REPL' now contains tables 'fut_settlement_account' and 'fut_margin_type'. • Table 'part_sa' added into stream 'FORTS_PART_REPL'. • Tables 'money_clearing_sa', 'fut_pos_sa', 'opt_pos_sa' are added into stream 'FORTS_CLR_REPL'. • Tables 'fut_vm_sa' and 'opt_vm_sa' added into stream 'FORTS_VM_REPL'. • Table 'part' of stream 'FORTS_PART_REPL' now contains fields 'num_clr_2delivery' and 'exp_weight'. • Field 'cal_exp_extra_risk' deleted from table 'part' of stream 'FORTS_PART_REPL'. • Table 'virtual_futures_params' of field 'FORTS_INFO_REPL' now contains fields 'exp_clearings_bf' and 'exp_clearings_cc'. • Fields 'allow_use_extra_exp_risk' and 'calc_extra_exp_risk' deleted from table 'virtual_futures_params' of stream 'FORTS_INFO_REPL'.
14.10.2015	Added description of 'CODHeartbeat' method.
14.10.2015	Table 'fut_sess_contents' now contains 2 new fields: 'pctyield_coeff' and 'pctyield_total'.
12.08.2015	Added new error codes (200 - 208).
23.01.2015	'Trading gate description' now contains section 'Handling abnormal situations'.
22.01.2015	Added section "Cancel on Disconnect".
16.12.2014	Edited list of error codes.
29.09.2014	Added details of table 'prohibition' of stream 'FUTINFO'.
18.08.2014	Added ASTS error codes.
24.07.2014	Tables 'fut_MM_info' and 'opt_MM_info' of stream 'FORTS_MM_REPL' now have contains market-makers obligations accurate to 7-symbol client code. Formats of mesages 'FutTransferClientPosition' and 'OptTransferClientPosition' are now equal. Table 'fut_ts_cons' deleted from stream 'FORTS_FUTINFO_REPL'.
17.07.2014	Field 'client_code' deleted from table 'ORDERS' of stream 'FORTS_ORDBOOK_REPL'.
25.04.2014	Stream 'FORTS_MM_REPL' now contains new table 'mm_agreement': table with numbers and types of contracts for the provision of marketmaking services.
15.04.2014	Added new commands: • Transfer futures position between BF

Date	Changes
	• Transfer option position between BF
14.01.2014	Added new fields: • 'fulfil_min' - Minimum percentage of the liabilities for the trading session 'fulfil_partial' - Percentage of partial fulfillment of the liabilities of the trading session 'fulfil_total' - Percentage of fulfillment of liabilities of the trading session 'is_fulfil_min' - Minimum sign of the liabilities for the trading session 'is_fulfil_partial' - Sign of partial fulfillment of the obligations of the trading 'is_fulfil_total' - Sign of fulfillment of obligations of the trading session into tables 'fut_MM_info', 'opt_MM_info' of stream 'FORTS_MM_REPL'.
31.05.2013	New field added: • 'rate_id' - Payment currency identifier into table 'clr_rate' of field 'FORTS_CLR_REPL'.
18.04.2013	Added anonymous stream 'orderbook': • 'FORTS_ORDBOOK_REPL' Added field: • 'ext_reserve' - Extra reserve into table 'money_clearing' of stream 'FORTS_CLR_REPL' Deleted stream 'FORTS_CLMONEY_REPL'.
12.04.2013	New field added: • 'exch_pay' - Exchange fee per 1 contract in Russian rubles. into table 'fut_sess_contents' of stream 'FORTS_FUTINFO_REPL'.
10.04.2013	New field added: • 'exch_pay' - Exchange fee per 1 contract in Russian rubles. into table 'opt_sess_contents' of stream 'FORTS_OPTINFO_REPL'.
26.03.2013	New field added: • 'rate_id' - Payment currency identifier into tables 'fut_vcb' and 'opt_vcb' of streams 'FORTS_FUTINFO_REPL' and 'FORTS_OPTINFO_REPL' Replication stream added: • 'MOEX_RATES_REPL' - Exchange rates in online mode. Added new table: • 'rates' - Currency exchange rates reference book. into stream 'FORTS_FUTINFO_REPL'.
27.11.2012	Changed description of table 'user_deal'.
01.11.2012	Added descriptions of two events for table 'sys_events'.
30.10.2012	Some changes applied: • Section 'FutChangeMoney - Changing limit for bying spots on BF' now contains extended description of parameter 'limit_spot_buy'. • Sections 'Method FutMoveOrder - Modify order' and 'Method OptMoveOrder - Modify order' now contain extended description of the command 'MoveOrder' logic.
22.10.2012	Some changes applied: • Modified sections: 'Users. How a user is linked to a trading participant', 'SPECTRA Plaza-2 gateway. Components, installation and setup', 'Recommendations for third-party companies on including the Moscow Exchange runtimes into user application when distributing the user software', 'Recovery and late logon'.

Date	Changes
	• Deleted section 'Technical center interface'.
10.02.12	Some changes applied: • Section 'Gate usage specifics' now contains subsection 'Commands'. • Added section 'Pausing trading session for extending limits of trading prices fluctuations'. • Corrected an error in futures price calculation formula. • Updated the Gate installator description.
09.02.2012	New field added: • 'login_from' - Login of a user who added the order into tables: • 'fut_rejected_orders' - Orders rejected during clearing session • 'opt_rejected_orders' - Orders rejected during clearing session of streams: • 'FORTS_FUTINFO_REPL' - Futures: Reference and session information • 'FORTS_OPTINFO_REPL' - Options: Reference and session information
24.01.2012	Table 'orders' of streams: • 'FORTS_FUTORDERBOOK_REPL' - Futures: orderbook snapshot • 'FORTS_OPTORDERBOOK_REPL' - Options: orderbook snapshot now contains fields: • 'init_moment' - Time of the order adding • 'init_amount' - Initial amount in the order
23.01.2012	Event table 'sys_events' added into streams: • 'FORTS_CLMONEY_REPL' - Money in clearing session • 'FORTS_CLR_REPL' - Clearing data
17.01.2012	Field 'exch_pay_spot_repo' containing Exchange fee value on repo added into table 'fut_vcb' of field 'FORTS_FUTINFO_REPL'.
12.01.2012	Added new replication stream: • 'FORTS_ORDLOG_REPL' - the stream transmits anonymized orders events.
02.11.2011	New fields added: • 'comment' - Trader's comment • 'ext_id' - External number into tables: • 'fut_rejected_orders' - Orders rejected during clearing session • 'opt_rejected_orders' - Orders rejected during clearing session
25.11.2011	Added section 'Usage of test examples'.
7.11.2011	Completed sections 'Introduction' and 'Trading gate description'. Added section 'System SPECTRA overview'.
20.10.2011	Fields added: • 'theor_price_limit' - Theoretical option price with limits • 'vm_real' - The accumulated variation margin on futures trades calculated based on the current market quote. Event table 'sys_events' added into streams: • 'FORTS_FUTTRADE_REPL' - Futures: orders and trades • 'FORTS_OPTTRADE_REPL' - Options: orders and trades

Date	Changes
	• 'FORTS_POS_REPL' - Information on positions • 'FORTS_PART_REPL' - information on funds and limits • 'FORTS_FUTINFO_REPL' - Futures: reference and session information • 'FORTS_OPTINFO_REPL' - Options: reference and session information • 'FORTS_INFO_REPL' - Additional reference information
4.10.2011	Added replication streams: • 'FORTS_CLR_REPL' - Various clearing information. • 'FORTS_MM_REPL' - Information on MM's liabilities Changed numbers of trading commands for complete processing time monitoring possibility (including the final link to the client transmission time).
14.09.2011	Corrected errors in default values of some commands. Now all string parameters' values are quoted by default.
15.04.2011	Added fields: • 'status' of table 'diler' of stream 'FORTS_FUTINFO_REPL' - CF and BF accounts information • 'status' of table 'investr' of stream 'FORTS_FUTINFO_REPL' - client accounts information • 'vm_order_reserve' of stream 'FORTS_PART_REPL' - Amount reserved for negative variation margin on orders • 'waprice' of stream 'FORTS_POS_REPL' - weighted average price Changes applied to commands structure: • *NOTE*: Changes applied to the format of the following commands: 'FutAddOrder', 'OptAddOrder' and 'FutAddMultilegOrder' - now each of the commands contains parameter 'dont_check_money'. Commands' identifiers have also changed. All prevoius identifiers are still valid with commands in previous format.. • Added command 'FutExchangeBFMoney' which allows to transfer funds between the accounts of a BF.
28.03.2011	Table 'multileag_deal' of stream 'FORTS_FUTTRADE_REPL' now contains field 'buyback_amount' - buyback amount for repo trades.
24.03.2011	Added stream 'RTS_INDEXLOG_REPL', which transmits history of RTS indexes changes.
01.02.2011	For command 'FutChangeClientVcb', parameter 'code_vcb' type has changed from 'c4' to 'c25'. The new command format now has message code 33. The return code for the command has not changed. Added list of return codes.
27.01.2011	An error corrected in description of parameter 'check_limit' of commands 'OptAddOrder' and 'OptMoveOrder'. The correct values are the following: 0 - do not verify, 1 - verify.
24.12.2010	Corrected some errors in command fields names along with default values of some commands: • The new default value of parameter 'ext_id' of command 'FutDelUserOrders' is now 0. • The new default value of parameters 'comment', 'hedge', 'broker_to', 'ext_id', 'trust', 'date_exp' of command 'FutAddMultiLegOrder' is now 0 or empty string, depending on the message type. • The new default value of parameters 'price1' and 'price2' of command 'OptMoveOrder' is now 0. • The new default value of parameter 'no_fut_discount' of command 'FutChangeClientMoney' is now 0. • The new default value of parameter 'limit_spot' of command 'FutChangeBrokerVcb' is now -1. • The return messages for commands 'FutChangeClientMoney', 'FutChangeBFMoney', 'FutChangeClientVcb' and 'OptChangeExpiration' now have their field 'Message' changed to 'message'. .
26.11.2010	Aggregated orderbooks no more contain field 'price2'. Field 'price' now has different meaning depending on the instrument flag 0x1000 (field 'signs' of table 'fut_sess_contents' of stream 'FORTS_FUTINFO_REPL') presense. If the flag is applied, the field 'price' contains rate value, otherwise it contains swap-price value.
15.10.2010	Added new instrument flags (field 'signs' of table 'fut_sess_contents' of stream 'FORTS_FUTINFO_REPL'): • 0x800 - flag of an RTS Money instrument • 0x1000 - flag of basic price for multileg instruments (0 - quoted in swap-price, 1 - quoted in rate).

Date	Changes
	Flag of multileg instruments 'multileg_type' (table 'fut_sess_contents' of stream 'FORTS_FUTINFO_REPL') value is now 2 for RTS Money swaps. Aggregated orderbooks now have a new field 'price2', which contains swap-price value.
14.09.2010	Streams 'FORTS_FUTCOMMON_REPL' and 'FORTS_OPTCOMMON_REPL' now contain opening price values and closing price values (fields 'open_price' and 'close_price'). Stream 'RTS_INDEX_REPL' now contains cap value and indexes volume value (fields 'cap' and 'volume').
07.07.2010	Table 'session' of stream 'FORTS_FUTINFO_REPL' now contains info on position transfer interval (fields 'pos_transfer_begin' and 'pos_transfer_end'). Added tables: • 'fut_sess_settl' of stream 'FORTS_FUTINFO_REPL' containing settlement prices values during the last clearing session. • 'opt_sess_settl' of stream 'FORTS_OPTINFO_REPL' containing volatility and option theoretical price values during the clearing session.
15.06.2010	Corrected an error in command 'FutAddMultiLegOrder' description: parameter 'isin_id' is now i4.
	In table 'delivery_report' of stream 'FORTS_FUTINFO_REPL', fields 'oblig_uni' and 'fulfil_uni', type i4, are replaced with fields 'oblig_qty' and 'fulfil_qty', type i8.
31.05.2010	Tables 'fut_sess_contents' and 'fut_instruments' of stream 'FORTS_FUTINFO_REPL' now contain field 'step_price_curr'. Table 'common' of streams 'FORTS_FUTCOMMON_REPL' and 'FORTS_OPTCOMMON_REPL' now contains fields with aggregated bid and offer values: 'orders_sell_qty', 'orders_sell_amount', 'orders_buy_qty', 'orders_buy_amount'.
17.05.2010	Added info on instruments parameters: • tables 'base_contracts_params', 'futures_params', 'virtual_futures_params', 'options_params' Added field 'step_price_clr' of table 'fut_sess_contents' of stream 'FORTS_FUTINFO_REPL', containing info on instrument price tick value during the evening clearing session. Added field 'step_price_interclr' of table 'fut_sess_contents' of stream 'FORTS_FUTINFO_REPL', containing info on instrument price tick value during the intermediate clearing session.
19.04.2010	Changes applied to many fields types, including but not limited to: • aggregated orderbook volume d16.5 -> i8 • order direction i4 -> i1 • flags of instruments (signs) i1 -> i4 Table 'money_clearing' is relocated from stream 'FORTS_FUTINFO_REPL' into stream 'FORTS_CLMONEY_REPL'. Objects renamed: • table 'repo_orders_log' -> 'multileg_orders_log' • table 'repo_deal' -> 'multileg_deal' • command 'FutAddRepo' -> 'FutAddMultiLegOrder' Added: • table 'multileg_dict' – multileg instruments dictionary • fields 'price_dir', 'multileg_type', 'legs_qty', tables 'fut_sess_contents' • fields containing IDs and trade prices are added into tables 'orders_log' of streams 'FORTS_FUTTRADE_REPL' and 'FORTS_OPTTRADE_REPL' • fields 'fee_sell', 'fee_buy' of table 'deal' of stream 'FORTS_OPTTRADE_REPL' • streams 'FORTS_FUTORDERBOOK_REPL' and 'FORTS_OPTORDERBOOK_REPL', transmitting current orderbooks data • table 'broker_params' of stream 'FORTS_INFO_REPL' • table 'fut_instruments' of stream 'FORTS_FUTINFO_REPL' • tables 'usd_online' of stream 'FORTS_FUTINFO_REPL'

Date	Changes
	Deleted: • field 'state' of table 'opt_sess_contents'
16.03.2010	Changed description of command 'FutAddRepo': instead of parameter 'swap_price', parameter 'repo_rate' is now used.
24.02.2010	Added: • description of tables 'repo_orders_log', 'repo_deals' • description of orders and repo trades statuses • descriptions of new statuses for orders and trades • description of command 'FutAddRepo' • field 'last_deal_id' of table 'position' of stream 'FORTS_POS_REPL'
18.01.2010	• Added descriptions of commands: 'FutChangeBrokerVcb', 'FutChangeClientProhibit', 'FutChangeMoney', 'OptChangeClientProhibit' • Added field 'limits_set' of table 'part' of stream 'FORTS_PART_REPL' • Corrected some mistakes in commands descriptions
15.01.2010	• Changed ID types of orders and trades (i4 -> i8) • Changed status types of orders and trades (i2 -> i4) • Corrected some mistakes in commands descriptions
25.11.2009	Corrected some mistakes in commands descriptions
03.11.2009	Now it is possible to specify BF codes when sending messages
30.10.2009	Added commands for setting client limits
10.08.2009	Added dictionary on option instruments
15.07.2009	Added description of reference replication streams
17.06.2009	Added descriptions of commands for managing futures and options orders.
27.03.2009	Added description of replication streams 'common'.
20.03.2009	Initial version of the document

1. Introduction

1.1. Document purpose

This document is aimed to overview all the details which users may demand to architect and develop software applications for accessing the derivatives market using the PLAZA II gateway software (SPECTRA). The following parts are available in this document:

- The SPECTRA system general overview, including overview of trading instruments, trading participants, trading operations, risk management, limiting of operations, etc.
- Configuration, installation and setup of the PLAZA II gateway software in the form of user manuals on software installation and setup with information on minimum hardware and software requirements. Also, some general references on using the PLAZA II gateway software are added.
- Information on the structure of transmitted data, including description of replication streams and transmitted tables.
- List of commands.
- Help information.

1.2. Target audience

This document is intended for business-analysts, system architects and developers, taking part in architecting and developing software for accessing the derivatives market using the PLAZA II gateway software.

1.3. Abbreviations

The following abbreviations may be used in the Document:

Abbreviation	Description
ASTS	Trading and clearing system of the FX Market of the Moscow Exchange
BF	Brokerage Firm (trading member)
CF	Clearing Firm (clearing member)
CM	Clearing Member
COD	Service 'Cancel On Disconnect'
EDM	Electronic Document Management
MM	Market Maker
NCC	National Clearing Centre
SMA	Service 'Sponsored Market Access'
TM	Trading Member
TS	Trading System
VM	Variation Margin

2. SPECTRA system overview

2.1. Trading Membership

Trading Membership may be subdivided to the following:

- Clearing Members (Clearing Firms)
- Trading Members (Brokerage Firms)
- Trading Member's clients and Clearing Member's clients.

Traditionally, Clearing Member and Trading Member belong to the same entity, i.e. the entity which perform trades and act as a counterparty for the performed trades. The information below is provided regarding exactly this kind of Trading Membership. However, effective SPECTRA version 6.2, the Derivatives Market rolls out a new model, with Trading Member and Clearing Member represented by two separate entities (for details see 2.7. Separate entities of Clearing Member and Trading Member). Please also note that the new model will in no way affect the existing trading members!

2.1.1. Clearing firms

Clearing firms are firms which incur liabilities for risks and cover risks of their clients and sub-brokers.

Clearing firms are authorized to:

- Perform trades on behalf of themselves and at for their own accounts;
- Perform trades on behalf of themselves and for their clients' accounts;
- Perform settlement directly with National Clearing Centre.
- Service their clients, including brokers;
- Exercise control over their clients and brokers during trading sessions.

Clearing firms are obliged to:

- Become members of Derivatives Market Section;
- Pay fees to Guarantee Fund;
- Provide collaterals for their own trades and for their clients' trades.

2.1.2. Brokerage firms

Unlike clearing firms, brokerage firms do not settle up with exchange directly; instead, they use their clearing firms. Also, brokerage firms are not obliged to obtain licences and pay fees to the Guarantee Fund.

Brokerage firms are authorized to:

- Perform trades on behalf of themselves;
- Perform trades on behalf of their clients;
- Place orders in the Trading system via the client terminal application
- Exercise control over their clients during trading sessions.

Brokerage firms are obliged to:

- Provide guarantees for their own trades and for their clients' trades.

2.1.3. Clients

Any physical or corporate person can participate in the SPECTRA market as a client on the authority of trading service agreement signed with a brokerage firm or with clearing firm directly.

2.1.4. System code pattern

There is a 7-symbol code pattern (XXYYZZZ) to identify each participant in the system, where

- XX — indicates a clearing firm
- YY — indicates a brokerage firm

- ZZZ — indicates a client

The 00 brokerage firm code indicates state of account of the clearing firm.

Example 1.

Q100 — indicates the Q1 clearing firm

Q1DU — indicates the DU sub-broker of the Q1 clearing firm

The 000 client code indicates state of account of the brokerage firm.

Example 2.

Q1DU000 — indicates state of account of the DU sub-broker of the Q1 clearing firm

2.1.5. Disclosure of data on participants

The list of clearing and brokerage firms is stored in the 'dealer' table of the 'FORTS_REFDATA_REPL stream, and the list of clients is stored in the 'investor' table of the FORTS_REFDATA_REPL stream. Disclosure of data on brokerage firms and clients is limited in accordance with user access rights.

Streams and tables also contain links to 7-symbol clients' codes and 4-symbol brokerage firms' codes.

2.1.6. Users. How a user is linked to a trading participant

A user (login) can be associated with various levels of participants:

- Clearing firm login. Users connected with this login are allowed to view data and perform trading operations on behalf of any brokerage firm or of any client of the clearing firm (please note that performing trading actions is only allowed when the user has sufficient rights!). Users also allowed to set limits for clients and sub-brokers by calling the appropriate operations.
- Brokerage firm login. Users connected with this login are allowed to view data and perform trading operations on behalf of all broker's clients within the clearing firm, and also set limits for the broker's clients.
- Client login. Users connected with this login are allowed to perform trading operations on behalf of a certain client of a brokerage firm and view data in accordance with the client login rights.

There is a special 4-symbol 'broker_code' field within the scheme of message-command (see Commands description). Every application using the clearing firm account is to fill in this field with a 4-symbol code of a brokerage company registered with SPECTRA when sending messages. Applications which use the client or the brokerage firm account are exempt from this rule.

2.2. Instruments

The SPECTRA instruments are structured hierarchically. Below you will find descriptions of the SPECTRA instruments starting from the root level.

2.2.1. Underlying assets

An underlying asset is an entity related to a certain contract. Therefore, it can be a stock in a stock exchange, a lot of tradable commodity in a commodities exchange or an index/exchange rate/indicator for settling futures. There are certain attributes characterising an underlying asset along with its instruments, which are:

- Trade section name;
- Various commision fees rates.
- Delivery type according to the contract (for details, see Delivery of assets and expiration of options):
  - Delivery of the asset itself;
  - Settlement type. The margin between the opening price and the closing price is the single amount of money to be paid after the trade is closed.
- Price step calculation currency. Now it can be one of the following:
  - RUB — when cost of price step is indicated in Russian roubles. The cost of price step is not typically a subject to change during the life of contract;
  - USR — when cost of price step is indicated in Russian roubles. The cost is converted into USD by using a special Moscow Exchange method of conversion (for details, see http://moex.com/n6126).
- Types of trading, where two types are existing: collateralized and non-collateralized. For the collateralized trading, a part of deposit can be pledged by transferring shares and other securities in accordance with the authorized list.

An underlying asset IS IN NO WAY A TRADING INSTRUMENT!

Data concerning underlying assets are contained in the 'fut_vcb' table of the 'FORTS_REFDATA_REPL stream.

2.2.2. Futures

Futures contracts are the main trading instruments in the SPECTRA system.

Each futures contract is linked to a certain underlying asset and has its own unique characteristics of the maturity (the date of delivery), lot characteristics, minimum price step and cost of the price step value.

Delivery dates for futures in the trading system are set at three-month (quarterly futures) or monthly (monthly futures) intervals. There can be more than one futures contract for each underlying asset.

Futures contract with various dates of delivery may form a calendar spread. In this case, when risks are calculated, the price correlation is always taken into account. As a result, the total collateral for the spread can be less than sum of collaterals for each futures contract itself.

Futures are quoted in price points. The price in roubles for a contract is calculated as following:

$$\text{PriceRub} = \text{PricePoints} * \frac{\text{step_price}}{\text{min_step}}$$

, where:

- PricePoints — indicates price in points;
- step_price — indicates cost of minimum price step
- min_step — indicates minimum price step in points.

Two more fields are required to fill when it comes about future contracts quoted in USR:

- Cost of price step in initial currency, i.e. in USD;
- Cost of price step in Russian roubles, which is fixed upon the main clearing session opening.

Each trading instrument, upon introduction into the system, becomes available for trading starting from its trading start date (instrument parameter); the instrument is tradable from the first available trading period on the specified calendar date. For information about instruments availability for trading in the additional/main trading session please refer to the value in the field ‘trade_period_access’ of table 'fut_sess_contents'.

Futures contract data ara stored in following tables of trade interface:

- 'FORTS_REFDATA_REPL' stream, 'fut_sess_contents' table. This is the main table, which contains a list of futures contract available on the current trading day;
- 'FORTS_REFDATA_REPL' stream, 'fut_instruments' table. The table contains limited data amount about all future contracts put into the system, including non-tradable contracts.
- 'FORTS_INFO_REPL' stream, 'futures_params' table. This table contains data about futures contracts. According to the data format the table can be loaded by the SpectraIM client application for calculating risks.

2.2.2.1. Daily futures contract with automatic prolongation

There are futures in the Spectra TS, which differ somewhat from the common ones in their behavior - these are one-day futures contract with automatic prolongation. CFD (Contract For Difference), as well as NDF (Non-Deliverable Forward) may be as an analogue of such contracts on the market.

The underlying assets of daily futures are foreign currency rates against the Russian ruble, indices calculated by Moscow Exchange, commodities (precious metals) and shares of Russian issuers.

The main features of the such instruments:

- Daily futures have no exercise date. Technically, the last trading date will be set far in the future.
- Settlement prices are formed on the basis of market data from the FX and Securities Markets of the Moscow Exchange
- The variation margin during the clearing session is calculated taking into account an additional component — funding. Funding is a charge for the deviation of perpetual futures prices from the spot price exceeding the level specified in the contract. Its purpose is to align the prices of perpetual contracts with the underlying assets. The funding value is published in the gateway in the 'fut_sess_settl' table of the 'FORTS_CLR_REPL' stream in the 'swap_rate' field.
- For the indicative variation margin, funding is calculated separately and published in the gateway in the 'common' table of the 'FORTS_COMMON_REPL' stream in the 'swap_rate' field.

- For perpetual futures on the index or the stock-equities, the evening clearing variation margin is calculated with the dividend amendment, which compensates for the change in the value of the index or stock-equities prices due to dividend ex-dates. The amendment value is published in the gateway in the 'fut_sess_settl' table of the 'FORTS_CLR_REPL' stream in the 'index_div' field.

- The indicative funding value and dividend adjustment for the current client position are calculated and published in the gateway in the indicative variation margin stream for the position 'FORTS_VM_REPL'.'fut_vm'/'fut_vm_sa' in the 'swap_rate' and 'index_div' fields, respectively.

Formulas and detailed description can be found in contract specifications.

In the directory of trade instruments, daily futures are marked with a special sign: bit 0x4000 (CFD) in ‘signs’ field of ‘fut_sess_contents’ table of FORTS_REFDATA_REPL stream.

2.2.3. Options

Options are another type of derivative financial instruments in the Spectra system. Unlike futures, an option does not mean an obligation, but the right to buy (sell) the underlying asset, which can be realized or not.

Distinguish between Call and Put options. A Call option (purchase option) is a contract that gives the holder (buyer) of the option the right to buy the underlying asset at a specified time in the future at a fixed price - the option exercise price (strike). A Put option (option to sell) is a contract that gives the holder (buyer) of the option the right to sell the underlying asset at a specified time in the future at a fixed price - the option exercise price (strike). Assets of the Securities and FX markets (shares, indexes, currencies, etc.) and futures contracts can serve as the underlying asset for options.

Options can be of the futures-style, with the payment of a variation margin between trading participants based on the settlement price determined at a fixed time specified in the schedule (19:00), and of the equity-style, with the payment of a premium to the option subscriber at the time of the performing of the trade (in the Clearing session).

According to the method of exercise, options are divided into European and American. European options can be exercised on the expiration date of the option only. American options differ in that the holder can exercise his right to sell/buy the underlying asset at any time without waiting for the option expiration date.

Options can be deliverable or cash-settled. A deliverable option involves physical delivery. The cash-settled option does not involve delivery, but provides for the recalculation of profit/loss between the participants in the form of accrual and write-off of funds.

The SPECTRA system currently supports American deliverable futures-style options on futures and European cash-settled equity-style options on spot assets (shares, indexes, currencies, etc.) and futures.

Options, like futures, have different exercise dates, but unlike futures, there are weekly options, with exercising in the middle of the nearest week and Friday evening clearing (equity options of US companies).

For options for trading, a certain subset of strikes is assigned, which lies in the vicinity of the current settlement price of the underlying asset to which the option is linked, therefore, the list of options assigned for trading, in general, can be different every day.

Options data are stored in the following tables:

- 'FORTS_REFDATA_REPL' stream, 'opt_sess_contents' table. This is the main table, which contains a list of contracts available on the current trading day.

- 'FORTS_REFDATA_REPL' stream, 'opt_vcb' table. This table contains a directory of underlying contracts for options.

- 'FORTS_REFDATA_REPL' stream, 'sess_option_series' table. This table contains a list of option series.

- 'FORTS_INFO_REPL' stream, 'options_params' table. This table contains data about option contracts. According to the data format the table can be loaded by the SpectraIM client application.

- 'FORTS_INFO_REPL' stream, 'option_series_params' table. This table contains parameters of options series.

2.2.4. Multi-leg instruments

The SPECTRA system supports multi-leg trading instruments, i.e. the instruments consisting of more than one components. This allows to use a trading strategy, when a client gets additional positions on two or more instruments when trade is complete. The instruments available now are calendar spreads for futures.

The list of the multi-leg instruments available in the system can be obtained in the 'fut_sess_contents' table of the 'FORTS_REFDATA_REPL' stream, by looking at the 'multileg_type' field. If a value in the field is not equal 0, than the record describes a compound instrument.

To obtain the list of components of compound instruments you should use the 'multileg_dict' table of the 'FORTS_REFDATA_REPL' stream, where every multi-leg instrument has two or more entries describing components of such instrument (see pic. 1). The 'multileg_dict' table entries refer back to 'fut_sess_contents', because the components of these instruments present as common trading instruments. We indicate a special coefficient for every single part, which should be multiplied by the amount from initial order to acquire the amount of a compound part of the order. The sign of this coefficient indicates the direction of order of the component — a positive value means that the component will be in the same direction as in the order by a multi-leg instrument, while a negative value means the opposite direction.

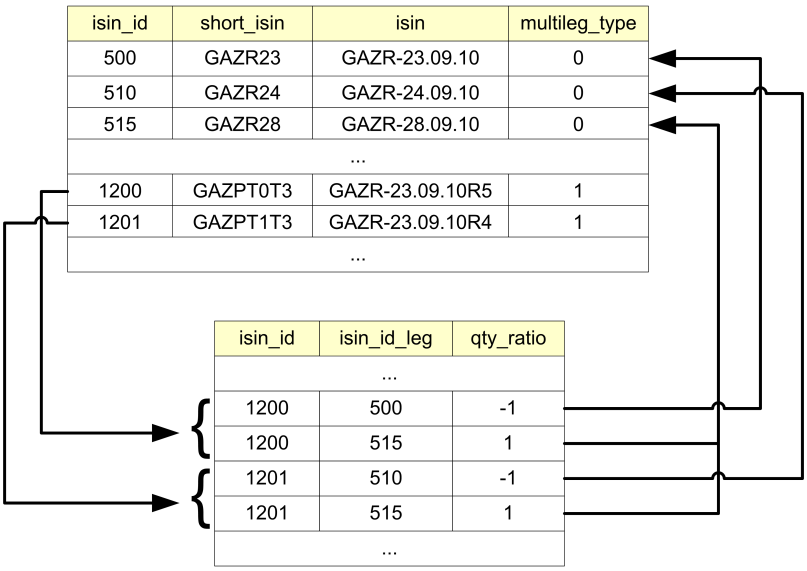

Figure 1. Multileg instruments

2.2.5. Identification of instruments

The SPECTRA system has four fields to identify each instrument:

1. 'isin_id' field, which contains the unique numeral code for each instrument.
2. 'isin' field, which contains the instrument's symbol code.
3. 'short_isin' field, which contains short symbol code for using in order books etc.
4. 'name' field, which contains a long 'humanized' instrument's description.

Example 3. Futures on RTS index value, to exercise in December 2010.

isin_id=

isin = RTS-12.10

short_isin = RIZ0

name = Futures contract on the RTS index value, to exersice on 15, December 2010.

A value in the 'isin_id' field is the primary unique instrument's code, which is used throughout of data structure of the system wherever a corresponded reference exists.

The 'isin' field contains the main symbol futures' code, which is used in order's instructions. The uniqueness and invariability in time of the 'isin' value is guaranteed.

The 'short_isin' field is an alternative symbol contract code. It has been implemented in order to ease access to the SPECTRA system data for news agencies.

2.2.6. Instruments with investor qualification restrictions

In accordance with the regulator's requirements, certain instruments may only be traded by qualified investors. In the Spectra System, the "Qualified Investor" attribute is implemented (bit 0x10000000000 in the xstatus field of the table investor of theFORTS_REFDATA_REPL) stream. This attribute is set by the broker via the client registration service. The change of the attribute is applied in the system during the Clearing session, both for existing and new (activation of template) clients.

To distinguish instruments with investor qualification restrictions, a special attribute "For Qualified Investors" is set. This attribute is set at the underlying futures contract level (bit 0x8 in the signs field of fut_vcb table of FORTS_REFDATA_REPL stream) and applies to all instruments under that underlying futures contract. When a client without the "Qualified Investor" attribute attempts to place an order for such an instrument, the system rejects the order with the error code "4227 - The qualified investor attribute is required to execute the trade".

Features of support for instruments with investor qualification restrictions:

- The transfer of a client's position (TransferClientPosition command) to a non-qualified investor is prohibited for instruments requiring the qualified investor attribute.
- In the case where it is necessary to remove the "Qualified Investor" attribute from a client and the client holds a position in a qualified instrument at that time, the following procedure is followed:
  - The broker prohibits opening new positions for the instrument

- The broker notifies the client and allows time for them to independently close the position
- If necessary, the broker submits closing orders on behalf of the client using their own login
- The broker removes the "Qualified Investor" attribute from the client (applied in the Clearing session)

2.3. Trading operations

2.3.1. Orders — general information

Order — a command, which is sent into the trading system by a trading participant, aimed to perform an action of buying or selling an instrument at specified price. There are two main types of orders available: negotiated and system.

System orders — a common type of orders available for all users of the system. System orders have to participate in auction along with offsetting orders. If there is an offsetting order available for any system order at a better or equal price, the order itself has to be exercised at the price equal to that of the offsetting order. The unexercised part of the order remains in the system as an order with less amount of instruments.

Orders can be subdivided into three types: day, immediate-or-cancel, fill-or-kill and book-or-cancel orders. A day order remains in queue after it has been fully or partly exercised. An immediate-or-cancel orders have to be removed from the system after auction ended, no matter whether it has been exercised fully or partly. At last, the fill-or-kill orders — the immediate-or-cancel orders which can only be exercised fully. Book-or-Cancel (BOC) - this is a kind of quote order, which either completely gets into the order-book, or is rejected by the system. Such an order can only be a passive side in the trade (Maker).

All orders can be also subdivided into common and multy-day orders, in accordance with their lifetime. Common orders do not have the date of expiration specified; such orders remain in queue until the end of the current trading day. Contrary, the expiration date for multi-day orders is specified, ranged from 1 day up to one year. Such orders are relisted automatically at opening of the next trading day; additionally each order receives a new ID and a link to the initial order's ID. When relisting, the orders are checked for having sufficient instrument, client and funds. Orders which are out-of-date are automatically removed after the trading day ends.

There are two additional fields added to meet the developers' needs:

- 'comment field' - a 20-symbol string; the field must not contain control characters (0x00-0x1F and 0x7F) and must be converted from cp1251 to utf-8.
- 'ext_id field' - a 4-byte number to store order's ID in the client application.

Note: The SPECTRA system does not check values of the additional fields for being unique.

Orders data are stored in the 'orders_log' tables of the 'FORTS_TRADE_REPL' and 'FORTS_ORDLOG_REPL' streams. The tables contain orders changing log, where every change is recorded as a separate record in the table. The table 'orders_log' of the stream 'FORTS_TRADE_REPL' contains information on the 'own' orders only. The 'own' orders are:

- For a client login - records about all orders, placed on behalf of this client;
- For brokerage firm or clearing firm login - records about all orders placed on behalf of clients of these firms.

Users can view all data on the 'own' orders, including data in service fields and user fields.

Clients are able to be subscribed for receiving the table 'orders_log' of the stream 'FORTS_ORDLOG_REPL; in this case, they will receive complete history of changes for all orders in the trading system in anonymous mode.

Users can do the following:

- Add an order;
- Delete an order according to its code in the SPECTRA system;
- Move an order (the 'MoveOrder' command). Moving of an order is implemented in two steps: deleting an 'old' order and adding a new one into the system (with a new code, which is sent to user after the order was added). Thus, at least two records (about deleting an order and adding a new one) will be added in the 'orders_log' table. You can move two orders at time by adding parameters ('order_id1', 'order_id2') to the 'MoveOrder' command, which can be useful for market-makers' needs. If you move only one order, then you should specify the 'order_id1' parameter only.
- Delete orders by mask. The following masks can be applied:
  - Direction of operation: buying or selling;
  - Order type: negotiated order or system order;
  - Client's code;
  - Underlying asset's code;
  - Order's ID in the client system ('ext_id');
  - Instrument's code;

- Instrument's group: futures, options, multi-leg instruments.

2.3.2. Negotiated orders

An order addressed to a certain client are called negotiated order. Unlike system orders, negotiated orders have some limitations for users in managing orders and selecting counterparts, namely the following:

- Negotiated orders may be added only by a BF's login, with the Brokerage Firm as the only allowed counterparty.
- For specifying a counterpart, the counterpart's RTS code is used in orders in 'broker_to' field. The brokerage firms which do not have the RTS code act as counterparts for negotiated orders.
- Instead of moving, negotiated orders can only be deleted and listed anew manually.
- Negotiated orders cannot indicate the order expiration date, i.e. they cannot be multi-day (GTD).
- Negotiated orders can only be exercised when the value of the 'match_ref' field of one order exactly matches the value of another (description of the field see Method AddOrder - Adding orders). Negotiated orders can also be exercised partly.

2.3.2.1. Negotiated mode with matching by an unique code

In this mode participants add negotiated orders with the 'match_ref' field filled in as mandatory and the NCC code (company code with flag dealer.status 0x10000 – NCC) indicated in the 'broker_to' field. In the 'match_ref' field a unique sequence of symbols is set, which the sides separately (outside the trading system) agree to use in their orders. Orders are matched by the 'match_ref' field value. Counterparties confirm their consent for the trade settlement on terms specified in the trade by using matching code value of the 'match_ref' field. In the "Negotiated mode with matching by a unique code" the counterparty is hidden at all stages of concluding and processing of a trade (order, matching, trade, reports).

And for the rest negotiated orders in this trading mode have the same restrictions as regular negotiated orders. To differentiate such orders from regular negotiated orders, the 'NegotiatedMatchByRef (0x80000000)' flag is set additionally to the 'Address (0x4000000)' flag.

2.3.3. Trades

Within SPECTRA trading system, a trade will be performed if an instrument price in one order meets the instrument price in an opposite order, i.e. sell or buy one for the same instrument. The price of the order settled first is the price of the trade. There are two types of trades: negotiated and system. Many trade's attributes are equivalent of that of the orders. Trades cannot be edited, or deleted from the system.

Data on own trades are stored in tables 'user_deal' and 'user_multileg_deal' of stream 'FORTS_TRADE_REPL'. The data on all trades in the system are distributed in accordance with the following rules: a user gets access only to their own part of the trade (buyer's or seller's). If a user acts as a brokerage firm, or a clearing firm, and both buyer and seller parts are the clients of the same firm, the user gets access to the data concerning both parts of the trade. Data on all trades are available for all users in tables 'deal' and 'multileg_deal' of stream 'FORTS_DEALS_REPL'. All data in tables are anonymised.

Along with the records containing the trades-related data, there are some additional records stored in tables containing trades data. These records cannot be classified as trades legally, but still they render some transactions within the system, which influence the participant's status. These trades are so called 'technical trades'. One can tell trading trades from the technical ones by values in fields 'xstatus_sell' and 'xstatus_buy' in tables 'user_deal' and 'user_multileg_deal' of stream 'FORTS_TRADE_REPL', or by the flag nosystem in table 'deal' and 'multileg_deal' of stream 'FORTS_DEALS_REPL' (for details see Flags applied to orders and trades).

2.3.4. Cross-trades

Cross-trade is a trade based on orders submitted from the same client account, or submitted from different accounts with the same TIN. By default, cross-trades are prohibited and rejected by the system. Cross-trades are allowed on an individual basis for negotiated orders. The system also provides the ability to cancel a passive order, rather than an active one, when a cross-transaction occurs.

The system provides two flags to determine the logic of processing cross-trades:

- 'Allow cross-trades' - allows cross-trades for negotiated orders. This flag must be set on both sides of the trade for a successful cross-trade. In anonymous mode, cross-trading is always prohibited.
- 'Cancel a passive order in a cross-trade' - allows to cancel a passive order instead of an active one for a cross-trade. A passive order cancellation implies the continuation of the active order matching within the transaction. If the transaction cannot be completed successfully during the matching process, then the canceled passive orders are recovered.

These flags are set at the level of the seven-digit client account based on the trading member's application. The set flags are broadcast in the gateway in the 'xstatus' field of the 'investor' table of the FORTS_REFDATA_REPL stream.

The table below shows scenarios for the behavior of anonymous orders leading to cross-trades for various types of orders and the values of the 'Cancel a passive order in a cross-trade' flag:

Active order type	The flag in the client account - the author of the passive order	Scenarios
Quote order (Day)	'Cancel a passive order in a cross-trade'=**NO**	**Passive**: Remains in the orderbook. **Active**: In the matching process, it comes to an order with the same TIN, then rolls back for all previously matched orders.

Active order type	The flag in the client account - the author of the passive order	Scenarios
	'Cancel a passive order in a cross-trade'=**YES**	**Passive**: Canceled. **Active**: In the matching process, it comes to an passive order with the same TIN, initiates the cancellation of a passive one, continues of matching further in the orderbook or remains as a quote order in the orderbook. If further in the transaction the matching of the active order cannot be completed successfully (for example, due to a meeting with another cross-order for which there is no permission to cancel it), then all orders, including the cross-passive one, will be recovered.
IOC order	'Cancel a passive order in a cross-trade'=**NO**	**Passive**: Remains in the orderbook. **Active**: In the matching process, it reaches an order with the same TIN, then the remainder is removed.
	'Cancel a passive order in a cross-trade'=**YES**	**Passive**: It is cancelled only if the active order can matches at least once with the next in queue after the cross-order. **Active**: Within one transaction, it reaches the passive one with the same TIN, initiates a preliminary cancellation of the passive order, then continues matching with the next orders in the orderbook queue. If after the passive order cancel, within this transaction, no matching has occurred, then the remainder of the active IOC-order is removed, and the passive order is recovered.
Fill-or-Kill (FOK) order	'Cancel a passive order in a cross-trade'=**NO**	**Passive**: Remains in the orderbook. **Active**: In the matching process, it reaches an order with the same TIN, then rolls back on all pre-matched orders.
	'Cancel a passive order in a cross-trade'=**YES**	**Passive**: It is cancelled only if the active order can be matched completely. **Active**: Within one transaction, it reaches the passive one with the same TIN, initiates a preliminary cancellation of the passive order, continues matching in the orderbook. If the active order cannot be completely matched, then all previously matched and canceled passive orders are recovered back to the orderbook.

The logic of processing situations with cross-trades is similar for negotiated orders. If cross-trades are not allowed (none of the counterparties has the 'Allow cross-trades' flag), then, depending on the value of the 'Cancel a passive order in a cross-trade' flag of the counterparty - the author of the passive order (NO/YES), the active order is rejected or the passive one is cancelled.

In the case of synthetic matching, two orderbooks with orders are analyzed independently – that is, both one and two passive orders can be cancelled. If cancellation is not allowed for both passive orders ('Cancel a passive order in a cross-trade'=**NO**), then the active order is rejected, otherwise the passive orders for which cancellation is allowed are cancelled (independently).('Cancel a passive order in a cross-trade'=**YES**).

The cancelled passive cross-orders have a distinctive signe 'DueToCrossCancel' (0x2000) in the 'status' field.

2.3.5. Specifics of trading multi-leg instruments

The SPECTRA system supports multi-leg trading instruments, i.e. the instruments consisting of more than one components. This allows to use a trading strategy, when a client gets additional positions on two or more instruments when trade is complete. The instruments available now are calendar spreads for futures.

The main specifics of multi-leg instruments trading:

- Prices in OrderBook can be ranked in two directions: straight or reverse.
- When listing the multi-leg order, a client is obliged to buy or sell two or more components. Therefore, calculation of collateral for such positions should be made in the appropriate way.
- Multi-leg orders cannot be moved.

2.3.6. Iceberg orders

An Iceberg order is a variation of a quote order. It allows you to hide a part of its volume from the market (that is, in the Order-book window) to minimize the impact on the large orders market price. Iceberg orders appear in the order-book in portions. The next portion "pops up" only after the visible part of the order will be executed. This process can be repeated until the whole hidden part is used.

The iceberg orders main features:

- Iceberg order can be on-exchange only. In terms of lifetime, iceberg orders can be ordinary and multi-day. According to the types of iceberg orders, they are divided into quote (Day) and Book-or-Cancel orders.
- When adding an iceberg order, it additionally indicates the parameters for calculating the size of the pop-up (visible) part. The pop-up part consists of a constant part ('disclose_const_amount') and a randomly calculated addition. The addition value is a random variable with a uniform distribution from the range [-Round(disclose_const_amount * variance_amount/100, 0); Round(disclose_const_amount *

variance_amount/100, 0)], where 'variance_amount' - variance amplitude from the constant part volume .Accordingly, when adding an iceberg order, two parameters are indicated in it:

- 'disclose_const_amount' - volume of the constant popup part. This parameter cannot be larger than the entire iceberg. A disclose volume cannot be less than the minimum lot for this instrument (values are published on the exchange website).
- 'variance_amount' - the value of the random deviation of the visible iceberg part (optional). The parameter value can be from zero to the value published on the exchange website. By default, the parameter is not set.

All specified parameters can take only integer positive values.

- Collateral is blocked for the full volume of the iceberg, not for the visible part only.
- When changing an iceberg order, only the price can be changed, the volume is not available for change.
- When deleting or changing an iceberg order, the entire iceberg is deleted or changed, including the visible part.
- In the tables of your orders and trades, iceberg orders and trades are marked with a special attribute in the fields
- In the tables of your orders and trades, iceberg orders and trades are marked with a special sign "Iceberg" (0x800000000000) in the 'xstatus' and 'xstatus_sell / xstatus_buy' fields.

2.3.6.1. Iceberg orders in the system information streams

An iceberg order consists of two parts: public - this is the visible part of the iceberg order, and private - the entire iceberg order, including the visible part. Accordingly, there are two sets of fields in the tables of your orders and trades (order ID, quantity in operation, rest, action, etc.):

1. Public data are broadcast in the fields with the prefix "public_":
   - Tabs 'orders_log' and 'multileg_orders_log':
     - public_order_id - ID of the visible part of the iceberg order.
     - public_amount - The number of contracts in the operation for the visible part of the iceberg order.
     - public_amount_rest - The remaining number of contracts in the visible part of the iceberg order.
     - public_action - Type of operation with the visible part of the iceberg order.
   - Tabs 'user_deal' and 'user_multileg_deal':
     - public_order_id_buy - ID of the visible part of the buyer's iceberg order.
     - public_order_id_sell - ID of the visible part of the seller's iceberg order.
2. Private data are broadcast in the fields with the prefix "private_":
   - Tabs 'orders_log' and 'multileg_orders_log':
     - private_order_id - ID of the entire iceberg order.
     - private_amount - The number of contracts in the operation for the entire iceberg order.
     - private_amount_rest - The remaining number of contracts in the entire iceberg order.
     - private_action - Type of operation with the entire iceberg order.
   - Tabs 'user_deal' and 'user_multileg_deal':
     - private_order_id_buy - ID of the entire buyer's iceberg order.
     - private_order_id_sell - ID of the entire seller's iceberg order.

Below is an example of entry in the stream for adding and matching of iceberg order with amount=1000 and visible amount=100 (without filtering):

public_ order_id	public_ amount	public_ amount_ rest	public_ action	price	moment	dir	client_ code	private_ order_id	private_ amount	private_ amount_ rest	pri- vate_ action	*comment*
101	**100**	**100**	**1**	**312**	**2019-01-11 11:55:58**	**1**	**OD01123**	**101**	**1000**	**1000**	**1**	*Add Iceberg*
102	1	1	1	312	2019-01-11 14:56:58	1	PJ99888	102	1	1	1	*Add stan- dard Order*
103	250	250	1	310	2019-01-11 16:58:58	2	FS01020	103	250	250	1	*Add stan- dard Order*
101	**100**	**0**	**2**	**312**	**2019-01-11 16:58:58**	**1**	**OD01123**	**101**	**100**	**900**	**2**	*Match Iceberg*

public_ order_id	public_ amount	public_ amount_ rest	public_ action	price	moment	dir	client_ code	private_ order_id	private_ amount	private_ amount_ rest	pri- vate_ action	*comment*
103	100	150	2	310	2019-01-11 16:58:58	2	FS01020	103	100	150	2	*Match stan- dard Order*
102	1	0	2	312	2019-01-11 16:58:58	1	PJ99888	102	1	0	2	*Match stan- dard Order*
103	1	149	2	310	2019-01-11 16:58:58	2	FS01020	103	1	149	2	*Match stan- dard Order*
104	**100**	**100**	**1**	**312**	**2019-01-11 16:58:58**	**1**	**OD01123**	**101**	**100**	**900**	**3**	*Pop-up Iceberg*
104	**100**	**0**	**2**	**312**	**2019-01-11 16:58:58**	**1**	**OD01123**	**101**	**100**	**800**	**2**	*Match Iceberg*
103	100	49	2	310	2019-01-11 16:58:58	2	FS01020	103	100	49	2	*Match stan- dard Order*
105	**100**	**100**	**1**	**312**	**2019-01-11 16:58:58**	**1**	**OD01123**	**101**	**100**	**800**	**3**	*Pop-up Iceberg*
105	**49**	**51**	**2**	**312**	**2019-01-11 16:58:58**	**1**	**OD01123**	**101**	**49**	**751**	**2**	*Match Iceberg*
103	49	0	2	310	2019-01-11 16:58:58	2	FS01020	103	49	0	2	*Match stan- dard Order*
105	**51**	**0**	**0**	**312**	**2019-01-11 17:00:58**	**1**	**OD01123**	**101**	**751**	**0**	**0**	*Cancel Iceberg*

Explanations for the table:

- Client OD01123 adds iceberg order with entire amount 1000 and visible part amount 100. New order with 'private_order_id'=101 (order ID), 'private_amount'=1000 (entire iceberg amount) and 'public_amount'=100 (visible part) is added to the system ('private_action'=1).
- Clients PJ99888 and FS01020 consistently add their standard orders to the system. Moreover, the order of client FS01020 is an opposite order that satisfies the price of two previous orders.
- Visible part of iceberg order and opposite order of client FS01020 are matched ('private_action'=2), size of the remaining iceberg 'private_amount_rest'=900.
- Then the standard orders of clients PJ99888 and FS01020 are matched.
- The next portion of the iceberg order pops up ('private_action'=3), which immediately is matched ('private_action'=2) with the remaining part of the order of client FS01020, size of the remaining iceberg 'private_amount_rest'=800.
- The next portion of the iceberg order pops up ('private_action'=3) and is matched with the remaining part of the order of client FS01020, size of the remaining iceberg 'private_amount_rest'=751.
- Then client OD01123 canceled iceberg order.
- Please note that when the next portion of the iceberg order pops up, it's visible part has number ('public_order_id') different from the identifier of the iceberg order itself ('private_order_id')!

For standard orders, private and public fields are filled with the same values and contain the usual values for ID, quantity in operation, remaining quantity and order operation code. Explanations for the filling of the fields from the example above:

public_ order_id	public_ amount	public_ amount_ rest	public_ action	id_ord	xamount	xam- ount_ rest	action	pri- vate_ order_id	pri- vate_ amount	private_ amount_ rest	pri- vate_ action	*comment*
101	100	100	1	101	1000	1000	1	101	1000	1000	1	*Add Iceberg*
102	**1**	**1**	**1**	**102**	**1**	**1**	**1**	**102**	**1**	**1**	**1**	*Add Order*
103	**250**	**250**	**1**	**103**	**250**	**250**	**1**	**103**	**250**	**250**	**1**	*Add Order*
101	100	0	2	101	100	900	2	101	100	900	2	*Match Iceberg*
103	**100**	**150**	**2**	**103**	**100**	**150**	**2**	**103**	**100**	**150**	**2**	*Match Order*
102	**1**	**0**	**2**	**102**	**1**	**0**	**2**	**102**	**1**	**0**	**2**	*Match Order*
103	**1**	**149**	**2**	**103**	**1**	**149**	**2**	**103**	**1**	**149**	**2**	*Match Order*
104	100	100	1	101	100	900	3	101	100	900	3	*Pop-up Iceberg*
104	100	0	2	101	100	800	2	101	100	800	2	*Match Iceberg*
103	**100**	**49**	**2**	**103**	**100**	**49**	**2**	**103**	**100**	**49**	**2**	*Match Order*

public_ order_id	public_ amount	public_ amount_ rest	public_ action	id_ord	xamount	xam- ount_ rest	action	pri- vate_ order_id	pri- vate_ amount	private_ amount_ rest	pri- vate_ action	*comment*
105	100	100	1	101	100	800	3	101	100	800	3	*Pop-up Iceberg*
105	49	51	2	101	49	751	2	101	49	751	2	*Match Iceberg*
103	**49**	**0**	**2**	**103**	**49**	**0**	**2**	**103**	**49**	**0**	**2**	*Match Order*
105	51	0	0	101	751	0	0	101	751	0	0	*Cancel Iceberg*

Anonymous streams of orders and trades contain only public fields, in which there is always the visible part of icebergs only.

2.3.6.2. Iceberg order operations

The following operations are possible for iceberg orders.

- Add order (command IcebergAddOrder).
- Delete order (command IcebergDelOrder). The command can work both on 'public_order_id' and on 'private_order_id'.
- Move order (command IcebergMoveOrder). The command can work both on 'public_order_id' and on 'private_order_id'.

Please note, that the 'IcebergMoveOrder' and 'cebergDelOrder' commands will work on 'public_order_id' only if the visible part with such a number is still in the system (has not been matched), otherwise an error will be returned about the absence of an order with such a number. Therefore, we recommend working with iceberg orders on 'private_order_id'.

2.3.6.3. Change of order ID during iceberg orders operations

When an iceberg order is added, its ID of the visible part ('public_order_id') is the same as ID of the entire iceberg order ('private_order_id'). When a new part pops up, a new ID ('public_order_id') is assigned to it, the ID of the entire iceberg order does not change. When an iceberg order is changed (move), a new 'private_order_id' is set for it.

When multi-day (GTD) iceberg orders are replaced in the clearing session, a new iceberg order with a new 'private_order_id' is set. This new order has 'the private_order_id' of the first iceberg order as the initial order (field 'id_ord1').

An example of changing order IDs during an iceberg order operations:

Operation	public_order_id	public_action	private_order_id	private_action	id_ord1
Add	100	1	100	1	
Matching into a trade	100	2	100	2	
New part pops up	105	1	100	3	
Replace in the clearing session	1106	1	1106	1	100
Move	1106	0	1106	0	100
	1200	1	1200	1	

Explanations for the table:

- **Add** - iceberg order is added ('private_action'=1) with ID 'private_order_id'=100 and ID of visible part 'public_order_id'=100.
- **Matching into a trade** - matching the visible part of the iceberg order ('private_action'=2) with the counter direction order.
- **New part pops up** - when a new part pops up ('private_action'=3), a new ID 'public_order_id'=105 is assigned to it, the ID of the entire iceberg order does not change.
- **Replace in the clearing session** - a new iceberg order ('private_action'=1) with a new 'private_order_id'=1106 is set in the clearing session,. This new order has 'the private_order_id' of the first iceberg order as the initial order (field 'id_ord1'=100).
- **Move** - old iceberg order is deleted (private_action=0) and new order with new 'private_order_id'=1200 is added (private_action=1).

Values of 'public_action':

- 0 - Order cancelled
- 1 - Order added
- 2 - Order exercised in a trade

Values of 'private_action':

- 0 - Order cancelled
- 1 - Order added
- 2 - Order exercised in a trade
- 3 - New visible part pops up

2.4. Delivery of assets and exercise of options

2.4.1. Deliveries on futures

There are three types of futures exist in terms of deliveries:

- Non-deliverable futures: upon expiration, only the amount of difference between the contract price and the current price of the asset will be delivered. The delivery is performed as technical closing of the position, and is marked with a special sign in the 'xstatus_sell' and 'xstatus_buy' fields of the appropriate table containing trades data (for details see Flags applied to orders and trades).
- Commodity futures: upon expiration, the assets and money are delivered. The delivery is performed as technical closing of the position, and is marked with a special sign in the 'xstatus_sell' and 'xstatus_buy' fields of the appropriate table containing trades data.
- Stock futures: upon settlement, the position for futures turns into a position on the T+ market (Moscow Exchange Main Market). The settlement is processed as a technical position closing trade on the derivatives market (the trade is marked with special flag in the 'xstatus_sell' and 'xstatus_buy' fields of the appropriate table containing trades data), and position opening trade on T+ market (added into the ASTS system of derivatives market). For more information see the section below.

2.4.1.1. Settlement of futures contracts of derivatives market for stock market (T+1 mode)

All deliverable futures contracts are settled via the automatic matching procedure for T+1 trades in the Main market section of Moscow Exchange (ASTS trading and clearing system).

In the SPECTRA clearing system, each Brokerage firm in order to make settlement is obtained with the firm code along with the trading-and-clearing account (TCA), both registered in the Trading and clearing system of the securities market. These two entities are used to perform the T+1 trades in order to fulfill obligations for the futures contracts. The client's account of the positions account register may have a separate TCA and client's code registered in the ASTS Securities Market.

Three trading-clearing accounts of different types (so called favorite TCAs, where one favorite TCA is used one for accruing own obligations, another one is used for accruing clients' obligations, and the third one is intended for Trust Management obligations) must be signed up within SPECTRA in advance in accordance with the Clearing Participant's application. Each of the three favorite TCAs will be assigned to the appropriate Brokerage Firm set by default.

The T+1 trades are matched on the ASTS Securities Market in a separate trading mode (SPEQ), with the settlement code Y1. The trades are matched between the National Clearing Centre and Securities Market Participants, with no additional confirmation from the Securities Market Participants.

If a T+1 trade cannot be performed due to the absence/incorrect details of the assigned Brokerage Firm/TCE, T+1 trades will be performed using a favorite TCE of the appropriate type (own, client, Trust Management). If the T+1 trades cannot be performed using a favorite TCA due to its absence/incorrect details, then the Clearing Participant's obligation on equity futures delivery will be considered as non-fulfilled for the given Brokerage Firm, and the appropriate fee in accordance with the amount of collateral on unfulfilled futures will be imposed.

After the settlement for securities has been fulfilled on the securities marked (in case of sufficient collateral amount), the futures position in SPECTRA system closes, and the collateral for this position releases. If the collateral amount is insufficient for the T+1 market position, then the futures position and its collateral remain blocked in the SPECTRA system until the margin request is executed on the T+1 market.

After the futures for securities are settled, the technical trades for closing futures positions appear in the trades table, marked with the 'Futures settlement trade' value in the 'xstatus_sell' and 'xstatus_buy' fields (for details see Flags applied to orders and trades). The technical trades for closing futures positions will also appear in the derivatives market reports 'f04.csv' and 'fut_deal.csv'.

For more information see http://moex.com/s1262.

Scenario for delivery execution of a futures contract

Instrument: Futures contract on ordinary shares of PJSC Aeroflot (ticker code AFLT-2.26, short code AFG6).

1. **Members and details**

   Derivatives Market (FORTS):

   - **Member code**: XX00000
   - **Brokerage firm code**: XX01000 (section type - "Client")
   - **Client section with open position**: XX01002

   Equity Market (T+ trading mode):

   - **Member firm code**: MC000XX00000
   - **Designated resources for delivery**:
     - **Depository account**: L01+00001F00 (type - "Client")
     - **Client code**: 76543

2. **Stage of forming the delivery position list (Last trading day of the futures)**

On the last trading day of the futures, during the clearing session, a list of delivery positions is generated.

Initial member position on the Derivatives Market:

- **Instrument**: AFLT-2.26 (AFG6)
- **Volume**: 4 contracts (100 shares per contract)
- **Price**: 6 000 RUB. per lot

Delivery trades (T+) are automatically executed without order submission on the trading day following the last trading day of the futures, during a special session. The futures execution day does not coincide with its last trading day.

3. **Delivery day (Execution day)**

   Delivery trades are formed based on the obligation list. If a member has correctly linked their Equity Market details and has sufficient funds on the Equity Market, trades are formed before the start of the main trading session. If the link is incorrect or funds are insufficient, trades are formed after corrections are made. The account linkage report toeqXXYY.csv is generated and distributed to members daily.

   On the Equity Market, T+1 trades are executed with the NCC.

   Status in the Equity Market gateway:

   In the trades table, the trade with the NCC is displayed:

   - **Instrument**: AFLT (Ordinary share of Aeroflot)
   - **Price**: 60.000000
   - **Quantity**: 400 ahares
   - **Volume**: 24 000 RUB
   - **Account**: L01+00001F00
   - **Trading board**: SPEQ (Delivery for FC (shares))
   - **Settlement code**: Y1

   In the directed orders table, two matching orders with the NCC are displayed:

   - **Instrument**: AFLT
   - **Trading board**: SPEQ
   - **Settlement code**: Y1
   - **Account**: L01+00001F00
   - **Price**: 60,000000
   - **Quantity**: 400 shares

   After trades are executed on the Equity Market, the Unified Limit is checked using the standard algorithm for T+ trades. If the Unified Limit on the Equity Market for the specified account (L01+00001F00 in this example) is non-negative, trades are executed with the NCC on the Derivatives Market.

   Status in the Derivatives Market gateway:

   In the trades table, the trade with the NCC is displayed:

   - **Instrument**: AFLT-2.26
   - **Price**: 60.000000
   - **Quantity**: 4 lots
   - **Seller code**: XX01002
   - **Indicator**: Non-system trade
   - **Status**: Fields xstatus_sell and xstatus_buy contain value "Futures execution trade"

   The position on the Derivatives Market is closed, and initial margin is released.

4. **Reporting**

   For the Equity Market member (MC000XX00000):

Delivery trade reports are generated

- SystemRef field contains the code of the Derivatives Market position registry section: XX01002
- For T+ trades executed as a result of deliverable futures execution, the BoardId field contains the trading board code Y1 (SPEQ board — "Delivery for FC (shares)")

For the Derivatives Market member (XX00000):

Trade report f04 is generated

- type = 11
- 'Signs' field contains the corresponding bitmask

2.4.2. Execution during the trading session (according to schedule) and at clearing

Contract expiration can occur during the trading session (with online position closure) and during the Clearing Session. The expiration method for a contract is defined at the level of the futures instrument and the Option Series (OS). Additionally, for instruments expiring during the trading session, an expiration time must be specified (this is set at the level of the Futures Base Contract (FBC) and the OS).

Contracts expiring during the trading session can be executed twice a day, at 14:00 and 19:00 Moscow Time. The following are eligible for execution during the trading session:

- Cash-settled futures contracts
- Margined options that expire on the same day as their underlying futures contracts
- Premium-paid options
- Margined options with a expiration date that does not match their futures contract

Execution within the Clearing Session framework:

- Deliverable futures contracts
- Prescheduled option exercise by the holder of an American-style option
- Exit from perpetual futures

2.4.2.1. Fixation of indicative rates for expiration

To calculate contracts quoted in currency, fixed values of indicative currency rates are required to determine the price step value (for clearing and for expiration). Indicative currency rates are fixed twice daily at 02:00 PM and 07:00 PM. Corresponding fixed values of the price step value for each contract will be reflected in the step_price_clr field in the fut_sess_contents and sess_option_series tables of the FORTS_REFDATA_REPL stream:

- for expiring instruments - at 02:00 PM and 07:00 PM;
- for other instruments - at the moment of rate fixation for the clearing session at 07:00 PM.

2.4.3. Option exercise

At present, the SPECTRA system supports American deliverable futures options and European cash-settled options on the spot assets (shares, indexes, currencies, etc.) and futures.

2.4.3.1. Exercise of American deliverable futures options

When exercising, the option position turns into a futures position with the price equal to strike of the exercising option contract. The exercise is processed during clearing session, and, technically, consists of closing of the option position and opening a futures position. Both of the positions are marked with a special flag in the fields 'xstatus_sell' and 'xstatus_buy' of the appropriate table containing trades data (for details see Flags applied to orders and trades).

There are two types of exercise available:

- Prescheduled exercise, processed according to a participant's request. A buyer is allowed, at any time, put the corresponding request into the system (for details see Method OptChangeExpiration - Request for the exercise of options). The requests are accepted during the whole trading day and executed in the evening during the clearing session.
- Automatic exercise, on the option expiration date. On the expiration date, each options that are 'in the money' (call whose strike is strictly less than the futures settlement price, and put, which strike is strictly more than the futures settlement price) exercises automatically.

  For 'at the money' option contracts (the call and put ones with their strike prices strictly equal to the appropriate futures settlement prices), automatic exercise is processed for the half of the open option position with the specified strike price. If the open position value is uneven, then rounding up (where 0.5=1) is applied for options call and rounding down is applied for options put (0.5=0) to calculate the settlement position value.

  Automatic exercise can be carried out both during trading (according to the schedule) and during the Clearing session (set at the level of the option series).

You can turn off the automatic exercise feature by adding a negative amount of option contracts into the 'OptChangeExpiration' request (field 'amount'). The amount of option contracts specified will not exercise automatically.

2.4.3.2. Exercise of European cash-settled options

For options that are European and cash-settled, only automatic exercise is provided. On the last day of circulation, only options that are 'in the money' are exercised. Accordingly, requests for exercise/refusal of exercise is not accepted by the system.

2.4.3.3. Option risk calculation before exercise

Current initial margin calculation algorithm can cause sudden collateral increase for clients. For more flexible management, new parameters allowing the broker to set IM calculation algorithm for clients, will be added into the trading system.

Expiration scenario parameters:

- **exp_clearings_bf** - this parameter is set by NCC globally which defines quantity of clearing sessions before expiration for series of options. During those clearing sessions IM calculated on expiration model"s basis will be blocked for Broker. Starting from (exp_clearings_bf/2) days before the expiration date volatility model will be used. This is applied during clearing session. Can differ for different underlying assets.
- **exp_clearings_sa** - number of clearing sessions before expiration with expiration scenario applied for Settlement Account. The setting is applied and changed by NCC for the whole market during the clearing session.
- **exp_weight** - weight of risk profile in accordance with expiration scenario.
  - **exp_weight (client)**: The setting may be applied by a Brokerage Firm by sending a non-trading transaction OptChangeRiskParametersNextSession for each client, will be applied during the nearest clearing session.
  - **exp_weight (broker)**: The setting may be applied by a Clearing Firm via the EDM system, by sending command 'ChangeBFParametersNextSession'. The setting 'exp_weight (broker)' will be used to calculate collateral value for a Brokerage Firm with the nett margining mode enabled.
  - **exp_weight_client_default**: The setting may be applied by a Clearing Firm via the EDM system by sending command 'ChangeBFClientDefaultParametersNextSession'. The setting 'exp_weight_client_default' will be applied for all BF's clients with non-specified setting ‘exp_weight (client)’, as the default setting.

  If the Broker does not set weights of risk-profile, for all his clients NCC default parameters will be applied.
- **exp_clearings_cc** - this parameter is set by NCC per all clearing participants and defines quantity of clearing sessions during which risk profile weight exp_weight might be applied for clients. Can be applied only after clearing session.
- **num_clr_2delivery (broker)** - the setting may be applied by a Brokerage Firm via sending a non-trading gateway transaction 'ChangeBFParametersNextSession'. The value stands for the number of clearing sessions before expiration, with risk profile weight applied to calculate collateral value for BF with the nett margining mode enabled. This setting overrides the NCC-applied setting ‘exp_clearings_bf’ if value ‘num_clr_2delivery (broker)’ is less than that of ‘exp_clearings_bf’.
- **num_clr_2delivery_client_default** - the setting may be applied by a Clearing Firm via the EDM system by sending command 'ChangeBFClientDefaultParametersNextSession'. The setting is applied for all BF’s clients with non-specified setting ‘num_clr_2delivery’, as the default setting.

2.4.4. Termination of obligations under daily futures contracts with auto-prolongation (‘perpetual’ futures)

Termination of obligations under perpetual futures can be carried out at the request of one of the side by submitting requests for exercise. A perpetual futures is exercised turning into the nearest standard future on the same spot asset. Requests for exercise of a perpetual futures can be submitted during the entire trading day ten trading days before the last day of the futures circulation on which the perpetual futures is exercised (schedule is published on the exchange website). The gateway command FuturesExecutionRequest is used for submit requests. Requests submitted to the trading system are transmitted in the gateway in the 'fut_exec_orders' table of the FORTS_REFDATA_REPL stream.

The exercising of the perpetual futures is carried out in the Clearing session of the day of the requests submission. The perpetual futures exercising consists of matching counter requests for exercising and, in case of unsatisfied requests for exercise and open positions with the opposite direction in the market, the procedure for their forced matching.

When exercising, a perpetual futures position becomes a common futures position. Technically, a position closing trade in a perpetual futures and a position opening trade in the nearest futures are formed, which are marked with special signs in the trades table in the 'xstatus_sell' and 'xstatus_buy' fields:

- PerpetualFuturesExecutionVoluntary (0x10000000000000) - the technical trade as a result of voluntary exiting a perpetual futures (based on the submitted requests);
- PerpetualFuturesExecutionForced (0x400000000000000) - the technical trade as a result of forced exiting a perpetual futures (realization of unsatisfied demand);
- PerpetualFuturesExecution (0x800000000000000) - the technical trade with linked instrument as a result of exiting a perpetual futures;

- PerpetualFuturesExecutionPartialVoluntary (0x2000000000000000) - the technical trade for exiting a perpetual futures with forced closure of one party's position.

A member who submits an order to execute a perpetual future contract, which is subsequently executed through the forced closure of the opposite position of another member who did not submit an order, must make a one-time payment to the member whose position was forcibly closed. The amount of this payment is determined according to the contract specifications. The purpose of this payment is to compensate the member whose position was forcibly closed for the execution of the opposing order to exit the perpetual future contract.

The one-time payment for the forced execution of perpetual futures is automatically paid during the clearing session when the exit orders are executed. The payment for the forced exit is transmitted through the gateway:

- In the FORTS_CLR_REPL stream, in the 'fut_pos table':
  - pos_enforcement_exec: Number of positions forcibly executed for exiting perpetual futures;
  - pay_enforcement_exec: Payment amount (with sign) for forced execution.
- In the FORTS_CLR_REPL stream, in the 'fut_sess_settl' table, in the 'field exp_enforcement_pay': Payment amount (in rubles) per contract for settlement of perpetual futures positions, in case of forced execution, from the party initiating the exit to the party forcibly closed out.

2.4.5. Trade Settlement TAS (*This service is not available in the current version*)

TAS (Trade at Settlement) service provides option to buy or sell futures contract on the forward settlement price of the current trading day plus a certain spread acceptable for the participant to pay additionally or bend on the settlement price of the contract. A forward settlement price of the current trading day is calculated during the clearing session. Counterparties should agree on the spread value and specify this value as the price of TAS order. In the trading system TAS is technically implemented as futures on futures: ordinary futures contract is based on the desired futures contract. Therefore, the desired futures contract serves as an underlying asset (UA-futures). The participant receives a TAS-futures position by making a trade based on the order with TAS spread.

Trades based on orders with TAS spread are settled during the clearing session. After the trade was settled, TAS-futures position is replaced by UA-futures position with a technical trade of position closing on the TAS-futures at a price of 0 and a trade of position opening on the UA-futures at the settlement price of the UA-futures. Trades in the trades table are marked with TASSettlement 0x10000 bit (Trade settlement based on an order indicating the TAS spread) in the 'xstatus_sell' and 'xstatus_buy' fields.

2.4.6. Flags applied to orders and trades

Flags applied to orders and trades:

Flag name	Bit mask	Description
Market orders		
Auction	0x1	Day order.
Opposite	0x2	Immediate-or-Cancel order (IOC).
FOK	0x80000	Fill-or-Kill order.
BOC	0x1000000000000000	Book-or-Cancel order.
Clearing trades		
NonQuote	0x4	Sign of an order/trade that is not included in the calculation of quotes. Applied to negotiated orders/trades, technical trades, clearing trades, multi-leg trades and RFS trades.
Exec	0x20	A trade resulting from the exercise of an option. The flag is set for option trades and for futures trades that appeared as a result of the exercise of options.
Expiration	0x80	Instrument, futures, or option expiration.
DUFlow	0x800	Technical trade of position transfer between trust managing BFs of Clearing Firms.
TASSettlement	0x10000	Trade Settlement TAS. (*Not available in the current version*)
OptionLapse	0x800000	Option expiration trade
ClearingTrade	0x2000000	Off-book clearing trade. Applied to all clearing trades.
FuturesExecution	0x40000000	Futures exercise trade.
CollateralInstrument	0x400000000	Collateral instrument trade.
PerpetualFuturesExecutionVoluntary	0x10000000000000	The technical trade executed as a result of exiting a perpetual futures (based on the submitted requests).

Flag name	Bit mask	Description
PerpetualFuturesExecutionForced	0x400000000000000	The technical trade executed as a result of forced exiting a perpetual futures (realization of unsatisfied demand).
PerpetualFuturesExecution	0x800000000000000	The technical trade with linked instrument executed as a result of exiting a perpetual futures.
PerpetualFuturesExecutionPartialVoluntary	0x2000000000000000	The technical trade for exiting a perpetual futures with forced closure of one party's position.
Negotiated orders/trades		
TransferClientPosition	0x8	Position transfer between BFs.
Address	0x4000000	Negotiated order, negotiated trade, indicative trade (RFS), or an order resulting from matched indicative quotes.
NegotiatedMatchByRef	0x80000000	Negotiated order or trade matched by reference.
TransferSource	0x200000000	Donor BF side in position transfer between BFs.
Multi-leg orders/trades		
REPOBack	0x4000	Far leg transaction.
Strategy	0x8000000	Multi-leg order/trade. Applied to all multi-leg transactions.
Other		
DontCheckMoney	0x10	Do not verify collateral for client accounts.
ExternalUseEveningExecution	0x100	A sign of an order or trade in the evening trading session.
DontCheckLimits	0x200	Do not verify limits for options.
Charge	0x400	End of logical transaction. The flag is set by the core on orders so that the number (and limits) of transactions can be obtained from the order log, that is, the sequence of orders generated by one transaction.
LastRec	0x1000	Sign of the last entry in the matching transaction.
DueToCrossCancel	0x2000	Sign of canceling a passive order in a cross-trade.
MoveOperation	0x100000	Sign of moving the order.
DeleteOperation	0x200000	Deleting one order.
BulkDeleteOperation	0x400000	Mass deletion of orders.
OppositeOrderTailDeleteDueToCrossTrade	0x20000000	Deleting the remainder of the Immediate-or-Cancel order due to a cross-trade.
CODBulkDeleteOperation	0x100000000	Sign of the operation for deleting the order by the 'Cancel On Disconnect' service.
FineOperation	0x1000000000	Applied on order cancellation due to an RFS fee.
UKSBulkDeleteOperation	0x2000000000	Orders are cancelled by UserKillSwitch
NCCRequest	0x4000000000	Operation resulting from request to NCC.
NCCBulkDeleteOperation	0x8000000000	Orders are cancelled by DelOrdersByBFLimit
LiqNettingRF	0x10000000000	Orders and trades formed in the process of liquidation netting
ActiveSide	0x20000000000	The active side in the trade. The order that led to the trade when added to the order-book.
PassiveSide	0x40000000000	The passive side in the trade. The order from the order-book involved in the trade.
Synthetic	0x200000000000	Sign of a synthetic order and the side of the trade corresponding to this synthetic order.

Flag name	Bit mask	Description
RFSOrder	0x400000000000	Order from the RFS system.
Iceberg	0x800000000000	Iceberg order/trade.
OperatorInputSA	0x1000000000000	A sign of an order or trade formed in SA blocking mode.
DontFineRF	0x80000000000000	No penalty for settlement transactions.
MorningSession	0x100000000000000	Sign of trade made during the morning trading session.
SyntheticPassive	0x200000000000000	Sign of a passive synthetic order.
DuringDiscreteAuction	0x4000000000000000	Sign of an order or trade in the opening auction.

In order to distinguish negotiated trades from indicative trades, it is recommended to check both 'Address' and 'RFSOrder' flag status. Please note that flag 'RFSOrder' is active for orders and trades resulting from matched indicative quotes (RFS), and inactive for negotiated trades.

The data in the PLAZA II gateways and reports are synchronized for providing convenience work of the back-offices. The 'signs_buy' and 'signs_sell' fields is used in the f04_XXYY.csv, f04clXXYYZZZ.csv, o04_XXYY.csv, o04clXXYYZZZ.csv reports. This fields is based on the bitmask of PLAZA II.

The 'ExternalUseEveningExecution' and 'MorningSession' bits are broadcast only in reports.

Since version 7.9, the following signs of orders and trades, are not currently used, will be redefined and used for other purposes:

- SpotTransfer (0x8000);
- REPO (0x20000);
- IQSOrder (0x800000000).

2.4.7. Trade types, created upon exercising and expiration of futures and options

Trade types, created upon execution and expiration of futures and options are listed in the table below:

Operation type	Position closing trade	Position opening trade	Date and time of trades availability in reports and in the gateway
Exercise of deliverable futures	• id in gateways and reports is unique and nonzero. • Trade price is rounded to minimal step price. • Technical trade is not a trade legally. • Signs in gateways and reports (bitmask): NonQuote, Expiration, FuturesExecution, ClearingTrade.	No.	On the execution day, in the morning.
Exercise of cash-settled futures	• id in gateways and reports is unique and nonzero. • Trade price is rounded to 5 places. • Technical trade is not a trade legally. • Signs in gateways and reports (bitmask): NonQuote, Expiration, FuturesExecution, ClearingTrade.	No.	On the execution day, in the evening.
Exercise of a deliverable option	• id in gateways and reports is unique and nonzero. • Trade price is 0. • Technical trade is not a trade legally. • Signs in gateways and reports (bitmask): NonQuote, Exec, ClearingTrade.	• id in gateways and reports is unique and nonzero. • Trade price is rounded to 5 places. • This trade is a trade legally. • Signs in gateways and reports (bitmask): NonQuote, Exec, ClearingTrade.	On the execution day, in the evening.
Exercise of a cash-settled option	• id in gateways and reports is unique and nonzero.	No.	On the execution day, in the evening.

Operation type	Position closing trade	Position opening trade	Date and time of trades availability in reports and in the gateway
	• Trade price is rounded to 5 places. • Technical trade is not a trade legally. • Signs in gateways and reports (bitmask): NonQuote, Exec, ClearingTrade.		
Expiration of option	• id in gateways and reports is unique and nonzero. • Trade price is 0. • Technical trade is not a trade legally. • Signs in gateways and reports (bitmask): NonQuote, Expiration, ClearingTrade, OptionLapse.	No.	On the futures execution day, in the evening.
Segregated Brokerage Firm position transfer	• id in gateways and reports is unique and nonzero. • Trade price is 0. • Technical trade is not a trade legally. • Signs in gateways and reports (bitmask): NonQuote, Address, TransferClientPosition, ClearingTrade.	• id in gateways and reports is unique and nonzero. • Trade price is 0. • Technical trade is not a trade legally. • Signs in gateways and reports (bitmask): NonQuote, Address, TransferClientPosition, TransferSource, ClearingTrade.	During the clearing session.
Voluntary exiting a perpetual futures (based on the submitted request)	• id in gateways and reports is unique and nonzero. • The trade price is equal to the settlement price of the perpetual futures. • Technical trade is not a trade legally. • Signs in gateways and reports (bitmask): NonQuote, PerpetualFuturesExecutionVoluntary, ClearingTrade, PerpetualFuturesExecutionPartialVoluntary.	• id in gateways and reports is unique and nonzero. • The transaction price is equal to the settlement price of the perpetual futures, multiplied by a coefficient that takes into account the difference in the lot volume of the instruments. • This trade is a trade legally. • Signs in gateways and reports (bitmask): NonQuote, PerpetualFuturesExecution, ClearingTrade.	In the evening on the day of the exercise request submission
Forced exiting a perpetual futures (realization of unsatisfied demand)	• id in gateways and reports is unique and nonzero. • The trade price is equal to the settlement price of the perpetual futures. • Technical trade is not a trade legally. • Signs in gateways and reports (bitmask): NonQuote, PerpetualFuturesExecutionForced, ClearingTrade, PerpetualFuturesExecutionPartialVoluntary.	• id in gateways and reports is unique and nonzero. • The transaction price is equal to the settlement price of the perpetual futures, multiplied by a coefficient that takes into account the difference in the lot volume of the instruments. • This trade is a trade legally. • Signs in gateways and reports (bitmask): NonQuote, PerpetualFuturesExecution, ClearingTrade.	In the evening on the day of the exercise request submission
Trade Settlement TAS (*Not available in the current version*)	• id in gateways and reports is unique and nonzero. • Trade price is 0. • Technical trade is not a trade legally. • Signs in gateways and reports (bitmask): NonQuote, ClearingTrade, TASSettlement.	• id in gateways and reports is unique and nonzero. • Trade price is equal to the settlement price of the underlying asset. • This trade is a trade legally. • Signs in gateways and reports (bitmask): NonQuote, ClearingTrade, TASSettlement.	On the execution day, in the evening.

Trades are shown as following:

Operation type	Operations info
Stock futures trade based on a negotiated order	• id in gateways and reports is unique and nonzero. • Trade price is rounded to the minimal price step. • This trade is a trade legally. • Signs in gateways and reports (bitmask): NonQuote, Address.
Stock futures trade based on a system order	• id in gateways and reports is unique and nonzero. • Trade price is rounded to the minimal price step. • This trade is a trade legally. • Signs in gateways and reports (bitmask):bits value is 0.
Stock futures option trade based on a negotiated order	• id in gateways and reports is unique and nonzero. • Trade price is rounded to the minimal price step. • This trade is a trade legally. • Signs in gateways and reports (bitmask): NonQuote, Address.
Stock futures option trade based on a system order	• id in gateways and reports is unique and nonzero. • Trade price is rounded to the minimal price step. • This trade is a trade legally. • Signs in gateways and reports (bitmask):bits value is 0.
Position transfer trade	• id in gateways and reports is unique and nonzero. • Trade price is rounded to 5 places. • This trade is not a trade legally. • Signs in gateways and reports (bitmask): NonQuote, Address, TransferClientPosition, TransferSource.
Technical trade based on the negotiated multi-leg order (near leg)	• id in gateways and reports is unique and nonzero. • Trade price is rounded to 5 places. • This trade is a trade legally. • Signs in gateways and reports (bitmask): NonQuote, Address, Strategy.
Technical trade based on the negotiated multi-leg order (far leg)	• id in gateways and reports is unique and nonzero. • Trade price is rounded to 5 places. • This trade is a trade legally. • Signs in gateways and reports (bitmask): NonQuote, Address, Strategy, REPOBack.
Technical trade based on the system multi-leg order (near leg)	• id in gateways and reports is unique and nonzero. • Trade price is rounded to 5 places. • This trade is a trade legally. • Signs in gateways and reports (bitmask): NonQuote, Strategy.
Technical trade based on the system multi-leg order (far leg)	• id in gateways and reports is unique and nonzero. • Trade price is rounded to 5 places. • This trade is a trade legally. • Signs in gateways and reports (bitmask): NonQuote, Strategy, REPOBack.
Equity futures trade based on indicative trade in RFS	• id in gateways and reports is unique and nonzero. • Trade price is rounded to the minimal price step.

Operation type	Operations info
	• This trade is a trade legally. • Signs in gateways and reports (bitmask): NonQuote, Address, RFSOrder.
Option on equity futures trade based on indicative trade in RFS	• id in gateways and reports is unique and nonzero. • Trade price is rounded to the minimal price step. • This trade is a trade legally. • Signs in gateways and reports (bitmask): NonQuote, Address, RFSOrder.
Technical trade for the near leg of a multi-leg order based on indicative trade in RFS	• id in gateways and reports is unique and nonzero. • Trade price is rounded to 5 places. • This trade is a trade legally. • Signs in gateways and reports (bitmask): NonQuote, Address, Strategy, RFSOrder.
Technical trade for the far leg of a multi-leg order based on indicative trade in RFS	• id in gateways and reports is unique and nonzero. • Trade price is rounded to 5 places. • This trade is a trade legally. • Signs in gateways and reports (bitmask): NonQuote, Address, Strategy, REPOBack, RFSOrder.

2.5. Trading and clearing schedule

2.5.1. Trading schedule. Trading days.

Trading in the SPECTRA system takes place within a Trading day. A Trading day is a period of time during which the Exchange conducts trading. A Trading day includes the following:

- Opening auction (if held) conducted before the Morning additional trading session, lasting from 06:50 AM to 07:00 AM, Moscow time zone.
- Morning additional trading session (if held), lasting from 07:00 AM to 09:00 AM, Moscow time zone.
- Opening auction (if held) conducted before the Main trading session, lasting from 08:50 AM to 09:00 AM, Moscow time zone.
- Main trading session, lasting from 09:00 AM to 07:00 PM, Moscow time zone.
- Evening additional trading session (if held), lasting from 07:00 PM to 11:50 PM, Moscow time zone.
- Opening auction (if held) conducted before the Additional weekend trading session, lasting from 09:50 AM to 10:00 AM, Moscow time zone.
- Additional weekend trading session (if held), lasting from 10:00 AM to 07:00 PM, Moscow time zone. There may be multiple Additional weekend trading sessions.

Within one Trading day, the same trading instruments are traded, and the same parameters for margin calculation are applied. Between Trading days, a number of critical operations for the SPECTRA system are performed, such as Clearing session, expiration of contracts, generation and distribution of reports, and more.

The schedule of Trading days is published in the gateway in the 'session' and 'trade_periods' tables in FORTS_REFDATA_REPL stream.

2.5.2. Settlement price and indicative rates determination

The settlement price is used for calculations in the clearing session. The settlement price determination is carried out according to the Methodology for determination of settlement prices for derivatives contracts, described in Appendix 1 to the Moscow Exchange derivatives trading rules. The settlement price is determined at a fixed time specified in the schedule, which is published in the gateway in the settl_price_calc_time field (19:00) of the 'session' table in the FORTS_REFDATA_REPL stream. The settlement price value, after its determination, is published in gateway in the settlement_price_clr field of 'fut_sess_contents' and 'opt_sess_contents' tables in the FORTS_REFDATA_REPL stream. The appearance of a new settlement price value in the tables can be tracked by the arrival of the synchronization event (47) settlpricecalc_finished in the sys_events table.

Before the settlement price determination point (settl_price_calc_time), the settlement_price_clr field will display the previous settlement price value. The logic of filling the settlement_price field remains unchanged - as currently, the field will be updated with the actual settlement price value during the clearing session.

2.5.2.1. Calculation of additional indicative margin and current operational risk values considering the settlement price

In addition to the published indicative variation margin values calculated based on the current market price, the gateway transmits the indicative variation margin value calculated using the settlement price fixed before clearing. The calculated value is published in the gateway in the vm_at_settl field in the fut_vm, opt_vm, fut_vm_sa, and opt_vm_sa tables in the FORTS_VM_REPL stream. Before the settlement price is fixed, the vm_at_settl calculation uses the current market price. After the settlement price is fixed, the vm_at_settl calculation will apply the settlement price.

In the FORTS_RMT_REPL stream, the rmt_im table also publishes additional values: the current operational risk calculated using the settlement price (rmt_posrisk_at_settl) and the indicative variation margin calculated using the settlement price (rmt_vm_at_settl). Before the settlement price is fixed, the current market price is used in the calculation of these values. After the settlement price is fixed, calculation will use the settlement price.

2.5.2.2. Fixation of indicative rates

To calculate contracts quoted in currency, fixed values of indicative currency rates are required to determine the price step value (for clearing and for expiration). Indicative currency rates are fixed twice daily at 02:00 PM and 07:00 PM. Corresponding fixed values of the price step value for each contract will be reflected in the step_price_clr field in the fut_sess_contents and sess_option_series tables of the FORTS_REFDATA_REPL:

- for expiring instruments - at 02:00 PM and 07:00 PM;
- for other instruments - at the moment of rate fixation for the clearing session at 07:00 PM.

2.5.3. Clearing Session (mtm)

The Clearing Session (mtm) is conducted after the Trading day T from 11:50 PM to 12:30 AM, Moscow time zone. The Clearing Session is necessary for calculating requirements and obligations under sections of 7кр-BF and Settlement Codes based on the results of the Trading day T. During the Clearing Session, the following actions are performed:

- Determination of requirements and obligations (variation margin, premiums, commissions) based on the results of Trading day T.
- Recalculation of price limits.
- Recalculation of initial margin.
- Expiration of contracts where applicable.
- Cancellation of orders for the current day, preparation for re-submission of Good-Till-Date orders.
- System date update and assignment of the next Trading day, including re-submission of orders.
- Generation of reports.

2.5.4. Settlement Session

The Settlement Session is necessary for executing the requirements and obligations calculated in the Clearing Session T under sections of 7кр-BF and Settlement Codes. The Settlement Session is conducted during Trading day T+1 at 08:00 PM Moscow time without interrupting trading. During the Settlement Session, the following actions are performed:

- Execution of obligations/requirements for variation margin and premiums determined in the previous Clearing Session under sections of 7кр-BF and Settlement Codes.
- Execution of obligations for fees based on trades concluded in the previous trading day.
- Fixation of liabilities of Clearing Members towards the NCC.
- Generation of reports on the execution results.

2.5.5. How different entities act on assigning a new Trading day

2.5.5.1. Reference data and session data

When a new Trading day is assigned, the data in the tables linked to the Trading day number are loaded anew. For the tables that are not linked to the Trading day number, new records are added in accordance with the new data available in the Trading day; the records which do not correspond the actual Trading day data will be deleted. The reference data are sent out within the tables of the 'FORTS_REFDATA_REPL' stream. As a result, the new record with a new Trading day number is added into the 'session' table.

2.5.5.2. Funds and positions

When a Trading day changes, the data on funds, limitations and clients positions are updated as following: only the records which have been modified are subject to change (including the 'FORTS_PART_REPL', 'FORTS_POS_REPL' and 'FORTS_INFO_REPL' streams).

2.5.5.3. Orders

Upon changing the Trading day, the multi-day orders are relisted automatically except those which are expired. The relisting is made by deleting an old order and adding a new one with a new number.

2.5.5.4. Instruments

When switching the Trading day, the system deletes expired trading instruments and adds new ones. New instruments can be traded starting from the main trading session; however, these new instruments appear in the system and are transmitted in the replication stream. They are also marked with a special sign in the 'fut_sess_contents' and 'opt_sess_contents' tables.

2.5.5.5. Replication streams

The replication streams can be closed and then reopen again by the trading system servers, yet some streams may transmit notification about changing the life number of a scheme.

For now, the following streams can be reopen without changing life numbers:

- 'FORTS_COMMON_REPL' — general market data.
- 'FORTS_VOLAT_REPL' — the current volatility values.
- 'FORTS_VM_REPL' — the current variation margin (premium) value

The following streams are not subjects to reopen:

- 'FORTS_REFDATA_REPL' — reference data
- 'FORTS_TRADE_REPL' — trading data
- 'FORTS_USERORDERBOOK_REPL' — snapshots of order books
- Streams with aggregated order books.
- 'FORTS_PART_REPL', 'FORTS_POS_REPL', 'FORTS_INFO_REPL'
- 'RTS_INDEX_REPL' — exchange indices
- 'FORTS_CLR_REPL'.

2.5.6. Event-sensitive scheme for data synchronizing

If a developed system demands the possibility of synchronizing the consistent states of data, then the event-sensitive scheme should be used. The following events are used to start synchronization:

- All data for a new Trading day are loaded and calculated (~00:24-00:25, Moscow time zone)
- Main clearing session has started (~23:50, Moscow time zone)
- All data after the main clearing session are recalculated (~00:24, Moscow time zone)
- Limits have been extended (during the trading session)
- The start of accepting orders in the opening auction (~8.50 Moscow time zone)
- The finish of accepting orders in the opening auction (~8.59 Moscow time zone)
- End of the opening auction (~9.00, Moscow time zone)
- Beginning of the settlement session (20.00, Moscow time zone)
- End of the settlement session (~20.09, Moscow time zone)
- Settlement price determination and fixation (19.00, Moscow time zone)

The 'sys_events' table is translated in the replication streams in order to inform outer systems about the events occurred

Field	Type	Description
replID	i8	Replication subsystem service field
replRev	i8	Replication subsystem service field
replAct	i8	Replication subsystem service field
event_id	i8	Unique event ID
sess_id	i4	Trading day ID
event_type	i4	Event type
message	c64	Description of the event

Field	Type	Description
server_time	t	Event time

The table is translated in all replication streams except the following:

- 'FORTS_USERORDERBOOK_REPL'
- 'FORTS_ORDBOOK_REPL'
- 'FORTS_MM_REPL'
- 'RTS_INDEX_REPL'
- 'FORTS_RISKINFOBACH_REPL'
- 'FORTS_RISKINFOBLACK_REPL'
- 'FORTS_TNPENALTY_REPL'
- 'MOEX_RATES_REPL'
- 'FORTS_REJECTEDORDERS_REPL'

The rules of the synchronization are the following: when a global event occurs in the system, and when all the data regarding this event are generated by all the subsystems, the new record is added to the 'sys_event' table containing the same 'event_id' value, with the 'event_type' value corresponding to the following event occurred:

- 1 (session_data_ready) - all data from the clearing system have been loaded into the trading system; this type of event is transmitted in all streams containing 'sys_events' table, except the 'FORTS_CLR_REPL' stream
- 3 (clearing_data_ready) - data are ready after the clearing session; this type of event is transmitted only in the 'FORTS_CLR_REPL' stream
- 5 (clearing_started) - clearing session has started; this type of event is transmitted in all streams containing 'sys_events' table, except the 'FORTS_CLR_REPL' stream
- 6 (extension_of_limits_finished) - limits have been extended; this type of event is transmitted in all streams containing 'sys_events' table, except the 'FORTS_CLR_REPL' stream
- 17 (clearing_trades_inserted) - Clearing trades have been generated; this type of event is transmitted in all streams containing 'sys_events' table, except the FORTS_CLR_REPL stream
- 18 (extension_of_limits_started_for_core) - limits have been extended; this type of event is transmitted in all streams containing 'sys_events' table, except the FORTS_CLR_REPL stream
- 22 (discrete_auction_matching_completed) - order matching completed and opening auction trades have been published; this type of event is transmitted in streams FORTS_TRADE_REPL, FORTS_ORDLOG_REPL, FORTS_DEALS_REPL, and FORTS_REFDATA_REPL
- 23 (discrete_auction_add_order_started) - the start of accepting orders in the opening auction; this type of event is transmitted in the 'FORTS_TRADE_REPL', 'FORTS_ORDLOG_REPL', 'FORTS_DEALS_REPL' and 'FORTS_REFDATA_REPL' streams
- 24 (discrete_auction_add_order_finished) - the finish of accepting orders in the opening auction; this type of event is transmitted in the 'FORTS_TRADE_REPL', 'FORTS_ORDLOG_REPL', 'FORTS_DEALS_REPL' and 'FORTS_REFDATA_REPL' streams
- 36 (start_of_calendar_date) - beginning of the calendar day; this type of event is transmitted in all streams containing 'sys_events' table, except the 'FORTS_CLR_REPL' stream
- 37 (settlement_session_started) - beginning of the settlement session; this type of event is transmitted in all streams containing 'sys_events' table, except the 'FORTS_CLR_REPL' stream
- 38 (settlement_session_finished) - end of the settlement session; this type of event is transmitted in all streams containing 'sys_events' table, except the 'FORTS_CLR_REPL' stream
- 47 (settlpricecalc_finished) - settlement prices are fixed; this type of event is transmitted in all streams containing 'sys_events' table, except the FORTS_CLR_REPL stream

An outer system may subscribe to receive the event table via all the available replication streams; when the data are ready, a notification will be sent to the outer system. The 'sys_event' table records, relating to the same event, will have the same 'event_id' field value in every replication stream. There are additional data available in the 'sess_id' and 'message' fields: the number of the current or upcoming trading day and a text message, respectively. Please also note that:

- The identity of service fields values (the 'replID' and 'replRev' fields) cannot be guaranteed for the same event in the different replication streams. You should view the 'event_id' value instead.
- The notification for the 'sys_event' table arrives AFTER all other data. It means that working in on-line mode, the system receives the newest data available, for example, instruments or the multi-day orders rolled over from the previous trading day, before adding records into the 'sys_events' table.

- Data consistency is not guaranteed in snapshot mode. Replication protocol does not remember the order of retrieving records between different tables in a stream and data are distributed in the order of description of tables in the schema, that is why snapshot recordings will not arrive in the order in which they arrived in on-line mode. For example, the 'session_data_ready' event at the moment of receiving a snapshot does not mean at all that the data is ready, because the 'session_data_ready' event may refer to the previous trading day. Therefore, you can use the notifications received in 'sys_events' to assess the consistent state of data in the system in online mode only.

2.5.7. Instrument status broadcast service

In the current version of SPECTRA, the calculation and broadcast of statuses for all instruments to clients occur with some delays compared to the actual time of status changes in the system, which negatively impacts users. Information about the current statuses of the trading session and instruments is available in the session, fut_sess_contents, and opt_sess_contents, tables of the FORTS_REFDATA_REPL stream. Within the framework of the import substitution program and to accelerate the broadcast of information on instrument and trading day statuses in the SPECTRA trading system, version 8.3 implements a new service: the Instrument status broadcast service.

The Instrument status broadcast service calculates the final statuses of instruments and trading day upon changes in their states and distributes them to consumers in the form of a replication stream. The service outputs three new streams:

- FORTS_SESSIONSTATE_REPL - Current trading day status.
- FORTS_INSTRUMENTSTATE_REPL - Statuses of instruments for the current trading day.
- FORTS_SECURITYGROUPSTATE_REPL - Group status of instruments for the current trading day.

The service distributes instrument statuses by instrument groups. An instrument can be subject to multiple group statuses, consequently, one instrument can belong to several groups. Distribution of statuses by groups (many instruments, few groups) helps save traffic, as there is no need to receive updates for all instruments, and speeds up the calculation of the instrument status because recalculations are performed only for instruments within a group. For example, in the updated for all instruments, and in this case, network broadcast takes seconds, receiving the status of one group takes milliseconds.

For users requiring even faster broadcast of instrument statuses, it is recommended to use data from the FORTS_SECURITYGROUPSTATE_REPL stream. By using instrument reference guides and the FORTS_SECURITYGROUPSTATE_REPL.security_group_state table, users can independently calculate the current statuses of instruments.

2.5.7.1. Algorithm for determining instrument statuses by group statuses

An instrument belongs to one or more groups. Each group has a trading status in the security_group_state table. The final instrument status is the most restrictive (strictest) status among all groups to which the instrument belongs. Statuses of instrument group:

- '4' - All prohibited.
- '2' - All prohibited except for order cancellation.
- '16' - Position increase prohibited.
- '8' - Order collection during the opening auction.
- '0' - All allowed.

Statuses are listed in order of decreasing priority (strictness of restriction).

Algorithm for determining instrument statuses by group statuses:

1. Note that data in the FORTS_REFDATA_REPL and FORTS_SECURITYGROUPSTATE_REPL streams must be synchronized using synchronous events.
2. Based on the FORTS_REFDATA_REPL stream, we receive a list of instruments and their connections to Underlying futures contracts, sections, trading modes, and instrument groups. Based on the session_data_ready event, we fix the list of instruments and their connections, which remains unchanged during trading.
3. The instrument_state container is constructed based on records from fut_sess_contents and opt_sess_contents, as well as records from FORTS_SECURITYGROUPSTATE_REPL.security_group_state. Container elements include:
   - isin_id - Unique instrument identifier. Retrieved from fut_sess_contents/opt_sess_contents.
   - section_id - Section identifier. Retrieved from fut_sess_contents.base_contract_code/opt_sess_contents.base_contract_code in fut_vcb/opt_vcb.
   - base_contract_id - Underlying futures contract identifier. Retrieved from fut_sess_contents.base_contract_code/opt_sess_contents.base_contract_code in fut_vcb/opt_vcb.
   - trade_mode_mask - Trading mode to which an instrument belongs. Obtained by subtracting one from the value in fut_sess_contents.trade_mode_id/opt_sess_contents.trade_mode_id and converting it to a mask with a single set bit (flag).
   - group_mask - Bitmask where bits (flags) with the numbers of the groups to which the instrument belongs are set. Value taken from fut_sess_contents.group_mask/opt_sess_contents.group_mask.

- The container map<security_group_id, state> security_groups_map. map is a sorted associative container that holds key-value pairs with unique keys.

4. Processing records from FORTS_SECURITYGROUPSTATE_REPL.security_group_state is performed as follows. For each record, based on the value of security_group_state.type, we determine the type of group status and perform the corresponding processing:

   a. security_group_state.type=1 - Trading day level.

      For all instruments in the instrument_state container, add an element to the security_groups_map container with fields: security_group_id, state. If such an element already exists in the container, update the state field within the element using the value taken from the FORTS_SECURITYGROUPSTATE_REPL.security_group_state record being processed.

   b. security_group_state.type=2 - Section level.

      In the instrument_state container, find all instruments where the section_id field matches the coresponding value from the processed record. For the found instruments, add an element to the security_groups_map container with fields: security_group_id, state. If such an element already exists in the container, update the state field within the element using the value taken from the FORTS_SECURITYGROUPSTATE_REPL.security_group_state record being processed.

   c. security_group_state.type=3 - Section + Trading mode level.

      In the instrument_state container, find all instruments where the section_id and trade_mode_mask fields match the coresponding values from the processed record. For the found instruments, add an element to the security_groups_map container with fields: security_group_id, state. If such an element already exists in the container, update the state field within the element using the value taken from the FORTS_SECURITYGROUPSTATE_REPL.security_group_state record being processed.

   d. security_group_state.type=4 - Underlying futures contract level.

      In the instrument_state container, find all instruments where the base_contract_id field matches the coresponding value from the processed record. For the found instruments, add an element to the security_groups_map container with fields: security_group_id, state. If such an element already exists in the container, update the state field within the element using the value taken from the FORTS_SECURITYGROUPSTATE_REPL.security_group_state record being processed.

   e. security_group_state.type=5 - Instruments within the trading mode.

      In the instrument_state container, find all instruments where the trade_mode_mask field matches the coresponding value from the processed record. For the found instruments, add an element to the security_groups_map container with fields: security_group_id, state. If such an element already exists in the container, update the state field within the element using the value taken from the FORTS_SECURITYGROUPSTATE_REPL.security_group_state record being processed.

   f. security_group_state.type=6 - Instruments within the trading mode + Underlying futures contract level.

      In the instrument_state container, find all instruments where the trade_mode_mask and base_contract_id fields match the corresponding values from the processed record. For the found instruments, add an element to the security_groups_map container with fields: security_group_id, state. If such an element already exists in the container, update the state field within the element using the value taken from the FORTS_SECURITYGROUPSTATE_REPL.security_group_state record being processed.

   g. security_group_state.type=7 - Instrument group level.

      In the instrument_state container, find all instruments where the group_mask field has the bit specified by group_mask field in the processed record set (instrument_state.group_mask & security_group_state.group_mask != 0) For the found instruments, add an element to the security_groups_map container with fields: security_group_id, state. If such an element already exists in the container, update the state field within the element using the value taken from the FORTS_SECURITYGROUPSTATE_REPL.security_group_state record being processed.

   h. security_group_state.type=8 - Instrument group + Underlying futures contract level.

      In the instrument_state container, find all instruments where the group_mask field has the bit specified by group_mask field in the processed record set (instrument_state.group_mask & security_group_state.group_mask != 0) and the base_contract_id field matches the coresponding value from the processed record. For the found instruments, add an element to the security_groups_map container with fields: security_group_id, state. If such an element already exists in the container, update the state field within the element using the value taken from the FORTS_SECURITYGROUPSTATE_REPL.security_group_state record being processed.

5. Determination of the current instrument status is performed as follows. For each element in the instrument_state container, iterate through the elements of the map<security_group_id, state> security_groups_map and determine the current instrument status as the worst (highest priority, strictest restriction) based on the state field values.

Additionally, in CGate (\SpectraCGate\SDK\samples | /usr/share/doc/cgate-examples), a test example of instruments_state has been added, which demonstrates the calculation of instrument statuses based on the FORTS_SECURITYGROUPSTATE_REPL stream.

2.5.8. Game and test mode trading schedule

Along with the SPECTRA trading system run in production mode, there are also a game trading system, and two test trading systems:

- T0, with the same trading system version as of that run in production mode

- T+1, with the trading system version the same as of the next planned one
- T+2, with a version of the trading system that is two step ahead of the current production system

The test trading schedule can be found on the Exchange website: https://www.moex.com/s438

Game system schedule:

- Opening auction before the morning trading session - order collection: 05:50:00 AM – 05:59:50 AM
- Morning trading session: 06:00:00 AM – 09:00:00 AM
- Opening auction: 08:50:00 AM – 08:59:50 AM
- Main trading session: 09:00:00 AM – 04:00:00 PM
- Daytime contract expiration: 01:00:00 PM
- Settlement price determination: 04:00:00 PM
- Evening contract expiration: 04:00:00 PM
- Evening trading session: 04:00:00 PM – 09:30:00 PM
- Settlement session: 05:00:00 PM
- Mark-to-market clearing session and contract expiration during clearing session: 09:30:00 PM
- Daily reports distribution within one hour – by 10:30 PM

Trading days align with those on the production system. Additional Weekend Trading Session is not conducted.

For the detailed trading schedule please refer either to table 'session' of stream 'FORTS_REFDATA_REPL', or to the table 'Session' available via the trading workstation (trading terminal).

2.5.9. Opening auction

Before the main trades in the SPECTRA system, an opening auction may be held. It is allowed to schedule several opening auctions within one Trading day. The opening auction is used to exclude an abnormal jump in the prices of derivatives at the opening of trading. This mechanism is implemented in the format of a discrete auction of orders submitted to the opening auction at a single opening auction price. The opening auction price is selected from the condition for performing trades with the maximum number of contracts for the orders announced at the time of the opening auction. The opening auction consists of a period for collecting orders and matching orders at the opening auction price, determined based on information about all orders at the end of the orders collection period.

The main parameters of the opening auction:

- Only futures, including perpetual futures, can participate in the opening auction. There are no calendar spreads and options. The sign of admission of the instrument to participate in the opening auction is set at the underlying asset level and applies to all instruments of this underlying asset.
- During the period of collecting orders, only exchange quotation orders (‘Day’ type), including icebergs, can be placed in the opening auction. BOOK, FOK, IOC orders are not allowed.
- Control of cross-orders: the intersection of the price levels of opposite orders with the same TIN is not allowed (a later application is removed), while the settings "Allow cross-trades" and "Cancel a passive order in a cross-trade" are not taken into account.
- The orders prices must be within the limits of the price corridor of the instrument.
- Restrictions on placing orders in the relevant trading period at the BF level also apply to the opening auction held before this trading period.
- Exchange orders from the previous trading period participate in the opening auction (including icebergs and Book-or-Cancel orders).
- During the period of collecting orders, you can place and cancel orders (if this is allowed for the instruments of this UA in this auction), the moving of orders is prohibited in the opening auction.
- The start time for collecting orders is 10 minutes before the start of the trading period. This moment in the system corresponds to the arrival of the 'discrete_action_add_order_started' synchro event in the sys_events table of the FORTS_TRADE_REPL, FORTS_ORDLOG_REPL, FORTS_DEALS_REPL and FORTS_REFDATA_REPL streams.
- The end time for collecting orders is random for all instruments in the interval specified in the 'discrete_auction.add_order_finish_from' and 'discrete_auction.add_order_finish_till' fields of the discrete_auction table in the FORTS_REFDATA_REPL stream. This moment in the system corresponds to the arrival of the 'discrete_action_add_order_finished' synchro event in the sys_events table of the FORTS_TRADE_REPL, FORTS_ORDLOG_REPL, FORTS_DEALS_REPL and FORTS_REFDATA_REPL streams.
- After the completion of the application collection phase and before the end of the auction, operations with applications, including deletion, are not allowed.
- After the auction and the conclusion of trades at the opening price, the remaining orders are matched, taking into account synthetic liquidity from calendar spreads. The trades prices taking into account synthetic matching may differ from the opening price. At this stage,

orders for calendar spreads can be canceled if the crossness conditions are violated. Completion of the opening auction can be tracked by receiving synchronous event (22) discrete_auction_matching_completed in the sys_events tables of the FORTS_TRADE_REPL, FORTS_ORDLOG_REPL, FORTS_DEALS_REPL, and FORTS_REFDATA_REPL.

- Orders that are not matched during the opening auction are placed in the main trading session.

Information about opening auctions in the tables of the trading interface:

- Information about the schedule of the opening auctions is contained in the discrete_auction and discrete_auction_base_contract tables of the FORTS_REFDATA_REPL stream of the trading interface.
- The opening auction price is transmitted in the gateway in the 'opening_auction_price' field of the common table of the FORTS_COMMON_REPL stream during the morning and main trading session. In the evening trading session, the opening auction price is zero.
- Information about the trading status of the instrument is transmitted in the state field (values 6, 7) in the fut_sess_contents table of the FORTS_REFDATA_REPL stream.
- All orders and trades in the opening auction have a special sign that is broadcast in the tables of orders and trades - DuringDiscreteAuction (0x4000000000000000).

2.5.10. Trading on weekends.

The system supports weekend trading. An additional weekend trading session (AWTS), with the option to conduct opening auction, starts on weekends and relates to the Trading day immediately following the given trading weekend.. Each weekend has its own additional trading session. Settlement and clearing sessions during weekend trading are not conducted. All settlement is performed on the closest next working day.

Additional weekend trading session timeframe: 10:00 AM - 07:00 PM. Opening auction is planned before each additional weekend trading session. Auction time: 9:50 AM - 10:00 AM. More information about weekend trading can be found in the Trading on weekends section.

2.6. Risk management and limitation of trading operations

2.6.1. Collaterals

The Risk Management System implemented into SPECTRA allows to dramatically reduce risks of non-fullfilment of obligations by permanent evaluation of market risks for every participant's position. The core of the system is the initial margin (hereinafter also referred to as IM) calculation algorithm, applied to open positions and orders recognized on clearing and trading participants' position accounts.

One of the key features of the SPECTRA Risk Management System is the calculating initial margin on orders and positions per one trading transaction in online mode. Therefore, it is almost impossible for non-pledged orders and trades to appear in the system, because the initial margin is always verified before any relating order appears in the system.

Another important feature of the SPECTRA Risk Management System is the three-level calculating scheme, in accordance to which the position accounts are subdivided into three groups:

Settlement Account - the upper-level account of a clearing participant (i.e. Clearing Firm). The Settlement Account is an independent account for recognizing collateral assets margined by a trading participant (and/or their clients), orders added for all lower-level accounts (sub-accounts) belonging to the Settlement Account, trades performed basing on these orders, and resultant positions. Therefore, a position for an instrument recognized under the Settlement Account is equal to the net amount of all positions for the given instrument which are recognized under the sub-accounts.

The amount of initial margin for a Settlement Account is calculated independently of the other settlement account. All settings of SPECTRA Risk Management System are specified by the Central Counterparty (clearing firm).

During a clearing session, the system calculates the clearing participant's obligations and requirements (variation margin, premium, commission fees, etc.). Also, the system calculates collateral sufficiency to meet initial margin requirements.

Settlement accounts are subdivided into three sections:

- own - trades are covered with clearing participant's assets;
- client - trades are covered with direct clients and clearing participant's 2nd-level clients' assets;
- Asset management - trades are covered with assets managed by a clearing participant.

For each clearing participant (Clearing Firm), there are at least two Settlement Accounts assigned: own and client.

Each Settlement Account will be identified by trading system SPECTRA in accordance with its unique 5-digit code.

Brokerage Firm - a sub-account of a Settlement account, which can be set up upon application by the clearing participant (Clearing Firm). Each Brokerage Firm belongs to a single Settlement Account, where the Settlement Account is subject to change upon the clearing participant application applied to the clearing entity. To make a Settlement Account available in SPECTRA , there should be at least one Brokerage Firm bound to it.

The clearing system of SPECTRA recognizes the initial margin deposited by a client, and/or their clients to client section of the Brokerage Firm. Detailed information on initial margin values are available in reports.

By default, initial margin value is calculated in half nett mode (margin_type =3 in command ChangeBFParametersNextSession) in accordance with risk values for positions recognised at client sections of Brokerage Firm. Nett mode is also available for Brokerage Firm for initial margin calculation (margin_type =4 in command ChangeBFParametersNextSession); using this mode, the initial margin value of a Brokerage Firm will be calculated according to nett sum of all positions for the given instrument at all sections of the Brokerage Firm, and total amount of orders added for sections of the Brokerage Firm.

All margining settings of a Brokerage Firm can be changed by a clearing participant (Clearing Firm) using the command ChangeBFParametersNextSession.

Separate Brokerage Firm (**SBF**) - a separated sub-account of a Settlement account, similar to common Brokerage Firms, purposed for recognizing collateral assets deposited by a client, and/or their clients, and not recognized at any section of common Brokerage Firms.

Detailed information on initial margin values are available in reports.

Also, each SBF contains a special account called liquidation account, which is purposed to recognize positions based upon trades performed by Clearing Centre in order to handle obligations unperformed by a clearing participant (for example, an unperformed Margin Call requirement for the Settlement Account). None of clearing participants (a Clearing Firm) is able to add orders with the liquidation account specified; excluding the orders aimed to lower the amount of an opened position for the given account. Also, clearing participants (Clearing Firms) are able to transfer positions from the liquidation account to other Brokerage Firms' accounts (command TransferClientPosition).

Client Account - a sub-account of Brokerage Firm. The low-level account, which can be opened upon the application by client, and specified as the 'client code' in order adding transaction. This is the primary account to recognize orders added by participant and/or client, trades performed upon these orders, and open positions; the initial margin value will be calculated using these orders and positions. One can change Client Account margining settings via commands ChangeClientParameters, ChangeClientParametersNextSession, ChangeBFClientDefaultParametersNextSession.

The clearing system of SPECTRA recognizes the initial margin deposited by a client, and/or their clients to a Client Account. Detailed information on initial margin values are available in reports.

2.6.1.1. Margining of calendar spreads

Margining of calendar spreads on futures (multi-leg orders), and opposite direction positions with different exercise dates for the same underlying asset (intermonth spread) may proceed in two modes:

- half nett - IM value will be calculated based upon the larger IM of the instruments in the spread;
- nett - IM value will be calculated based upon price variable rate value of the instruments in the spread.

For a Settlement Account, the solely available margining mode is the nett mode.

For a Brokerage Firm, it is possible to change calendar spread margining mode only if IM value for the Brokerage Firm proceeds in the nett mode. To change the calendar spread margining mode, please use setting 'calendar_spread_margin_type' of command ChangeBFParametersNextSession.

To change the calendar spread margining mode for a Client Account, please use setting 'calendar_spread_margin_type' of command ChangeClientParametersNextSession.

2.6.2. Trading limits

Trading limits are aimed to restrict a participant, and/or their clients, from adding orders and open positions for position accounts.

Trading limit for a Settlement Account can be calculated based upon total imputed value of IM recognized for the given Settlement Account, i.e. total value of IM recognized for all sub-accounts of the given Settlement Account. Collateral assets may consists of Russian Rubles, foreign exchanges, and securities.

Trading limit for a Settlement Account can be changed by depositing, withdrawal, or transferring collaterals, based upon requests applied to the clearing entity, or clearing depositary (as well as to other settlement entities, once there have been any collateral deposited) by participant via the appropriate EDM systems. Another way to change trading limit is to transfer collateral (Russian Rubles) from one sub-account of Settlement Accounts (Brokerage Firm/Separated Brokerage Firm) to another using command ExchangeBFMoney.

Trading limits are used to reserve negative variating margins, withdraw fees and premiums, accrue premiums and reserve collaterals.

By default, trading limit for a Settlement Account (similar to that of Settlement Accounts) will be calculated based upon total imputed value of IM recognized for sub-accounts of a Brokerage Firm. Collateral assets may consists of Russian Rubles, foreign exchanges, and securities. To change trading limit of a Brokerage Firm, please use command ChangeBFMoney.

For a Brokerage Firm, it is possible to switch trading limit calculation mode, in order to calculate the trading limit independently of value of IM recognized for the Brokerage Firm's sub-accounts. To switch to that mode, please use command ChangeBFLimit. Also, trading limit will be changed in accordance with the profit/loss value resulted from the clearing session (variation margin, premium and fees).

To change trading limit mode, please use setting 'limit_tied_to_money' of command ChangeBFParametersNextSession.

Trading limit for Client Account does not depend on IM value recognized for the given account, To manage trading limits for Client Accounts, please use command ChangeClientMoney, with the following possibilities:

- Set up/change/delete trading limits;

- The client's trading results will be automatically applied for limits in the next trading day.

As a rule, one is able to add an order only if there are sufficient limits to cover the required IM for all three levels (CLient Account, Brokerage Firm, Settlement Account). It is possible to switch off the limit sufficiency verification for Brokerage Firm and Client Account using commands ChangeBFParametersNextSession and ChangeClientMoney, appropriately.

Please note that it is not possible to switch off the limit sufficiency verification for a Settlement Account.

2.6.2.1. Unified Collateral Pool

If a Settlement Account belongs to Unified Collateral Pool (UCP), then, instead of collateral assets, clearing system of SPECTRA recognizes asset profiles transferred to its sub-accounts (Brokerage Firms, SBF) from clearing systems of Securities Market and FX Market. The asset profiles are transferred based upon the clearing participant's requests applied to the clearing entity via the appropriate electronic document management system. Please note that it is impossible to transfer IM to a Brokerage Firm of a Settlement Account belonging to UCP. Please also note, that it is impossible to transfer profiles between Brokerage Firms of different Settlement Accounts belonging to the Unified Collateral Pool.

An asset profile is recognized in SPECTRA trading system as imputed value of the asset according to its profile sign (+/-, to change the trading limit accordingly), and as position for a separate instrument (excluding Russian Ruble related profile), at the same time. The position for a separate instrument will be added either onto the Brokerage Firm's client account (if the profile transferred onto the same client account), or onto a Separate Brokerage Firm (with the client code '000'; if the profile transferred with no client account specified for this transfer transaction). There is a dedicated instrument (name suffixed with '_CLT') in SPECTRA for every asset eligible for profile transfer, the one is restricted to add orders and perform trades for. The only method to change a position opened for that dedicated instrument is to transfer the appropriate profile to/from SPECTRA clearing system. The IM calculation at participant's accounts/sub-accounts for the dedicated instrument position is similar to that of a futures position for the same underlying asset, i.e. the position will be margined exactly the same way.

All other specifics of managing the IM value and trading limits for sub-accounts of Unified Collateral Pool are similar to that of the standard Settlement Accounts.

2.6.3. Limitations on trading operations and opening positions for clients

The SPECTRA system provides an opportunity to introduce additional restrictions on client's trading operations, which are formulated in the system as prohibitions.

2.6.3.1. Prohibitions - general information

The SPECTRA system allows to prohibit a client (a group of clients) to do some transactions for a trading instrument or a group of instruments. General principles of the prohibitions mechanism:

- Prohibitions apply to a standard subset of accounts: CF, BF, clients. Please note that there are currently no prohibitions applied for CF.
- Prohibitions apply to the following subsets of trading instruments:
  - a single future or all options on the same underlying instrument;
  - instruments with a common underlying asset;
  - trading section instruments;
  - all instruments.
- Prohibitions can be used not only by trading administrators, but also by participants who have the appropriate authority. Automatic prohibitions are also provided.
- Prohibitions take effect immediately after setting. There is no way to set the interval for prohibitions application.

Prohibitions attributes

Attribute	Description
xprohibition_id	The unique ID of the prohibition.
client_code	Client code. It may contain 2, 4 or 7 characters depending on which subset of accounts the prohibition apply to.
initiator	Initiator of the prohibition. The value depends on the login level that set the prohibition. It can take the following values: • 1 - CF Chief trader; • 2 - CC Administrator; • 3 - TS Administrator. The parameter is used to determine whether the prohibition applies to the login submitting the transaction. The table below shows the matrix of the impact of prohibitions from different initiators on different levels of logins submitting a transaction.

Attribute	Description
	Please note that all prohibitions set by the user using gateway commands (see below) have an initiator - '1 - CF Chief trader'.
section	Section ID. It is used to set a prohibition of trading all instruments in the section. It can take the following values: • 1 - Securities; • 2 - Commodity; • 3 - Money.
base_contract_code	Underlying asset code. The prohibition is set for instruments based on this underlying asset only.
isin_id	Instrument unique ID. It is used to prohibit trading for a specific instrument.
priority	Priority of prohibition. Automatically calculated from the prohibition parameters (see below).
group_mask	Bitmask of groups for which there is a prohibition. It can take the following values: • 0x40000000 - Futures. • 0x80000000 - Options.
type	Type of prohibition. It can take the following values: • 0 - No prohibitions. Used for pinpoint permission in case of a broader prohibition; • 1 - Prohibited to open positions; • 2 - Prohibited to add any orders; • 3 - Prohibited to open sell positions; • 0x08 - BF prohibition to add requests for exercising; • 0x10 - Chief Trader prohibition to add requests for exercising; but to himself - it is possible; • 0x20 - Prohibition of requests without auto-confirmation (RFS); • 0x40 - Prohibition to request liquidity stream (RFS); • 0x80 - Prohibition to perform trades with insufficient number of quotes (RFS); • 0x100 - Prohibition to request liquidity stream with limited lifetime of quotes (RFS). Prohibition types 0-3 are ordered by severity as follows: 2 > 1 > 3 > 0.
is_legacy	Sign of a user prohibition. It can take the following values: • 0 - The prohibition set by the Trading Administrator or Clearing Administrator; these prohibition cannot be changed by trader; • 1 - The prohibition set by a trader and can be changed by means of the gateway API.

Table 1. The matrix of the influence of prohibitions

Login level / Initiator	CF Chief trader	CC Administrator	TS Administrator
Client	+	+	+
BF	+	+	+
CF, **Trader**	+	+	+
CF, Chief trader	-	+	+

Prohibitions priorities

Transaction parameters can satisfy several prohibitions. To determine the effective prohibition, all prohibitions are sorted in order of priority and the prohibition with the highest priority is selected as effective. Priorities are fixed and determined by a combination of prohibition parameters, the more detailed the prohibition, the higher its priority. Priorities are automatically calculated by the system based on the prohibition parameters. The following priorities are provided (listed in order from highest to lowest priority):

Priority	Account group	Instrument group
9	Client code	instrument, isin_id != 0
8	Client code	underlying asset, base_contract_code != 0
7	Client code	all instruments

Priority	Account group	Instrument group
6	BF code	instrument, isin_id != 0
5	BF code	underlying asset, base_contract_code != 0
4	BF code	all instruments
3	CF code	instrument, isin_id != 0
2	CF code	underlying asset, base_contract_code != 0
1	CF code	all instruments

Applying of prohibitions

The following is a very brief description of the algorithm for finding an effective prohibition for an order:

- For each initiator that affects the login from which the order was added (see the description of the 'initiator' parameter above), all prohibitions related to the instrument and the seven-digit client section specified in the order are selected.
- Among this set of prohibitions, the prohibitions with the highest priority are selected, and among them the strictest prohibition is selected (see the description of the 'type' parameter above).
- Among the prohibitions selected for each initiator, the most strict prohibition is selected, without regard to priority.

Thus, since the prohibitions are intended to restrict operations, the most strict one is selected among the prohibitions of different initiators. That is, even an administrator will not be able to overcome the prohibition on orders set for all instruments on the BF (priority 4) by some BF level login, or the prohibition on orders set for all instruments on the CF by the Chief trader (priority 1). Because the prohibition on orders is the most strict prohibition. If it was set by another initiator, then the prohibition can be canceled (or overcome for a specific instrument and / or client) by this initiator only.

Interfaces

Prohibitions are broadcast in the gateway in 'prohibition' table in 'FORTS_PROHIBITION_REPL' stream.

In the gateway, 'FutChangeClientProhibit' and 'OptChangeClientProhibit' commands are provided to manage prohibitions. It is possible to forbid opening positions and placing orders for a specific client (for all clients), instrument (for all instruments) or underlying asset (for all UAs).

Setting prohibitions (prohibition configuration examples)

To set prohibitions, the commands mentioned above are provided in the gateway. When setting prohibitions, special attention should be paid to their correct configuration. An incorrectly constructed prohibition can lead to an unreasonably large number of prohibitions in the system, which, in turn, negatively affects the performance of the entire prohibition management system as a whole, and above all for the user himself who has set an incorrect prohibition (brakes when opening streams and sending control commands for prohibitions). For example, instead of setting many prohibitions for almost every client of the BF, it is much more efficient to set one general prohibition for the BF, and set the appropriate permissions for the necessary clients. The following are examples of configuring prohibitions in various situations. The configurations provided in the scenarios allow minimizing the total number of prohibitions.

I. Examples of setting prohibitions when all clients have the same rights.

1. Prohibit all operations for n UA (list of prohibited underlying assets). n < 0.5*M, where M – total number of UA.
   - Set n BF prohibitions on UA. Prohibition type - state=2 (prohibition on placing any orders), prohibition priority - priority=5 (BF code/ UA, base_contract_code != 0).

     The addition of a new UA included in the list of prohibited UA requires an explicit prohibition.

     The addition of a new UA that is not included in the list of prohibited UA does not require setting a new prohibition.
2. Prohibit all operations for n UA (list of prohibited UA). n > 0.5*M, where M – total number of UA. The scenario differs from the previous one in the ratio between the number of prohibited and allowed UA.
   - Set BF prohibition on all UA. Prohibition type - state=2 (prohibition on placing any orders), prohibition priority - priority=4 (BF code/ all instruments).
   - Set M-n BF permissions on allowed UA (inverse set to prohibited UA). Prohibition type - state=0 (no prohibitions), prohibition priority - priority=5 (BF code/ UA, base_contract_code != 0).

     The addition of a new UA included in the list of prohibited UA does not requires setting prohibition.

     The addition of a new UA that is not included in the list of prohibited UA requires setting a permission for it.

II. Examples of setting prohibitions when clients are divided into two groups: qualified investors and non-qualified investors.

Number of qualified investors = m. Number of non-qualified investors = n.

Non-qualified investors should have access to UA in group L only . Qualified investors have access to all UA.

1. Allow unqualified investors to trade on UA from L group only and prohibit on other UA. Allow qualified investors to trade on all UA. m < n - fewer qualified investors.

   - Set BF prohibition on all UA. Prohibition type - state=2 (prohibition on placing any orders), prohibition priority - priority=4 (BF code/ all instruments).
   - Set BF prohibitions on UA from L group. Prohibition type - state=0 (no prohibitions), prohibition priority - priority=5 (BF code/ UA, base_contract_code != 0). That is, it will be allowed to all clients of the BF.
   - Set permissions to qualified BF investors on all UA. Prohibition type - state=0 (no prohibitions), prohibition priority - priority=7 (Client code/ all instruments). That is, it will be allowed to all qualified BF investors only.

     The addition of a new qualified investor requires setting a permission for him.

     The addition of a new non-qualified investor does not require changes to the prohibitions.

2. Allow unqualified investors to trade on UA from L group only and prohibit on other UA. Allow qualified investors to trade on all UA. m >= n - more qualified investors. The scenario differs from the previous one in the ratio between the number of qualified and non-qualified investors.

   - Set prohibitions to non-qualified BF investors on UA that are not included in L group. Prohibition type - state=2 (prohibition on placing any orders), prohibition priority - priority=8 (Client code / UA, base_contract_code != 0).

     The addition of a new qualified investor does not require changes to the prohibitions.

     The addition of a new non-qualified investor requires setting a prohibition for him.

Prohibitions with custom priority

The system provides an opportunity to set prohibitions with the custom priority , and this priority will be higher than those that are set by the system automatically (priority from '1' to '9'). This feature allows to configure prohibitions configuration scenarios more flexible. The following priorities are provided:

- 10 – low custom priority;
- 11 – medium custom priority;
- 12 – high custom priority;

New 'client_priority' (i4) field was added to FutChangeClientProhibit and OptChangeClientProhibit commands for settings prohibitions with custom priorities. This field takes the value '10', '11' or '12' (low, medium, high). If the 'client_priority' field contains the value '10', '11' or '12', then a prohibition is set with the specified user priority. If the custom priority is not specified (the value is '0'), the priority will be set automatically, in accordance with the prohibition parameters.

2.6.3.2. Automatic prohibitions

Also, the system provides possibilities of automatic prohibition on opening positions and adding orders in case of occurrence of a large negative trading limit. The following parameters are used to manage the prohibition settings:

- Pr_state - Automatic prohibition application; 0 - do not apply prohibition, 1 - apply prohibition.
- Pr_type - Prohibition type; 0 - prohibition on opening positions, 1 - prohibition on adding orders.
- Pr_coeff - Multiplying coefficient; a positive fractional number with an accuracy of 2 decimal places.
- Del_ord - Action on applying the prohibition; 0 - do not cancel orders, 1 - cancel orders.

The settings can be applied by a clearing participant acting on behalf of a Brokerage Firm. There are two sets of parameters available for applying prohibitions, with one set is applicable for the Brokerage Firm's clients, and the other one is applicable for the Brokerage Firm as a whole.

The prohibition settings are applied based on a clearing participant's request sent to the Clearing Firm using the appropriate electronic document flow systems. The new settings will be applied at the next clearing session. Please note that the prohibition request should be sent not later than 1 hour prior to the clearing session.

Apply prohibitions. After the limits have been extended during the clearing sessions, the prohibition will be automatically applied for the appropriate 7-symbol client section, or Brokerage Firm, when all the following conditions are met simultaneously:

$$\begin{cases} Limits_set = 1 \\ Pr_state = 1 \\ FreeMoney < -Pr_coeff * max(Trade_limit, 0) \end{cases}$$

, where

- Limits_set – Client limit verification setting;

- Trade_limit – Trading limit, i. e. funds and collateral, including liquidity ratio.
- FreeMoney – Free funds available for the client section/Brokerage Firm

The prohibition type is specified via parameter 'Pr_type'. If 'Del_ord=1', then all active orders will be automatically cancelled at applying the prohibition. Limit verifications are held independently for the Brokerage Firm and for the clients.

Cancel prohibitions. The applied prohibitions cannot be cancelled by the Brokerage FIrm; instead, the prohibitions will be cancelled automatically following the every minute verification result, when there is at least one condition met:

$$\left[\begin{array}{l} FreeMoney \geq 0 \\ Pr_state = 0 \\ Limits_set = 0 \end{array}\right.$$

Example. If a prohibition is set on opening positions, a client is able to cancel the orders, or close the position which causes the increased collateral requirements. The prohibition then will be automatically cancelled in not more than a minute.

One is unable to cancel prohibitions during the night sessions (12 AM - 09 AM Moscow time), despite the availability of trading system.

By default, prohibitions are disabled from being set for all Brokerage Firms (Pr_state=0).

2.6.4. Position (obligation) transfer

The SPECTRA system provides possibility to transfer a positions from one Brokerage Firm client to another client of the same Brokerage Firm.

To transfer a position from one clearing register to another, a clearing participant should add a new transaction into the Trading system.

Verification procedures on position transfer are the same as that of adding an order. Additionally, it is verified that volume of the position to be transferred does not exceed that of the donor account. Also, the VAT/personal data (including separated brokerage firms accounts) must be equal for both accounts.

Technically, the position transfer is a trade, performed between a donor account and a recipient account. Juridically, it is not a trade (for details see Trade types, created upon exercising and expiration of futures and options). Position transfer is visible both in the gateway and in the reports (f04/o04).

2.6.5. Pausing trading session for extending limits of trading prices fluctuations

Technically, the following actions take place in the SPECTRA system when pausing trading:

- When the condition is set to pause trading for a certain underlying asset, then the trading pauses for this asset.
- The trading administrators calculate the new extended limits of prices fluctuations.
- The amount of collateral is recalculated for every position, which includes the underlying asset (if the limits extend, then the amount grows)
- After the collateral is recalculated, the trading still pauses, allowing participants to delete orders.
- The trading resumes in the standard mode.

The corresponding notifications are sent on every action listed above (see the 'sys_message' table of the 'FORTS_REFDATA_REPL' stream):

- Warning about the upcoming trading pausing for a certain instrument if the prices remain unchanged.
- Notification about pausing the trading.
- Notification about successful recalculation of collateral (orders can now be deleted).
- Notification about resuming the trading.

2.6.6. Risk parameters forecast information for trading participants

The Trading System provides risk parameters forecast information to the trading participants (via service ForecastIM). The service recalculates the Initial Margin value at a specified time interval, forecasting a new most probable value which would occur after the limits were extended. Then, the new data will be transmitted to the trading participants.

This will be done in the following steps:

- At the time interval of 1 minute, the market condition is being analyzed for instruments due to which the limit extension procedure may, or will be performed (once an instrument's price value stays close to a specified limit for more than X minutes).
- Once such instruments are detected, the Initial Margin will be recalculated for client portfolios. The new risk parameters will be applied for the instruments, according to the new limit values set after extension.

- The recalculated funds are transmitted within the table part_sa_forecast of the replication stream FORTS_FORECASTIM_REPL.
- Once the limit extension risk is over due to the market condition change, or when the limits have been already extended before, the service halts recalculation and transmitting of the Initial Margin value. All the previously received data will be declared as non-valid (all forecasted data in the appropriate table will be cleared at receiving command CLEARDELETED with the maximum possible revision value).

If the limits have been already extended twice for the same instrument during a single trading day, there will be no more risk parameters forecasted and forecast information transmitted for this given instrument during this trading day.

2.6.7. Blocking the brokerage part of the client fee

The SPECTRA system allows the broker to block part of the client fee on the exchange side. The blocked fee is taken into the client account, reducing the client's free funds (money_free) by the amount of the blocked part. The blocking is carried out during the trading day and is reset in the clearing session.

Brokerage fee is calculated based on the exchange fee. The brokerage part of the fee is calculated according to the following formula for each trade:

$$broker_fee = N \times \max\left(lower_fee, \min\left(upper_fee, \mathrm{round}\left(multiplier \times {}^{ex_fee}/_{N}, 2\right) + additive\right)\right)$$

- *N* – number of contracts per trade;
- *lower_fee* – minimum brokerage fee per contract;
- *upper_fee* – maximum brokerage fee per contract;
- *multiplier* – multiplier to the amount of exchange and clearing fees;
- *ex_fee* – clearing and exchange fees per trade;
- *additive* – constant addition per contract.

The broker can set 'lower_fee', 'upper_fee', 'multiplier' and 'additive' parameters using the command SetBrokerFeeParamNextSession. Parameters can be set for an individual client and for the entire brokerage firm. The parameters set for the BF are used in the calculation for all of its clients. The set parameters will be applied in the next trading day. The set parameters are broadcast in the gateway in the stream FORTS_BROKER_FEE_PARAMS_REPL.

For example, the broker always takes half the exchange fee - then 'multiplier' = 0.5, 'additive' = 0, 'lower_fee' = 0.01, 'upper_fee' = inf. Or the broker always takes 2 rubles for any contract - then 'multiplier' = 0, 'additive' = 2, 'lower_fee' = 2, 'upper_fee' = 2.

The brokerage fee is broadcast in the gateway in the table 'part' of the stream FORTS_PART_REPL (total by client) and in the stream FORTS_BROKER_FEE_REPL (by trades).

2.6.8. Negative prices support in SPECTRA

Starting from this release SPECTRA supports negative prices, i. e. correct system behavior if futures prices and options strikes fall below zero during trades or as the result of clearing. There are two different modes for every underlying asset:

- Mode in which futures prices and options strikes are not limited. Negative and zero futures prices and options strikes are allowed in this mode. In this case options prices, volatility and risks are calculated based on Bachelier model or modified Black-Scholes model, which takes into account only intrinsic value of an option in negative range.
- Mode in which futures prices and options strikes are limited to be positive. In this mode prices cannot fall below zero during or as the result of trades. Option prices are calculated based on Black-Scholes model (Bachelier model may be used as alternative). However, this mode allows manual negative exercise price and /or indicative current market price setting in case of corresponding NCC decision. Nevertheless futures prices and options strikes have positive limits.

Mode and option pricing model are set on underlying contract level and are effective for all instruments of this underlying contract. Modes and option pricing models can be switch during the clearing session. To set the mode and risk model the following parameter of underlying contract is used:

- negative_prices: 1 – futures prices and options strikes are not limited; 0 - futures prices and options strikes are limited to be positive only.
- option_model: 1 – Bachelier model; 0 - Black-Scholes model.

Current parameter values is published in the 'fut_vcb'/'opt_vcb' tables of FORTS_REFDATA_REPL data stream in the gateway.

In the prohibition mode of negative prices (negative_prices = 0), if the NCC decides accordingly, there is a possibility to manually set indicative current market price, which is published in FORTS_COMMON_REPL data stream. This price affects indicative current variation margin translated in FORTS_VM_REPL stream and current theoretical options price published in FORTS_VOLAT_REPL stream. To indicate that current market price for futures is set manually the following parameter is used:

- price_assigned_by_admin – attribute of manual current market price setting by trades Administrator.

Fields of the trading interface tables, where negative values may appear in the negative prices mode ('negative_prices' = 1):

Stream	Table	Field	Description
FORTS_TRADE_REPL	orders_log	price	Price of the order
FORTS_TRADE_REPL	orders_log	deal_price	Price of the trade
FORTS_TRADE_REPL	user_deal	price	Price of the trade
FORTS_ORDLOG_REPL	orders_log	price	Price of the order
FORTS_ORDLOG_REPL	orders_log	deal_price	Price of the trade
FORTS_USERORDERBOOK_REPL	orders	price	Price of the order
FORTS_ORDBOOK_REPL	orders	price	Price of the order
FORTS_COMMON_REPL	common	best_buy	Best bid
FORTS_COMMON_REPL	common	best_sell	Best offer
FORTS_COMMON_REPL	common	open_price	Opening price
FORTS_COMMON_REPL	common	close_price	Closing price
FORTS_COMMON_REPL	common	price	Price of the last trade
FORTS_COMMON_REPL	common	min_price	The low price
FORTS_COMMON_REPL	common	max_price	The high price
FORTS_COMMON_REPL	common	avr_price	Average weighted price
FORTS_COMMON_REPL	common	settlement_price_open	Settlement price at the start of the trading day
FORTS_COMMON_REPL	common	market_price	Current market price
Streams of the aggregated order-books FORTS_AGGRXX	orders_aggr	price	Price level
FORTS_POS_REPL	position	waprice	Volume-weighted average price
FORTS_POS_REPL	position_sa	waprice	Volume-weighted average price
FORTS_REFDATA_REPL	fut_sess_contents	limit_up	Upper price limit
FORTS_REFDATA_REPL	fut_sess_contents	limit_down	Lower price limit
FORTS_REFDATA_REPL	fut_sess_contents	settlement_price_open	Settlement price at the start of the trading day
FORTS_REFDATA_REPL	fut_sess_contents	settlement_price	Settlement price after the last clearing session
FORTS_REFDATA_REPL	fut_instruments	settlement_price_open	Settlement price at the start of the trading day
FORTS_REFDATA_REPL	fut_instruments	settlement_price	Settlement price after the last clearing session
FORTS_REFDATA_REPL	opt_sess_contents	strike	Exercise price
FORTS_MM_REPL	fut_MM_info	price_edge_sell	Price of the worst sell order included in the spread
FORTS_MM_REPL	fut_MM_info	price_edge_buy	Price of the worst buy order included in the spread
FORTS_CLR_REPL	fut_sess_settl	settl_price	Settlement price
FORTS_INFO_REPL	futures_params	risk_range_center	Risk calculation center
FORTS_INFO_REPL	futures_params	settlement_price	Settlement price after the last clearing session
FORTS_INFO_REPL	options_params	strike	Exercise price
FORTS_REJECTEDORDERS_REPL	rejected_orders	price	Price of the order

In the positive prices mode ('negative_prices' = 0), in the case of an appropriate decision by the NCC, it is possible:

- use of negative futures exercise price;
- broadcast of a negative value as an indicative current market price set by the Trading Administrator ('price_assigned_by_admin' = 1) in the 'market_price' field.

Negative and zero values in the instruments trading codes are displayed as follows:

The codes with strike "-10":

- Contract short code ('short_isin'): "BR-10BF0".
- Long contract code ('isin'): "BR-7.20M250620CA-10".

The codes with strike "0":

- Contract short code ('short_isin'): "BR0BF0".
- Long contract code ('isin'): "BR-7.20M250620CA0".

2.6.9. Exchange services under Directive 6681-U - Regulation of margin trading for brokerage clients

Under the Central Bank Directive 6681-U, additional requirements are imposed on brokers to comply with the following standards: Risk Coverage Standard 1 (when executing a client order, >0) and Risk Coverage Standard 2 (when the client's portfolio value changes, >0). The Directive allows the use of the "market method" for participants, where Risk Coverage Standard is calculated using the exchange indicative variation margin for portfolio revaluation, and exchange collateral calculated according to the rules for determining the amount of individual clearing collateral required to secure obligations executed using the client's portfolio.

2.6.9.1. Initial margin considering the Client risk level category

In the SPECTRA trading system, at the end-client level (7-digit client code), a flag can be set for the Client risk level category:

- "0" - Not specified (default value)
- "1" -Low-risk client
- "2" - Medium-risk client
- "3" - High-risk client
- "4" - Exceptional-risk client

The risk level category is assigned by the participant through the unified client registration service. If the participant does not provide a category for their client, the default category (value "0") is used. It is possible to change the category during the day; risk level category (medium-risk client, high-risk client, low-risk client, or exceptional-risk client) is applied during the clearing session. When registering online in the Derivatives market, the category specified by the participant online is applied.

Client risk level category is transmitted through gateway streams FORTS_REFDATA_REPL and FORTS_INFO_REPL in table 'investor', in field 'client_risk_level'.

The client's collateral (7-digit client code) is calculated considering the Client risk level category. Risk parameters corresponding to the category are taken into account (categorical market risk rates). If the client's risk level is not specified (value "0"), the basic values of risk rates set by the NCC are used.

2.6.9.2. Risk Coverage Standard calculation in the market method

The Risk Coverage Standard calculation is performed on the broker's side using the output data of the new RMT service (Regulation of margin trading):

- **IM w/o orders** - Exchange initial margin without orders. It is calculated:
  - without considering orders,
  - taking into account the Client risk level category,
  - taking into account the current realized risk (considering the current price),
  - based on the risk parameters at the begginning of the trading day after the expansion of limits.
- **VM** - Indicative variation margin considering the current indicative currency rate (calculated similarly to the current indicative variation margin, including the variation margin on closed positions).
- **NOV** - Net Option Value. Current Net Option Value for a portfolio with 7-digit client code, considering the current rate and current option prices.
- **premium** - Accrued premium for premium options from trades, payable in the next clearing session, converted to RUB. Does not include the financial result of position execution.

The service performs calculations at least once every 10 minutes for all 7-digit client codes, with data transmitted in the stream FORTS_RMT_REPL.

Formulas for Risk Coverage Standard calculation:

```
Risk Coverage Standard 1 = Initial portfolio value + VM + NOV + premium - IM w/o orders
```

```
Risk Coverage Standard 2 = Initial portfolio value + VM + NOV + premium - (IM w/o orders)/2
```

Provided that the initial limit (money_amount) <= the initial portfolio value, it can be stated that the exchange pre-trade (money_free>0) , considering the category, ensures the non-exceedance of Risk Coverage Standard 1 (>0) calculated using the market method in the vast majority of cases.

2.6.9.3. Current operational risk per position

Current operational risk per position - Value calculated according to the rules for determining the amount of individual clearing collateral, reflects the size of collateral if clearing were to occur right now. An additional predictive value that can be used by the participant to make decisions regarding the client's positions. It is also calculated by the RMT service:

- taking into account current risk parameters and market data,
- without considering orders,
- taking into account the Client risk level category.

The service performs calculations at least once every 10 minutes for all 7-digit client codes, with data transmitted in the stream FORTS_RMT_REPL.

2.6.10. Model of participant funds accounting after transition to T+1 settlement cycle

The T+1 settlement cycle assumes trading on day T and settlements on the next day T+1. Upon completion of trading on day T, a clearing session is conducted. In the clearing session, for trades on day T, the variation margin, premium, and commissions and penalties (calculated accruals/deductions) are determined. These amounts form the MTM registers, which store information about the clearing session date, calculation date, type of accrual/deduction, and calculation amount. The participant's balance determined at the end of the clearing session (day T) does not include these calculated amounts.

Based on the results of the clearing session, and taking into consideration balances and calculated accruals/deductions, the participant's trading limit is established to ensure trading. In cases where the participant lacks sufficient funds for collateral, a margin call is issued the next day, which must be fulfilled within the specified day.

On the next day (t+1), during the settlement session at 08:00 PM, the actual fulfillment of obligations according to calculated accruals/deductions takes place, updating the actual balances and resetting the MTM registers. The participant's balance at the time of the settlement session on day t+1 is determined as the beginning of the day balance plus the sum of inputs/outputs for t+1 at the time of the settlement session plus the sum of fulfilled MTM register obligations.

The general concept of changing limit and balance states is presented in the table below.

	Limits (part)	Balances (money_clearing)	Publication
Clearing session 11:50 PM, Day T	• Updated trading **limits** part.money_amount by **PnL** (**MTM register**) for **7кр** and **BF** (where part.is_auto_update_limit =1) • **Limits** part.money_amount set by **BF** (where dealer.limit_tied_money = 1), for **SA** part_sa.money_amount: equal to **balances + PnL** • **PnL=charges+vm+premium**	• **Balances** for SF-BF-7кр and SA do not change by **PnL**, **PnL** stored in MTM register by requirements/obligations • **Balances** for SF-BF-7кр and SA **change by inputs/outputs** during trading session, as well as by the revaluation of currency and asset profiles - net inputs/outputs (**in_out_netto**) • **PnL=charges+vm+premium**	Calculated balances are transmitted to clearing replica and clearing reports (**mon**, **moncb**)
Settlement session 08:00 PM, Day T+1	Unchanged	• Updated **balances** money_clearing.money_end by **PnL** and net inputs/outputs (**in_out_netto**) for the current day at the time of the settlement session • Updated **balances** money_clearing_sa.money_end by **PnL** and net inputs/outputs (**in_out_netto**) for the current day	Calculated balances are transmitted to clearing replica and clearing reports (**monsettl**)

2.6.10.1. Trading limits

Trading limits, as participant funds, are recorded in core registers by sections 7кр-BF and SA. They are transmitted to external streams (FORTS_PART_REPL) part.money_amount, by SA - part_sa.money_amount. The trading limit is not equivalent to the account balance and may not reflect the actual amount of money held by the Participant and its clients.

After the clearing session, the initial value of money_amount (Total rubles and discounted pledges in rubles), includes planned data from mtm registers (planned sums of var. margin, premium, commissions, and penalties to be calculated in settlement session). During the trading day, the money_amount changes with the account inputs/outputs of the current day.

Within money_amount, the value actual_amount_of_base_currency— actual rubles amount—is separately highlighted. It is calculated as actual_amount_of_base_currency=money_amount-money_pledge_amount-sum of MTM registers (charges_mtm+vm_mtm +premium_mtm). This amount can change during the day based on inputs/outputs and during the settlement session based on the MTM registers. The attribute is only applicable for Settlement Accounts and Brokerage Firms, for clients actual_amount_of_base_currency is always 0.

2.6.10.2. Balances on accounts

Balances on accounts, as participant funds, are recorded in back-office registers. They are transmitted to external streams (FORTS_CLR_REPL), by 7кк-BF-SF - table money_clearing, by SA - table money_clearing_sa, field amount_end (in rubles, detailed by currencies in pledge_details.xamount). The account balance includes actual and planned registers.

Actual and planned balances are updated twice a day, after the settlement and clearing sessions. The data in tables money_clearing/money_clearing_sa include the results of the last settlement or clearing session and contain both planned (requirements and obligations) and actual balances and obligations resulting from the settlement and clearing sessions.

Attributes reflecting actual data:

- **date_money_clearing** - Date and time of the last settlement or clearing session.
- **amount_beg** - Balance at the start of the current trading day.
- **amount_end** - Balance at the end of the settlement or clearing session.
- **in_out_netto** - Net inputs/outputs. Includes the sum from the start of the current trading day up to the time of the settlement or clearing session.
- **charges** - Contains information about all actually fulfilled obligations for commission payments and penalties resulting from the settlement session. Includes sums **fee_fut** (Actually calculated exchange and clearing fees for anonymous futures trades) and **fee_opt** (Actually calculated exchange and clearing fees for anonymous option trades). Retained for compatibility as separate detailed information.
- **vm** - Contains information about the total amount of actually paid/withheld variation margin.
- **premium** - Contains information about the total amount of actually paid/withheld premium for options.

Attributes reflecting planned data:

- **settl_datetime** - Planned date for settlement of obligations.
- **amount_end_mtm** - Planned balance including execution of planned accruals/deductions (MTM registers).
- **MTM registers** - Display planned accruals/deductions based on the clearing session results. Include three registers:
  - **charges_mtm** - Register of planned deductions for all commissions (brokerage, clearing, for execution, etc.), penalties. Includes sums **fee_fut_mtm** (Planned obligation for exchange and clearing fees for anonymous futures trades) and **fee_opt_mtm** (Planned obligation for exchange and clearing fees for anonymous option trades).
  - **vm_mtm** - Register of planned accruals/deductions for variation margin.
  - **premium_mtm** - Register of planned accruals/deductions for premium.

Trends of data changes in the money_clearing table at the time of the clearing and settlement sessions:

Date	Period	amount_beg ***Balance at the start of the period***	amount_end ***Balance at the end of the period***	in_out_netto ***Net inputs/outputs***	Planned sums charges_mtm vm_mtm premium_mtm	Actual balances charges vm premium
T	Clearing session	Unchanged **amount_beg**= **amount_end** from the previous clearing session (T-1). This is the balance at the start of the trading day	**amount_end**= amount_beg + charges + vm + premium + in_out_netto (for the entire trading day) The sum of actual registers at the time of the clearing session	The sum of inputs/outputs for the entire day T The register is reset in the clearing, and the value calculated at the time of the clearing session is transmitted in streams and reports	Determined requirements and obligations based on clearing session, values accounted in MTM registers	Unchanged Values calculated at the time of the clearing session are transmitted in streams and reports
T+1	Settlement session	Changed **amount_beg**= **amount_end** from the previous clearing ses-	**amount_end**= amount_beg + charges + vm + premium + in_out_netto	The sum accumulated from the start of T+1 until the settlement session	Reset by actual calculations results	Updated by MTM register values calculated

Date	Period	amount_beg *Balance at the start of the period*	amount_end *Balance at the end of the period*	in_out_netto *Net inputs/outputs*	Planned sums charges_mtm vm_mtm premium_mtm	Actual balances charges vm premium
		sion (T). This is the balance at the start of the trading day				
T+1	Clearing session	Unchanged **amount_beg**= **amount_end** from the previous clearing session (T). This is the balance at the start of the trading day	**amount_end**= amount_beg + charges + vm + premium + in_out_netto (for the entire trading day) The sum of actual registers at the time of clearing session	The sum of inputs/outputs for the entire day T+1 The register is reset in the clearing, and the value calculated at the time of the clearing session is transmitted in streams and reports	Determined requirements and obligations based on clearing session, values accounted in MTM registers	Unchanged Values calculated at the time of the clearing session are transmitted in streams and reports
etc.	...	...	...	...	...	...

2.6.10.3. Withdrawal of funds

Before the settlement session

After the clearing session and before the settlement session, the maximum amount in rubles available for withdrawal is determined as the minimum of the free limit and the maximum of the ruble balance and the ruble balance on the MTM registers.

Maximum amount for withdrawal = MIN (money_free; MAX (ruble balance; ruble balance + mtm registers))

"Ruble balance" - is the actual balance in rubles without taking into account the valuation of pledges and mtm registers.

After the settlement session

After the settlement session, the actual balance of the currency "ruble" becomes equal to the ruble balance before the settlement session plus the mtm registers.

Maximum amount for withdrawal = MIN (money_free; ruble balance).

2.6.10.4. Commissions

With the implementation of the Unified trading session on the Derivatives market, changes are made to monthly/quarterly deductions. Commission deductions are carried out in two stages:

- First, the monthly/quarterly commission is accrued and included in the charges_mtm register during the clearing session on the settlement day (first/last day of the month/quarter).
- The deduction of the commission occurs the next day during the settlement session.

2.7. SMA Login (Sponsored Market Access)

SMA, i.e. Sponsored Market Access is a method of providing trading participants' clients with technical access to the trading and clearing system of the Derivatives Market of Moscow Exchange. Using SMA, a client is able to send requests to the trading participant (i.e. the sponsoring entity) for adding orders directly into the trading system under control and responsibility of the client.

In order to gain access to the trading system, each trading participant's client is granted with a personal ID, i.e. SMA login, using which a client is able to add orders directly to the trading system via gateways PLAZA II, FIX, and TWIME.

In order to control the client's transactions performed via their SMA login, each SMA login is bound to the trading participant's login called MASTER login. Using their MASTER login, a trading participant is able to connect to the trading system, add orders and carry control of performing orders. A trading participant is able to use a single MASTER login for more than one SMA login. Also, each SMA login can be bound to more than one MASTER login. All logins are available via gateways in table 'user' of stream FORTS_REFDATA_REPL, where SMA logins are flagged with 1 in 3rd bit of bitmask 'sma_flags'. All SMA-MASTER login bindings are available via gateways in table 'sma_master' of stream FORTS_REFDATA_REPL.

In order to obtain their SMA login, a trading participant should file an application containing their MASTER login into the Client Centre of Moscow Exchange in order to carry control over transactions performed via SMA login.

The following risk management services are provided by the Moscow Exchange in order to prevent erroneous orders from being added into the trading system:

- Pre-Trade control - some settings to carry additional control over adding orders;

- Cancel On Drop-Copy Disconnect - the service guarantees that all orders added from an SMA login are available in the trading system only when the appropriate MASTER login is connected and active. Every order added via an SMA login contain references to the MASTER login (field 'aspref' of tables 'orders_log' and 'multileg_orders_log');
- UserKillSwitch - forced deactivation of SMA login by a trading participant.

Pre-Trade control service provides some additional restrictions/verifications which can be imposed/applied on adding orders from an SMA login. The verifications can be applied for selected SMA logins, instruments, or client codes, where instruments are:

- <Underlying asset>: <Derivative type>, where <Derivative type> = {Futures, Option, Calendar Spread} - Instrument*
- <Underlying asset>: <Derivative type>, where <Derivative type> = {Futures, Option} - Instrument**

The following verifications can be applied:

Verification number	Verification details	Binding	Unit of measure	Applied
1	Price fluctuation against the current price	SMA login or SMA login x Instrument**	Percent	Immediately
2	Maximum volume of order	SMA login or SMA login x Instrument*	Number of contracts	Immediately
3	Disallow negotiated mode	SMA login	Yes/No	Immediately
4	Maximum volume of order in Russian Ruble	SMA login or SMA login x Instrument*	Russian Ruble	Immediately
5	Maximum volume of all orders per trading day (gross)	SMA login or SMA login x Instrument*	Russian Ruble	Immediately
6	Maximum position (long) in contracts	SMA login x Instrument** x Client code	Number of contracts	Immediately
7	Maximum position (short) in contracts	SMA login x Instrument** x Client code	Number of contracts	Immediately

To apply/cancel the verifications, the gateway methods SetSmaPreTradeCheck and DelSmaPreTradeCheck can be used, appropriately. Information on the already applied verifications is available via gateways in table 'sma_pre_trade_check' of field FORTS_REFDATA_REPL.

Cancel On Drop-Copy Disconnect - the service guarantees that all orders added from an SMA login are available in the trading system only when the appropriate MASTER login is connected and active.

When an order is being added from an SMA login, the service verifies whether there is at least a single MASTER login bound to the given SMA login. If there is no MASTER login detected, the order will be rejected, and the appropriate error message will be issued. If there is at least one bound MASTER login connected and alive, the order will be processed, and a reference to the given MASTER login (MASTER login ID) will be added into field 'aspref'.

The service constantly verifies in real-life mode (similar to that of the Cancel On Disconnect service) if MASTER logins are alive. If there is no transactional activity detected from a MASTER login, the MASTER login will be deactivated. Once there is no any MASTER login remains connected for an SMA login, all orders added via the given SMA login will be cancelled.

All active orders added via an SMA login will be cancelled automatically on technological break in the end of the trading day if the Cancel On Drop-Copy Disconnect service is enabled for these SMA logins.

In order to get access to the Cancel On Drop-Copy Disconnect service, the appropriate application should be filed to the Client Centre of Moscow Exchange.

Method UserKillSwitch allows trading participants manually deactivate (and activate again) an SMA login, with the supported possibility to automatically cancel all active orders of the given SMA login. Once deactivated, an SMA login is no more able to perform trading transactions till the end of the trading day. The SMA login will be activated again after the trading system restart due to technological break, or due to a failure.

2.8. Separate entities of Clearing Member and Trading Member

Effective SPECTRA version 6.2, the Derivatives Market rolls out a new model, with Trading Member and Clearing Member represented by two separate entities, in order to allow clients to have access for trading without necessity to be Clearing Member. On the other hand, to perform their obligations regarding the trades performed on the Derivatives Market, a Clearing Member may not be a Trading Member. Therefore, a member in the Trading System SPECTRA may be of one of the following categories:

- Clearing Member (CM). Clearing Member may serve one or more Trading Members acting as a counterparty for the trades performed by these Trading Members.
- Trading Member (TM). Trading Member is eligible to perform trades on auction. However, obligations and requirements regarding the trades performed by the Trading Member will be fulfilled by a Clearing Member that serves the given Trading member.
- Clearing Member + Trading Member (CM+TM). Before SPECTRA version 6.2), all members on the Derivatives Market belong to this category. The members are eligible to perform trades on auction, and act as NCC counterparties for the performed trades. For these members, the trading and clearing procedures remain the same as they have been before.

2.8.1. Who belongs to whom: a view inside SPECTRA

Inside SPECTRA, Clearing Member is represented by a Clearing FIrm; the Clearing Firm itself may act either as Clearing Member, or as an entity representing both Clearing Member + Trading Member.

A Trading member is represented by Brokerage firms of a Clearing Firm which belongs to the appropriate Clearing Member. Also:

- a single Trading Member may be served by several Brokerage Firms, each belonging to the same Clearing Firm (Clearing Member);
- a single Trading Member may be served by several Brokerage Firms belonging to different Clearing Firms (different Clearing Members);

For a Brokerage Firm, Trading Member may be either of the same entity as Clearing Member (CM=TM), or of different entities (CM!=TM).

If a Clearing Firm acts as Clearing Member, it may have only those Brokerage Firms where Trading Member and Clearing Member are of different entities (CM!=TM). If a Clearing Firm act as Clearing Member + Trading Member (CM+TM), it may have Brokerage Firms where Trading Members either of the same entity as Clearing Member, or of a different entity.

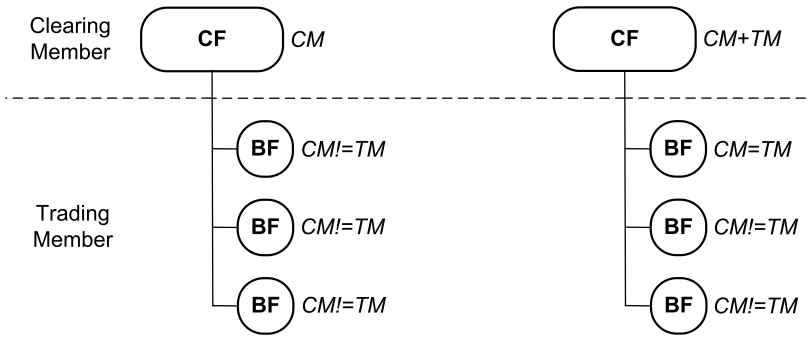

Figure 2. Hierarchy

2.8.2. Separation of Members' rights

Following the separation of Clearing and Trading Member entities, the Member's rights are also separated for the Clearing Member and Trading Member. Therefore, different Members are granted with different rights to view data in and perform operations.

The basic rights granted to Trading Member (and not available to Clearing Members) include possibility to add/cancel exchange orders and perform trades based on the added orders. Clearing Member, on their side, is also able to perform actions with orders indirectly via submitting requests to NCC, while NCC will add orders acting on behalf of the Clearing Member, and cancel other orders (for details see 2.7.3. Reconciliation of unfulfilled obligations).

The basic rights granted to Clearing Member (and not available to Trading Members) include possibility to manage collateral (i.e. withdraw collateral, and transfer collateral between settlement accounts), and also manage risks for Trading Members (when risk occurs for trades performed by the Trading Member), where by risk management we mean setting limits for the Trading Member, and ability to reduce (troubled) positions of the Trading Member.

The Members' rights are separated for different the SPECTRA logins. Therefore, the following rights are granted to the logins:

- 'Login CM' - login granted with rights of Clearing Member.
- 'Login TM' - login granted with rights of Trading Member.
- 'Login CM+TM' - login granted with rights of both Clearing and Trading Members.

Below there is the list of rights for 'Login CM':

Transaction	Available commands
Set limits for BF	ChangeBFMoney
Set BF trading limits for BF	ChangeBFLimit
Transfer of funds between two BFs of the same SA	ExchangeBFMoney
Set risk parameters for BF	ChangeBFParametersNextSession
Transfer positions between two BFs (only CF login)	TransferClientPosition
Submit requests to NCC for performing trades with Trading Member. Technically, the request will be submitted as a standard market order with the special flag.	AddOrder
Cancel requests to NCC for performing trades with Trading Member. Technically, the request will be submitted as cancelling a standard market order with the special flag.	DelOrder
Change requests to NCC for performing trades with Trading Member. The request is technically implemented as changing an order with special flag.	MoveOrder

Transaction	Available commands
Request to NCC for collateral sufficiency verification of BF.	DelOrdersByBFLimit
Add/cancel orders to early Option exercise.	OptChangeExpiration
Submission of request for early exercise of options, for cancellation of automatic exercise of options.	FuturesExecutionRequest
Manage SMA logins.	SetSmaPreTradeCheck; DelSmaPreTradeCheck; UserKillSwitch
Recalculate central strike.	OptRecalcCS

Below there is the list of rights for 'Login TM':

Transaction	Available commands
Add trading orders.	AddOrder; IcebergAddOrder
Cancel trading orders.	DelOrder; DelUserOrders; IcebergDelOrder
Change trading orders.	MoveOrder; IcebergMoveOrder
Add/cancel orders to early Option exercise.	OptChangeExpiration
Submission of request for early exercise of options, for cancellation of automatic exercise of options..	FuturesExecutionRequest
Manage SMA logins.	SetSmaPreTradeCheck; DelSmaPreTradeCheck; UserKillSwitch
Recalculate central strike.	OptRecalcCS

'Login CM+TM' is granted with rights of both Clearing and Trading Members, excluding the rights to submit requests to NCC which are exclusively reserved for 'Login CM'.

Clearing Member may obtain logins of the following levels:

- Clearing Firm.
- Brokerage Firm.
- Client.

Trading Member may obtain logins of the following levels:

- Brokerage Firm.
- Client.

Depending on the login level, its rights may vary as follows:

- For BF with CM=TM, both Clearing Firm and Brokerage Firm level logins will be granted with 'Login CM+TM' rights;
- For BF with CM!=TM, Clearing Firm level login will be granted with 'Login CM' rights;
- For BF with CM!=TM, Brokerage Firm level login will be granted with rights in accordance with flag ['Login CM' | 'Login TM'];
- For any Brokerage Firm, a client level login is granted with 'Login TM' rights.

If Clearing Member is not of the same entity as Trading Member, they may manage their clients (client accounts) in accordance with two different models. By managing here we mean ability to view various data regarding the clients (assets, limits, individual risk settings, etc), as well as ability to set limits, prohibitions, expiration rules, etc. The two models of managing clients are as follows:

- Clients are managed by Trading Member. The Trading Member manages client accounts belonging to their Brokerage Firms (default model). According to this model, Clearing Member does not have access to data on Trading Members' clients, i.e. assets, limits and individual risk settings for each client; the client management commands are also unavailable to the Clearing Member.
- Clients are managed by Clearing Member. The client accounts belonging to the Trading Member's Brokerage Firms are managed by the Clearing Member via an agency contract.

The appropriate model can be selected by adding a special flag for the BF, i.e. 'Clients are managed by TM'/'Clients are managed by CM'.

Below there is a couple of examples of each model usage:

1. In order to provide clients with access for trading, a regional brokerage firm signs an agreement with a Clearing Member to be served by the clearing broker model. To perform their obligation against the regional broker, the Clearing Member establishes a Brokerage Firm and register the regional broker as a Trading Member at the Exchange. According to that, it is the regional broker's clients who will trade at the Exchange, therefore, the broker itself wishes to manage these clients; that is why the model 'Clients are managed by Trading Member' will be applicable here.

2. In order to provide clients with access for trading, a non-resident company submit a request to NCC to become a Clearing Member. After that, this non-resident company signs an agency contract for managing their clients with a Brokerage Firm (either a Russian subsidiary of the non-resident company, or a large CM+TM company). According to that, it is the non-resident company's clients who will trade at the Exchange, and the non-resident wishes to manage them by itself; that is why the model 'Clients are managed by Clearing Member' will be applicable here.

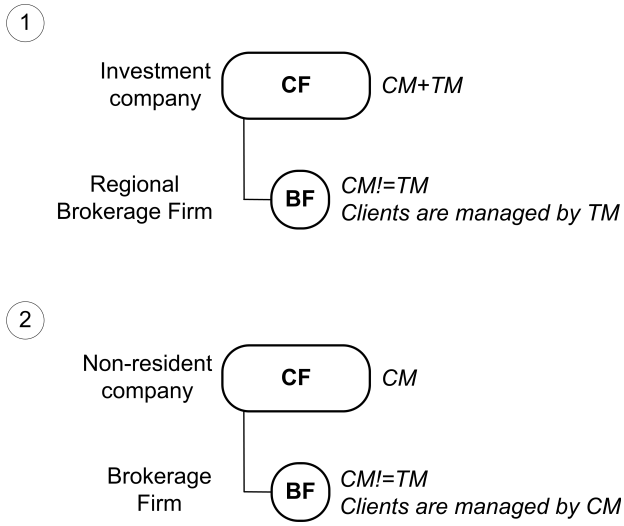

Figure 3. Examples

Below there is a list of rights for managing client accounts. Please note that these rights will be granted to the logins with their login types ('Login CM'/'Login TM') corresponding with the appropriate BF model ('Clients are managed by Clearing Member'/'Clients are managed by Trading Member').

Transaction	Available commands, tables
Set limits for client accounts	ChangeClientMoney
Set risk settings for client accounts	ChangeClientParameters; ChangeClientParameter-sNextSession; ChangeBFClientDefaultParameter-sNextSession
Set transactional prohibitions for client accounts.	FutChangeClientProhibit; OptChangeClientProhibit
Set risk settings for Options.	OptChangeRiskParametersNextSession
Transfer client positions. Available only for BF logins with type 'Login CM'.	TransferClientPosition
View data on client accounts: amount of collateral and individual risk settings.	FORTS_PART_REPL.part; FORTS_CLR_REPL.money_clearing; FORTS_CLR_REPL.pledge_details

The rights listed above are also available to login type 'Login CM+TM'.

2.8.3. Management of a Trading Member obligations by a Clearing Member

A Clearing Member is not able to add and cancel orders directly in the trading system. However, a Clearing Member is able to submit requests to NCC in order to manage unfulfilled obligations of of Trading Member against Clearing Member, also in case of collateral insufficiency of the Trading Member. The request may be submitted by Clearing member for any of the client accounts of the Trading Member. According to this request, NCC will add either an anonymous, or a negotiated order, based on on which trades will be performed to fulfill the obligations.

Technically, the request will be submitted as a standard market order with the special flag 'Request to NCC for performing trades with Trading Member'. The request may be submitted either as anonymous, or a negotiated order. Each order with the flag 'Request to NCC...' have bit NCCRequest (0x4000000000) active in its bitmask. The same bit will be active also in trades performed based on these requests.

In order to avoid position insufficiency, a Clearing Member is able to cancel orders added by a Trading Member. To do this, the Clearing Member may execute command 'DelOrdersByBFLimit' with 'MassCancelRequestType=Z' to NCC for collateral verification of Brokerage Firms under the Trading Members, that are served by the Clearing Member; if a Brokerage Firm experiences negative free cash limit, all active orders on all client accounts served by the given Brokerage Firm will be cancelled. Each of the cancelled orders will have bit NCCBulkDeleteOperation (0x8000000000) active in its bitmask.

2.9. Synthetic matching

Synthetic matching - the trades formation on the basis of orders from different order-books (order-books of different instruments). The purpose of synthetic matching is to increase the instruments liquidity by combining several order-books. For example, synthetic matching

allows match calendar spread orders not only with a counter order inside this instrument order-book, but also with orders from order-books of its futures-legs. Thus, the calendar spread order takes into account the counter interest from other order-books of their legs.

2.9.1. Synthetic orders

In synthetic matching, three orders for different trading instruments can be matched to a trade if the prices of these instruments are linked by a certain ratio. For example, the price of a calendar spread is equal to the difference between the price of the far leg and the price of the near leg. Then buy order on the calendar spread RTS-9.18-12.18 at the price 1000 (participant "A"), buy order on the futures RTS-9.18 at the price 114000 (participant "B") and sell order on the futures RTS-12.18 at the price 115,000 (participant "C") can be performed simultaneously. As a result, participant "B" will get a long futures position RTS-9.18 at a price 114000. Participant "C" will get a short futures position RTS-12.18 at a price 115000. Participant "A" will get two positions: a short futures position RTS-9.18 at a price 114000 and a long futures position RTS-12.18 at a price 115000, at what their prices are related by the ratio 115000 - 114000 = 1000. Thus, all three orders can be satisfied correctly.

On the Moscow Exchange, trades are performed with a central counterparty (NCC). In this example, three trades will be performed:

- for calendar spread RTS-9.18-12.18 between participant A and NCC
- for futures RTS-9.18 between participant B and NCC
- for futures RTS-12.18 between participant C and NCC

In the process of synthetic matching, orders are automatically generated submitted on behalf of the NCC in the trading system core. Such orders are called synthetic. Synthetic order - an order created by the core during synthetic matching when the conditions for orders matching are met. A synthetic order is a real order submitted by a central counterparty and appears in trades generated during synthetic matching. In anonymous and user's orders streams, synthetic orders have the special attribute 'Synthetic' (0x200000000000) in 'xstatus' field.

In this example, the following synthetic orders are generated on behalf of NCC:

- sell order for calendar spread RTS-9.18-12.18 at a price 1000 (opposite order to participant A order)
- sell order for futures RTS-9.18 at a price 114000 (opposite order to participant B order)
- buy order for futures RTS-12.18 at a price 115000 (opposite order to participant C order)

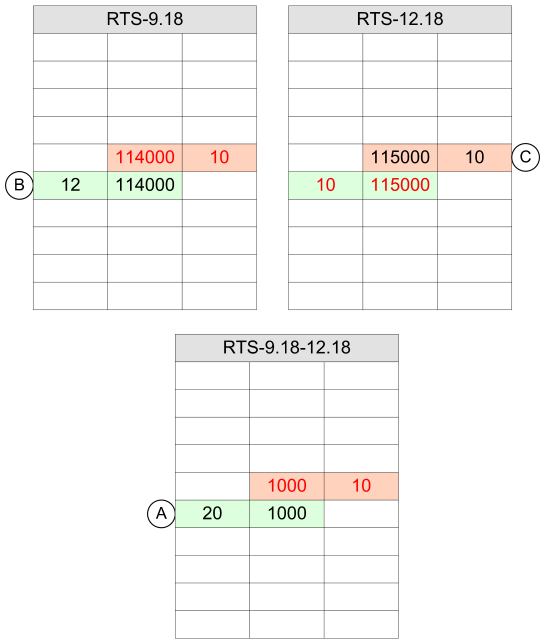

Figure 4. Calendar spread order-book

A synthetic order is generated by the trading system on the basis of two real orders submitted by participants for two other instruments. In our example, the system generated a synthetic sell order for the calendar spread RTS-9.18-12.18 at a price 1000 based on a buy order for the RTS-9.18 futures at a price 114000 (from participant “B”) and a sell order for the RTS-12.18 futures on price 115000 (from participant "C").

Two main scenarios of synthetic matching:

Case 1: The orders for futures generate a synthetic order for calendar spread.

Example (see Figure 4, “Calendar spread order-book”):

- In order-book for calendar spread RTS-9.18-12.18, a buy order is received with amount 20 at price 1000 from participant "A".
- Participant "B" sets buy order for RTS-9.18 (order-book for near leg) with amount 12 at price 114000.

- Participant "C" sets sell order for RTS-12.18 (order-book for far leg) with amount 10 at price 115000 (incoming active order). At this point, the matching begins.
- Based on orders for near and far legs, a sell synthetic order for calendar spread RTS-9.18-12.18 appears with amount 10 (the minimum amount of three orders participating in the matching) at price 1000 (115000-114000: i.e. price of the far leg minus price of the middle leg) in order-book for calendar spread. A synthetic order is set on behalf of the NCC and is executed to the trade with an order for a calendar spread from participant "A".
- A sell synthetic order for RTS-9.18 with volume 10 and at price 114000 on behalf of the NCC appears in order-book for the near leg, and is matched with the order from participant "B" to the trade. A buy synthetic order for RTS-12.18 with volume 10 and at price 115000 on behalf of the NCC appears in order-book for the far leg, and is matched with the order from participant "C" to the trade.
- Thus, three trades are formed: for near futures (counterparties: “B” and NCC) at price 114000, for far futures (counterparties: “C” and NCC) at price 115000 and for calendar spread (counterparties: “A” and NCC) at price 1000. Also, two technical trades are formed, they display the movement for legs of calendar spread. For both technical trades, counterparties are “A” and NCC.
- Two orders left: for RTS-9.18-12.18 with volume 10 and for RTS-9.18 with volume 2.

Case 2: Calendar spread order and futures-leg order generate a synthetic order for the second leg of this calendar spread.

Example (see Figure 4, “Calendar spread order-book”):

- Participant "B" sets buy order for RTS-9.18 (order-book for near leg) with amount 12 at price 114000.
- In order-book for calendar spread RTS-9.18-12.18, a buy order is received with amount 20 at price 1000 from participant "A".
- Participant "C" sets sell order for RTS-12.18 (order-book for far leg) with amount 10 at price 115000 (incoming active order). At this point, the matching begins.
- Based on orders for near leg and calendar spread, a buy synthetic order for far leg RTS-12.18 appears with amount 10 at price 115000. A synthetic order is set on behalf of the NCC and is executed to the trade with an order from participant "C".
- A sell synthetic order for RTS-9.18 with volume 10 and at price 114000 on behalf of the NCC appears in order-book for the near leg, and is matched with the order from participant "B" to the trade. A sell synthetic order for RTS-9.18-12.18 with volume 10 and at price 1000 on behalf of the NCC appears in order-book for the calendar spread, and is matched with the order for calendar spread from participant "A" to the trade.
- Thus, three trades are formed: for near futures (counterparties: “B” and NCC) at price 114000, for far futures (counterparties: “C” and NCC) at price 115000 and for calendar spread (counterparties: “A” and NCC) at price 1000. Also, two technical trades are formed, they display the movement for legs of calendar spread. For both technical trades, counterparties are “A” and NCC.
- Two orders left: for RTS-9.18-12.18 with volume 10 and for RTS-9.18 with volume 2.

Six synthetic matching variant are possible depending on the incoming active order:

Active order	Counter real order	Formation of a counter (passive) synthetic order	Counter passive synthetic order price
Buy near futures	Sell near futures	Sell far futures + Buy calendar spread	Far futures price - Calendar spread price
Sell near futures	Buy near futures	Buy far futures + Sell calendar spread	Far futures price - Calendar spread price
Buy far futures	Sell far futures	Sell near futures + Sell calendar spread	Near futures price + Calendar spread price
Sell far futures	Buy far futures	Buy near futures + Buy calendar spread	Near futures price + Calendar spread price
Buy calendar spread	Sell calendar spread	Buy near futures + Sell far futures	Far futures - Near futures
Sell calendar spread	Buy calendar spread	Sell near futures + Buy far futures	Far futures price - Near futures price

The first priority of matching is price. Regardless of the order type (synthetic or real), the active one matches with the passive one with the best price. If the prices of passive synthetic and passive real orders are the same, then first the active order is matched with the one received earlier. Since a calendar spread has two legs adding at different times, the time of such spread is determined by the time of the last leg received. In each order-book (calendar spread, near and far futures), the trade price is determined by the passive order price, as in the current implementation.

2.9.2. Synthetic liquidity in aggregated order-books

In aggregated order-books, by default, the depth of five price levels is generated, formed by indicative synthetic orders. An indicative synthetic order is a “virtual order” used to form an aggregated order-book reflecting the available synthetic liquidity. The matching of such an order leads to the trade performing by synthetic matching.

Example

- There are three empty order-books RTS-9.18, RTS-12.18, RTS-9.18-12.18.
- In order-book for RTS-9.18 (order-book 1), participant "A" adds a buy order with amount 12 contracts at price 114000. Then in order-book for RTS-12.18 (order-book 2), participant "B" adds a sell order with amount 10 contracts at price 115000.
- As a result, in order-book for RTS-9.18-12.18 calendar spread (order-book 3), a sell indicative synthetic order appears at price 115,000 - 114,000 = 1000 with amount 10 contracts, formed from orders added by participants "A" and "B".
- Participant "C" can buy RTS-9.18-12.18 calendar spread in order-book 3 at price 1000 with amount 10 contracts.
- Based on orders from participants "A" and "B" (in order-books 1 and 2), a sell synthetic order for RTS-9.18-12.18 calendar spread with amount 10 at price 1000 appears in order-book 3. A synthetic order is set on behalf of the NCC and is executed to the trade with an order on calendar spread from participant "C".
- A sell synthetic order for RTS-9.18 with volume 10 and at price 114000 on behalf of the NCC appears in order-book 1, and is matched with the order from participant "A" to the trade. A buy synthetic order for RTS-12.18 with volume 10 and at price 115000 on behalf of the NCC appears in order-book 2, and is matched with the order from participant "B" to the trade.
- Thus, three trades are formed: for near futures (counterparties: “A” and NCC) at price 114000, for far futures (counterparties: “B” and NCC) at price 115000 and for calendar spread (counterparties: “C” and NCC) at price 1000. Also, two technical trades are formed, they display the movement for legs of calendar spread. For both technical trades, counterparties are “A” and NCC.

Synthetic liquidity is broadcast in the aggregates stream (FORTS_AGGR##_REPL) together with liquidity on real orders. If inside one price level there are both real orders and indicative synthetic volumes, then in addition to the total volume ('volume' field), a synthetic volume ('synth_volume' field) is broadcast in a separate field.

Consider an example where synthetic liquidity is added to a standard aggregated order-book. There are aggregated order-books for calendar spread and its legs with natural liquidity.

Si-6.19		
	65355	19
	65354	25
	65353	5
	65352	2
	65351	38
2	65349	
7	65348	
27	65347	
7	65346	
92	65345	

Si-9.19		
	66218	20
	66212	3
	66210	3
	66202	1
	66200	5
1	66198	
1	66197	
4	66191	
4	66190	
1	66185	

Si-6.19-9.19		
	860	10
10	831	

Figure 5. Aggregated order-books for calendar spread

The same order-books, but taking into account synthetic liquidity, look as follows.

Si-6.19				
synth_ volume	volume	price	volume	synth_ volume
		65381	1	1
		65379	3	3
		65371	1	1
		65369	5	5
		65355	19	
		65354	25	
		65353	5	
		65352	2	
		65351	38	
	2	65349		
	7	65348		
	27	65347		
	7	65346		
	92	65345		
1	1	65338		
1	1	65337		
4	4	65331		
4	4	65330		

Si-9.19				
synth_ volume	volume	price	volume	synth_ volume
		66218	20	
		66212	3	
		66211	10	10
		66210	3	
		66202	1	
		66200	5	
	1	66198		
	1	66197		
	4	66191		
	4	66190		
	1	66185		
2	2	66180		
7	7	66179		
1	1	66178		

Si-6.19-9.19				
synth_ volume	volume	price	volume	synth_ volume
		871	20	20
		865	3	3
		862	3	3
		860	10	
		854	1	1
		852	3	3
		851	2	2
1	1	847		
1	1	846		
4	4	840		
4	4	839		
1	1	834		
	10	831		

Figure 6. Aggregated order-books with synthetic liquidity

These order-books show all the calculated synthetic liquidity without taking into account the limit on the number of levels of aggregated order-book of synthetic liquidity. Price levels with synthetic liquidity are highlighted in red.

Synthetic liquidity for calendar spread significantly narrowed the price spread and made the order-book of calendar spread more attractive for traders. This is the purpose of synthetic matching - to show traders the best available price and potentially greater volume of execution of their orders at the best average execution price.

In the order-book for the far leg, the situation has not changed so much. Although here, if a client adds a buy order with volume 15 contracts at price 66212, he will make trades not only with orders for the same instrument (levels 5 on 66200, 1 on 66202, 3 on 66210, which execute 9 contracts from 15) , but also he will make a trade in synthetic matching for the 6 remain contracts at price 66211. This synthetic matching uses 6 contracts for the sale of Si-6.19-9.19 at 860 and 6 contracts for the sale of Si-6.19 at 65351.

In the order-book for the near leg, synthetic liquidity remains in the background, since the near leg is the most liquid instrument with the lowest price spread.

Synthetic liquidity in aggregated order-books is updated with the frequency of updating aggregated order-books themselves. The frequency of updating aggregated order-books is lower than the frequency of trading events in the trading system, so the synthetic liquidity in the order-book is not updated for each order or trade. A participant who wants to evaluate the full depth of synthetic liquidity (more than 5 price levels) and its change at each trading event (transaction) must independently calculate the available synthetic liquidity based on information in the public orders_log.

In public orders_log, synthetic orders appear only at the time of synthetic matching in the amount necessary to conclude a transaction, i.e. synthetic orders are fully executed within the transaction in which they are generated. Therefore, if the user makes the order-books by orders_log (without synthetics) and checks it, for example, with the data broadcasted in FORTS_AGGR##_REPL, then these order-books will differ - the order-book from FORTS_AGGR##_REPL may contains prices that are “not visible” in the order-book made by orders_log.

In the stream FORTS_COMMON_REPL, the fields with the best prices and volumes at the best price are calculated taking into account synthetic liquidity, wherein the old fields (such as 'best_buy', 'best_sell', 'xamount_buy', 'xamount_sell', etc.) contain the sum of natural and synthetic liquidity, and new fields (with postfix '_native') contain natural liquidity without taking into account synthetic liquidity.

2.10. Settlement trades

Settlement trades are concluded by NCC on behalf of and at for settlement account (SA) of the Clearing Member.

If a Clearing Member fails to fulfill its obligations in time, then NCC considers such a participant to be a Defaulting Clearing Member (DCM). NCC, on behalf of and for settlement account of the Defaulting Clearing Member, concludes trades that lead to a reduction in position and fulfillment of obligations. The purpose of such trades is to eliminate the insufficient of collateral for obligations with a maturing and non-maturing execution date. The procedure is described in more detail in the Clearing Rules in the "Procedure for Margin Calls and default funds Margin Calls" section.

NCC concludes settlement trade on behalf of and according to the pre-agreed settlement account of the Non-defaulting Clearing Member, if the settlement trade with Defaulting Clearing Member cannot be concluded via the order book. The procedure is described in more detail in the Clearing Rules in the "Procedure for the closing and/or balancing trades execution" section. Commissions (fines) are not charged for trades with Non-defaulting Clearing Member.

2.10.1. Reasons for settlement trades

The flag of settlement trades is broadcast in the gateway in the tables of yours orders 'orders_log' and 'multileg_orders_log' ('reason' field) and trades 'user_deal' and 'user_multileg_deal' ('reason_buy' and 'reason_sell' fields), and also in reports: 'f04', 'f04cl', 'f04_trade', 'f04cl_trade', 'o04', 'o04cl', 'o04_trade', 'o04cl_trade'.

Field value 'reason/ reason_buy/ reason_sell'	Reason	Member
0	Common trade.	Clearing Member
4	Balancing Derivatives Contracts entered into with the Non-defaulting Clearing Member without submitting orders.	Non-defaulting Clearing Member
6	Closing Derivatives Contracts entered into under the cross-default procedure.	Defaulting Clearing Member
7	Closing Derivatives Contracts entered into upon non-fulfillment of the Margin Call.	Defaulting Clearing Member
8	Closing Derivatives Contracts entered into in into upon non-fulfillment of the Delivery Obligation on the deliverable Derivatives Contracts for precious metals.	Defaulting Clearing Member
11	Balancing trade for the obligations of the Defaulting Clearing Member.	Non-defaulting Clearing Member
100	Other	Defaulting Clearing Member

In the 'f04', 'f04cl', 'f04_trade', 'f04cl_trade', 'o04', 'o04cl', 'o04_trade', 'o04cl_trade' reports, the reason for settlement trades is in the 'type_buy' and 'type_sell' fields.

Futures trades reports 'f04', 'f04cl', 'f04_trade', 'f04cl_trade':

- "3" - for balancing Derivatives contracts entered into with the Non-defaulting Clearing Member without submitting orders;
- "21" - for closing Derivatives contracts entered into under the cross-default procedure;
- "22" - for closing Derivatives contracts entered into upon non-fulfillment of the Margin Call;
- "23" - for closing Derivatives contracts entered into in into upon non-fulfillment of the Delivery Obligation on the deliverable Derivatives contracts for precious metals;
- "29" - for balancing trades for obligations of the Defaulting Clearing Member.

Options trades reports 'o04cl', 'o04_trade', 'o04cl_trade':

- "3" - for balancing Derivatives contracts entered into with the Non-defaulting Clearing Member without submitting orders;
- "6" - for closing Derivatives contracts entered into under the cross-default procedure;
- "7" - for closing Derivatives contracts entered into upon non-fulfillment of the Margin Call;
- "9" - for balancing trades for obligations of the Defaulting Clearing Member.

2.10.2. Fines and fees

For settlement trades, a penalty is charged from the Defaulting Clearing Member instead of a commission. The amount of the penalty for concluding the closing Derivatives contracts is equal to the sum of 5 exchange fees established by Moscow Exchange and 5 clearing commissions from the amount of the closing Derivatives contracts.The penalty is calculated for each settlement trade and is accounted for the 7-digit section of the Clearing Member specified in the settlement trade.

Information about penalties is broadcast in the gateway in the 'penalty' field of the 'part' table in the FORTS_PART_REPL stream(in total for 7-digit section), and also in the 'penalty' table of the FORTS_FEE_REPL stream (in the context of trades).

Fines and commissions for concluding settlement trades on behalf of the Non-Defaulting Clearing Member (balancing Derivatives contracts entered into with the Non-defaulting Clearing Member without submitting orders) are not charged.

Fines for the conclusion of closing Derivatives contracts with Defaulting Clearing Member are not charged:

- if the Clearing Member is under the Liquidation Netting procedure;
- if the Clearing Member is in the status "Suspension of clearing services for the Clearing Member due to cancellation of the license to carry out professional activities in the Securities Market".

Information on these locks is broadcast in the gateway in the 'clearing_members' table of the FORTS_REFDATA_REPL stream.

Settlement trades for which no fines were charged due to cancellation of the license to carry out professional activities in the Securities Market are marked in the tables of trades in the 'xstatus_sell' and 'xstatus_buy' fields with a special sign:

- DontFineRF (0x80000000000000) - No penalty for settlement trades.

Information on the amount of the fine is included as a new type of payment in 'pay' reports on the date of debiting. Fines are recorded in the MTM register as part of the charges value.. Reports:

- ·chargesXXYY.csv - for the current trading day;
- ·chargesclXXYY.csv - for the current trading day;
- payXXYY.csv - for the next trading day;
- payclXXYY.csv - for the next trading day.

2.11. Options on spot assets

The peculiarity of options on spot assets is that the underlying asset for them is the assets of the Securities and FX markets (shares, indexes, currencies, etc.), and not futures on them. Their feature is their underlying asset (UA) which is a share, not the futures. That is, logically there appears a direct connection of an option series (OS) directly with the UA, bypassing the futures. However, OS for such options will be technically recorded as a special futures (collateral). This instrument is already available at Trading System Spectra and is used to transfer asset profiles to sub-accounts of the Unified Pool settlement codes. Thus, the hierarchical structure of the instruments does not change and, with respect to equity options, remains fully identical to options on options.

An option on spot asset is defined by the following attributes:

- Option series of a spot asset option is linked to a collateralized futures instrument: the 'underlying_id' field of the corresponding record from the 'FORTS_REFDATA_REPL:sess_option_series' table contains futures identifier, which can be found in the 'FORTS_REFDATA_REPL:fut_sess_contents' table by the 'isin_id' field; in the received record the bit '0x40000' (collateral instrument attribute) should be set in the 'signs' field;
- The underlying asset for the collateralized futures instrument shall be a spot asset (e.g. a stock).

2.11.1. Calculation of free cash for options on spot assets

Options on spot assets have a value and (at the request of participants) will be used as portfolio collateral and will also affect the amount of free cash (FreeMoney). The 'FreeMoney' adjustment will be available through a new 'NetOptionValue' (NOV) parameter, which will be calculated at the next clearing session as the sum of the products of the book values and the volumes of the corresponding option positions in the portfolio, taking into account the sign:

$$NetOptionValue = vol_i * RC_i * \frac{MinStepPrice_i}{MinStep_i}, \text{ where}$$

- vol_i – position volume in the ith option contract by the end of the current clearing session;
- RC_i – estimated price of the ith option contract by the end of the current clearing session.

NetOptionValue ('net_option_value' field of the 'part' and 'part_sa' tables in the 'FORTS_PART_REPL' stream) is determined for each position recording level (7CC, BF, SA). The NOV value is always equal to zero for futures and futures-style options on futures.

2.11.2. Amount of the premium payable/receivable at the nearest clearing session

Since there is no variation margin for options on spot assets, VM values generated by TS will always be zero for such instruments. As such, a new premium indicator appears (the 'premium' field in the 'opt_vm' table of the 'FORTS_VM_REPL' stream). The calculated value **does not include** the financial result of exercising the option position on the expiration date of the spot assets options. This value is indicative calculated for information purposes only. As far as settlements can be made not only in RUB, the premium in the currency of settlement is transmitted in a separate 'premium_in_settl_currency' field of the 'opt_vm' table in the 'FORTS_VM_REPL' stream.

When accruing or debiting a currency premium, the trading limit will change for 7CC and BF (with the option of free limit management). The amount of the change in the limit is equal to the volume of the premium in foreign currency, converted into rubles at the exchange rate fixed at the time of clearing.

2.11.3. Exercise of options on spot assets

Since the options on spot assets to be introduced are European and cash-settled, only options that are 'in the money' will be automatically exercised and no exercise/withdrawal requests will be accepted for such options.

As stated above, the underlying asset price obtained from the 'LCLOSEPRICE' field of 'SECURITIES' table of the Equity Market trading gateway is used to determine the strike price of the option on the expiration date. This price is fixed in the collateral instrument dimension in the 'underlying_price' field of the 'option_series_settl' table of the FORTS_CLR_REPL stream. On other days, this field contains the collateral price determined at the time of clearing according to the settlement price methodology.

The premium for option contracts with settlements in rubles, received/written off in the clearing session, is broadcast in the 'premium' field in the 'opt_pos' and 'opt_pos_sa' tables of the 'FORTS_CLR_REPL' stream. The broadcast value includes the financial result of exercising the option position on the expiration date of the equity options. For option contracts with settlements in the currency, premium received/written off in the clearing session, is broadcast in the 'premium_in_settl_currency ' field in the 'opt_pos' and 'opt_pos_sa' tables of the 'FORTS_CLR_REPL' stream. This field is filled with zero for settlements in ruble Similarly, the ruble premium ('premium' field) is zero for settlements in a currency.

2.11.4. Margining of options on spot assets

Since the underlying asset of options on spot assets is a collateral futures (in physical terms, a spot asset), its risk parameters, unlike real futures traded, contain nothing but market risk. Therefore, all of the necessary values (risk-free interest rate, interest rate mismatch and dividend risk rates) are taken into account directly when setting margin requirements for the options themselves. Expiration risks are **not** calculated for European equity-style options because the contracts are cash-settled, not deliverable.

A new level of IM - an option series - appears in the margining hierarchy of the collateral calculation system. Previously, the minimum level was the futures and its risks were netted against the risks of all linked option series.

There are additional fields for describing underlying asset and option series, the values from which will be used in option pricing formulas. To calculate the theoretical prices of options, two pricing models are used: Black-Scholes and Bachelier. In normal operation, the Bachelier model is not applied to equity-style options, since negative prices for such UAs are not assumed. The Black-Scholes model with discrete dividend payout is used to calculate theoretical prices for equity options. As dividends are divided into forecasted and declared, the cash flow contains two types of information. The first type includes the amount of expected discounted dividends and the second type includes the amount of declared dividends.

2.11.5. Prohibitions on options on spot assets

Due to the fact that all equity option series (within the same UA) will be started not on different futures, but on a single collateral one, it is impossible to prohibit trading in groups of OS, since in the current release it is possible to set certain restrictions only on all options within one futures at once. Therefore, it is possible to set prohibitions on all equity options on the same UA at once - a prohibition on options with the isin of the collateral futures (opt_sess_contents.fut_isin_id). Or complete prohibition on options - group_mask = 0x80000000 (options).

2.11.6. Margining of premium options without initial margin and NOV at the 7CC level

The system provides a simplified method for calculating free cash for premium options by excluding initial margin and NetOptionValue (NOV), in order to simplify premium option accounting in members' trading systems and make it more transparent for their clients. This

feature is implemented by enabling a special mode for premium option position accounting at the client level (the feature is not supported at the Brokerage Firm or Settlement account level). For 7CC in this mode, initial margin for premium option positions and orders, as well as NOV, are no longer calculated, and opening negative positions in premium options is prohibited. Instead of initial margin, a specially calculated value - "Premium reserve" (part: premium_blocked) - is used. This reserve contains the accumulated financial result of daily premium option turnover, a reserve for orders, and currency risk associated with orders and trades. This reserve is taken into account when calculating free cash: positive values reduce the Trading limit, while negative values increase it.

The special mode for premium option position accounting can be enabled for a specific 7CC (the mode status is published in the gateway in the investor:not_use_nov field) or for all 7CC of a Brokerage Firm (published in the gateway in the dealer:not_use_nov_client_default field). The special mode is applied during the clearing session. The special mode can be enabled for a 7CC only if the member has no negative positions in premium options and the number of contracts in sell orders does not exceed the position volume. Management (enable/disable) of the special mode is performed using the gateway commands ChangeClientNotUseNovMode and ChangeBFClientDefaultNotUseNovMode.

2.12. Trading on weekends

In version 8.3, trading on the Derivatives market on weekends has been implemented in the SPECTRA system. Organized trading on weekends is conducted within the Monday trading session by adding additional weekend trading sessions. Each weekend day has its own additional weekend trading session. After the additional weekend trading session, no clearing is held; trades from the weekend are included in Monday's clearing session.

General concept of weekend trading:

- Trading on weekends is a sequence of additional weekend trading sessions conducted within the closest next working day's trading day.
- Settlement and clearing sessions during weekend trading are not conducted. All settlement is performed on the closest next working day.
- Available trading modes on weekends include: Anonymous trades (continuous order matching), Negotiated trades including trades with matching by a unique code, and Request for Stream.
- All order types available on weekdays are accessible on weekends. Orders placed after Friday's evening clearing session or with a specified calendar date (GTD) and falling into a narrowed price range transition into additional weekend trading session. GTD orders with dates not exceeding the trading day are canceled by the end of the calendar day (similar to weekdays).
- All instruments traded on the Derivatives market are admitted, except for currency contracts. The list of premium options on shares admitted for trading on weekends will be synchronized with the list of shares admitted for trading on weekends on the Securities market.
- Operational features:
  - Expiration/Early expiration on weekends is not conducted. Applications for early expiration of options or exit from perpetual futures on weekends are allowed.
  - When concluding trades during additional weekend trading session, commission obligations are accounted for in the free limit from the moment of trade conclusion.
  - Commission/charges deductions occur during the clearing session on the closest next working day.
  - Margin requirements for additional weekend trading session are not established. Margin requirements are established and executed on the closest next working day.
  - Funding for perpetual futures during additional weekend trading session is not calculated and remains at 0.
  - Instrument parameters (risk parameters, price increment, lot size, etc.) are not changed on weekends. Launching new instruments on weekends is not conducted.
  - No new members are admitted; online registration of end clients is available.

2.12.1. Session model. Trading periods

Due to the decision to conduct trading on weekends, a new additional trading session is introduced with the possibility of conducting opening auctions. This session starts on weekends and relates to the trading session of the closest next working day after the weekend trading days. Each weekend has its own additional trading session. There can be more than two consecutive sessions if weekends include holidays. There can be a single session if Saturday is a working day or used for scheduled maintenance. Multiple sessions with a shift can occur, for example, if Saturday is a usual working day, Sunday and Monday are weekend trading days, and Tuesday is a usual working day.

Additional weekend trading session timeframe: 10:00 AM - 07:00 PM. Opening auction is planned before each additional weekend trading session. Auction time: 9:50 AM - 10:00 AM.

A new table, trade_periods - parameters of trading periods, has been alongside the 'session' table to store the schedule of additional trading sessions. This table is transmitted through the gateway in the FORTS_REFDATA_REPL.

Table 2. trade_periods

Field	Type	Description
replID	i8	Service field of the replication subsystem

Field	Type	Description
replRev	i8	Service field of the replication subsystem
replAct	i8	Service field of the replication subsystem
trade_period_id	i8	Trading period identifier
sess_id	i4	Trading day ID
trade_period_type	i8	Trading period type: • 1 - Main trading session • 2 - Additional evening trading session • 4 - Additional weekend trading session • 8 - Additional morning trading session
trade_period_begin	t	Date and time of the start of the trading period
trade_period_end	t	Date and time of the end of the trading period
trade_period_name	c255	Trading period name
trade_period_flags	i8	Additional properties' flags of the trading period: • 0x01 - Pseudo extension was performed before the trading period

Example of trading day schedule

Regular weekends: March 22, 2025 (Saturday), March 23, 2025 (Sunday)

Table 3. session

sess_id	begin	end	eve_begin	eve_end	mon_begin	mon_end
100	21.03.2025 9:00	21.03.2025 19:00	21.03.2025 19:00	21.03.2025 23:50	21.03.2025 7:00	21.03.2025 9:00
101	24.03.2025 9:00	24.03.2025 19:00	24.03.2025 19:00	24.03.2025 23:50	24.03.2025 7:00	24.03.2025 9:00

Table 4. trade_periods

trade_period_id	sess_id	trade_period_type	trade_period_begin	trade_period_end	trade_period_name
1	100	8	21.03.2025 7:00	21.03.2025 9:00	morning session
2	100	1	21.03.2025 9:00	21.03.2025 19:00	main session
3	100	2	21.03.2025 19:00	21.03.2025 23:50	evening session
4	101	4	22.03.2025 10:00	22.03.2025 19:00	weekend session
5	101	4	23.03.2025 10:00	23.03.2025 19:00	weekend session
6	101	8	24.03.2025 7:00	24.03.2025 9:00	morning session
7	101	1	24.03.2025 9:00	24.03.2025 19:00	main session
8	101	2	24.03.2025 19:00	24.03.2025 23:50	evening session

2.12.2. Access of brokerage firms to weekend trading

By default, all members are admitted to trades within additional weekend trading sessions. Members who do not wish to engage in trades during additional weekend sessions must submit a permanent application to limit the start time of trade execution with the Clearing center for each brokerage firm. The status update occurs in the Evening clearing session. The admission/denial status is preserved until modified. Admission allows to submit or amend orders (including "instructions to the NCC") during additional weekend trading sessions for the given brokerage firm. Cancellation of orders (and "instructions to NCC"), including bulk cancellations, are permitted. There are also no restrictions on performing non-trading operations.

Active orders of trading members who have submitted an application of withdrawal from weekend trading for the brokerage firm are automatically canceled after the completion of the trading period preceding the additional weekend trading session.

The access flag of brokerage firms to weekend trading is transmitted through the gateway in the order_allowed_in_weekend_session field of the 'dealer' table in the FORTS_REFDATA_REPL and FORTS_INFO_REPL streams.

2.12.3. Instrument admission to additional weekend trading session

No admission of new instruments or changes to instrument parameters (price increment, lot size, etc.) during additional weekend trading sessions. Current instruments are admitted by assigning them to trades within the "Additional Weekend Trading Session" period. The trading flag in the additional weekend trading session for instruments is transmitted through the gateway in the trade_period_access field of the 'fut_instruments', 'fut_sess_contents' and 'opt_sess_contents' tables in FORTS_REFDATA_REPL stream.

2.12.4. Opening auction crossing trading periods

It is now possible to schedule multiple opening auctions within one trading day, including:

- Opening auction without crossing the current trading period: Morning session opening auction, weekend session opening auction;
- Opening auction crossing the current trading period: The main session opening auction overlaps with the period of the additional morning session.

Example of opening auction schedule

Regular weekends: March 22, 2025 (Saturday), March 23, (Sunday). In practice, 4 opening auctions occur in the extended Saturday-Monday trading day, differing in behavior and time periods

- 2 weekend session opening auctions start at 09:50 AM (Saturday, Sunday) – do not overlap with trading.
- Morning additional session opening auction on Monday starts at 06:50 AM – does not overlap with other trading.
- Main session opening auction on Monday starts at 08:50 AM – overlaps with the additional morning session. Instruments not admitted to the morning additional session participate.

Intersection highlighted in bold

Table 5. session

sess_id	begin	end	eve_begin	eve_end	mon_begin	mon_end
100	21.03.2025 9:00	21.03.2025 19:00	21.03.2025 19:00	21.03.2025 23:50	21.03.2025 7:00	21.03.2025 9:00
101	24.03.2025 9:00	24.03.2025 19:00	24.03.2025 19:00	24.03.2025 23:50	24.03.2025 7:00	24.03.2025 9:00

Table 6. trade_periods

trade_period_id	sess_id	trade_period_type	trade_period_begin	trade_period_end	trade_period_name
1	100	8	21.03.2025 7:00	21.03.2025 9:00	morning session
2	100	1	21.03.2025 9:00	21.03.2025 19:00	main session
3	100	2	21.03.2025 19:00	21.03.2025 23:50	evening session
4	101	4	22.03.2025 10:00	22.03.2025 19:00	weekend session
5	101	4	23.03.2025 10:00	23.03.2025 19:00	weekend session
6	**101**	**8**	**24.03.2025 7:00**	**24.03.2025 9:00**	**morning session**
7	101	1	24.03.2025 9:00	24.03.2025 19:00	main session
8	101	2	24.03.2025 19:00	24.03.2025 23:50	evening session

Table 7. discrete_auction

discrete_auction_id	sess_id	add_order_start	add_order_finish_from	add_order_finish_till	trade_period_type
morning opening auction on Friday	100	21.03.2025 06:50	21.03.2025 06:55	21.03.2025 06:58	8
weekend opening auction on Saturday	101	22.03.2025 09:50	22.03.2025 09:55	22.03.2025 09:58	4
weekend opening auction on Sunday	101	23.03.2025 09:50	23.03.2025 09:55	23.03.2025 09:58	4
morning opening auction on Monday	101	24.03.2025 06:50	24.03.2025 06:55	24.03.2025 06:58	8
main opening auction on Monday	**101**	**24.03.2025 08:50**	**24.03.2025 08:55**	**24.03.2025 08:58**	**1**

The instruments will participate in all opening auctions before the session period during which trading of these instruments' underlying futures contracts is allowed, provided that the order collection period of this opening auctions does not overlap with the previous trading period during which trading of these instruments' underlying futures contracts is allowed.

2.12.5. Price ranges and risk rates in additional weekend trading session

In weekend trading, a symmetrical narrowing of price ranges is conducted, relative to the boundaries calculated during Friday's clearing session. The narrowing applies only to the weekend period and does not affect options contracts. The cancellation of the narrowing of price ranges occurs a few hours before the start of Monday's morning session.

During the narrowing of boundaries, all active buy orders for futures contracts/calendar spreads above the upper price corridor/size of the spread and all active sell orders for futures contracts/calendar spreads below the lower price corridor/size of the spread are removed from the trading system.

Dynamic adjustments of price ranges during trading on weekends are not provided.

No changes to risk rates are planned for weekends. In additional weekend trading session, rates disclosed for Monday are used.

Volatility curves are fixed on weekends. Curve recalculation stops at Friday's trading close (11:50 PM).

2.12.6. Single calendar day data in public SPECTRA interfaces

Version 8.3 introduces the calculation and publication of position and order snapshots at the start of the current day through the gateway. Thus, the client application can load a completed snapshot of positions and active orders at the start of the calendar day and subsequently receive streams of orders and trades submitted and concluded during the current calendar day.

A new synchronous event - start_of_calendar_date was introduced. This event is generated and published daily in all streams after trading concludes (after GTD order cancellation). Upon this event, PosBuilder and OrderBook publish position and order snapshots at the moment of the synchronous event. Meta-information about the exec_log revision, on which the snapshot was generated, is also published. Clients can use the snapshots and the initial revision to open exec_log-derived streams to receive data only for the current day.

2.12.6.1. Position snapshots

Fields xday_open_qty, xday_open_buys_qty, xday_open_sells_qty were added to the 'position' and 'position_sa' tables in the FORTS_POS_REPL stream. These fields publish the position snapshot at the start of the calendar day upon receiving the start_of_calendar_date event. A new auxiliary table, info, has been added to the stream, containing one record with the following fields:

- trades_rev – The revision of the start_of_calendar_date synchronous event;
- trades_lifenum – Life number of the stream corresponding to the start_of_calendar_date synchronous event;
- server_time – date and time of position snapshot creation (for verification).

2.12.6.2. Order snapshots

To publish active order snapshots at the start of the day, two new tables, orders_currentday and info_currentday, have been added to the FORTS_ORDBOOK_REPL and FORTS_USERORDERBOOK_REPL streams. The orders_currentday table publishes the active order snapshot at the start of the calendar day upon receiving the start_of_calendar_date event. The orders_currentday table is a copy of the orders table with modifications to present active order information in the format "as if this order was just submitted in its current form":

- The public_action and private_action fields always display 1 (order submission).

The info_currentday table contains one record with the fields:

- logRev – The revision of the start_of_calendar_date synchronous event;
- lifenum – Life number of the stream;
- server_time – Date and time of order snapshot creation (for verification).

Additionally, the multileg_orders and multileg_orders_currentday tables with multileg orders have been added to the FORTS_ORDBOOK_REPL and FORTS_USERORDERBOOK_REPL streams.

2.12.6.3. Scenario for working with new data

To open the current day's trade stream, open the FORTS_POS_REPL stream and read the auxiliary fields from the 'info' table. When calling the listener for FORTS_TRADE_REPL and FORTS_DEALS_REPL streams, use the following syntax:

```
CG_RESULT cg_lsn_open(cg_listener_t* lsn, const char* settings)
```

Pass the following string to the settings parameter:

```
mode=snapshot+online,lifenum=<trades_lifenum>,rev.deal=<trades_rev>,rev.heart_beat=<trades_rev>
```

where <trades_lifenum> and <trades_rev> are read from the 'info' table of the 'position' stream.

This ensures receiving trades and other tables only for the current day. Current position calculation upon receiving a trade: add the trade volume to xday_open_qty considering the direction.

Orders_log table requires special consideration due to potential multi-day orders. The following approach is recommended. After receiving the order snapshot, work with the orders_log table in the FORTS_TRADE_REPL stream as described above for trades, or use the following URL syntax when creating a listener for the orders_log table:

```
p2ordbook://FORTS_TRADE_REPL;snapshot=FORTS_USERORDERBOOK_REPL;
online.scheme=|FILE|.\forts_scheme.ini|ORDLOG;online.data=orders_log;
snapshot.scheme=|FILE|.\forts_scheme.ini|OrderBook;snapshot.data=orders_currentday;
snapshot.bind=info_currentday.logRev
```

Opening the stream of your orders through the pair of streams: "order snapshot" + "order log".

In this case, the CGate library will open the order snapshot stream, receive the snapshot, automatically transition to the online data with the revision from the info_currentday table, and receive data online.

Tables 'multileg_orders_currentday' and 'multileg_orders_log' should be processed similarly.

Note: For analyzing the active order snapshot at the start of the day, the values of the 'private/public _ amount' and 'private/public _action' fields have no meaning and should be ignored. To obtain the number of contracts in the order at the start of the current day, you should look at the 'private/public _amount_rest' fields. To obtain the initial number in the application, you should use the 'private/public _init_amount' fields.

2.12.7. Price range check for option orders during additional weekend trading session

During additional weekend trading session, the Spectra Trading System performs checks to ensure that option prices in client orders comply with the price limits set in the system. The logic is standard: buy orders with prices above the upper price limit and sell orders with prices below the lower price limit are considered invalid and rejected by the system.

The calculation of price limits during additional weekend trading session is performed for each option based on factors affecting prices (underlying asset price, volatility scenarios, interest rates) in accordance with the existing methodology https://www.moex.com/a1898, with the following adjustments:

- The maximum and minimum volatility scenarios for the underlying asset are set to the calculated underlying asset volatility at 11:50 PM on the last trading day before additional weekend trading session;
- The upper and lower bounds of the futures contract (collateral) for the option are used as the maximum and minimum underlying asset prices.

The calculated values are published in the gateway in the 'volat' table of the FORTS_VOLAT_REPL stream in the fields 'up_prem_weekend' and 'down_prem_weekend' (outside additional weekend trading session, these fields are set to -1). These values are displayed in the Derivatives Market Terminal in the 'Financial instruments' window under the fields "Upper price limit for AWTS" (weekend_price_lmt_high) and "Lower price limit for AWTS" (weekend_price_lmt_low).

Checking option orders against weekend session price limits is mandatory for the anonymous trading boards. Additionally, clients retain the ability to trigger a check against the current option algorithm when submitting an order (is_check_limit), where the price is verified against the price ranges volat.up_prem and volat.down_prem.

2.13. Margin Calls

Margin Calls are issued in accordance with the Clearing Rules of NCC at 10:00 AM MSK. The planned date and time of automatic Margin Call fixation is transmitted in the 'session' table of the FORTS_REFDATA_REPL stream in the margin_call_fix_schedule field.

2.13.1. Margin Call fixation

At the moment of Margin Call fixation (by schedule or as a result of recalculation), in the FORTS_PART_REPL stream, in the part_sa table, the following fields will be populated:

- margin_call_moment - date and time of Margin Call fixation;
- margin_call_fix - fixed Margin Call amount;
- margin_call_current - current status of fixed Margin Call.

At the moment of Margin Call fixation, margin_call_fix and margin_call_current values are identical.

Additionally, at the moment of Margin Call fixation, if debt exists, clearing members will receive a new report MTYYYYMMDD.zip (YYYYMMDD – date) "Margin Call Report".

2.13.2. Recalculation of Margin Calls

The NCC may initiate a procedure for recalculation of Margin Calls. In this case:

- The values of margin_call_moment, margin_call_fix, and margin_call_current are updated to the new values;
- The new values become effective for the remainder of the trading day.

2.13.3. Values status during the trading day

- margin_call_fix – the value remains unchanged throughout the entire trading day from the last fixation;
- margin_call_current – may change only upon repayment of debt by the amount of repayment.

Changes related to an increase in debt (e.g., adjustment of initial margin due to expansion of price limits), do not affect the current status of fixed Margin Calls.

The values of fixed Margin Calls and their current status are informational only; they do not reduce the trading limit and are not considered during fund withdrawals.

After the clearing session and before the next actual Margin Call fixation, the fields margin_call_moment, margin_call_fix, and margin_call_current in the part_sa table of the FORTS_PART_REPL stream remain empty.

3. Trading gate description

3.1. PLAZA II gateway. Components, installation and setup.

3.1.1. Components and architecture

The PLAZA II gateway consists of the following software components:

- The 'P2MQRouter' module. This module provides the following services:
  - Establishing TCP-connections to the Exchange servers.

    Normally, the PLAZA II gateway uses four TCP-connections to the Exchange servers:

    - Connection for sending requests and commands
    - Connection for receiving the main market data such as aggregated order-books streams and the streams 'FORTS_ORDLOG_REPL', 'FORTS_DEALS_REPL', 'FORTS_TRADE_REPL' and 'FORTS_COMMON_REPL'.
    - Connection for receiving auxiliary and reference streams
    - Connection for receiving snapshots (at the first connection or when recovering after loss of connection)

    'P2MQRouter' software handles all TCP-connections, with settings specified in the INI files where connection 'Other Data' is specified as the default outcoming connection and the other connections are specified as outcoming direct connections. This structure is used as the standard to establish connection directly to the Exchange server farm, but connection via a Brokerage Firm server server may require a different structure; in this case, clients should apply to the server's owners for more details about connection.
  - Receiving/sending P2-messages.
  - Encrypting data sent by participants and decrypting data received from the Exchange.
  - Authentication of participants in the Exchange network.
- 'cgate' - the gateway library.

  The library is the official software interface, provided to trading participants along with their clients as well as to software developers. The interface provides availability to create and send messages into the trading system and receive trading data from the trading system (data replication). There are x32 and x64 versions available for Windows systems, as well as a version for Linux OS.
- PLAZA II system libraries.
- Software development kit, including additional utilities, command files, documentation and test examples.

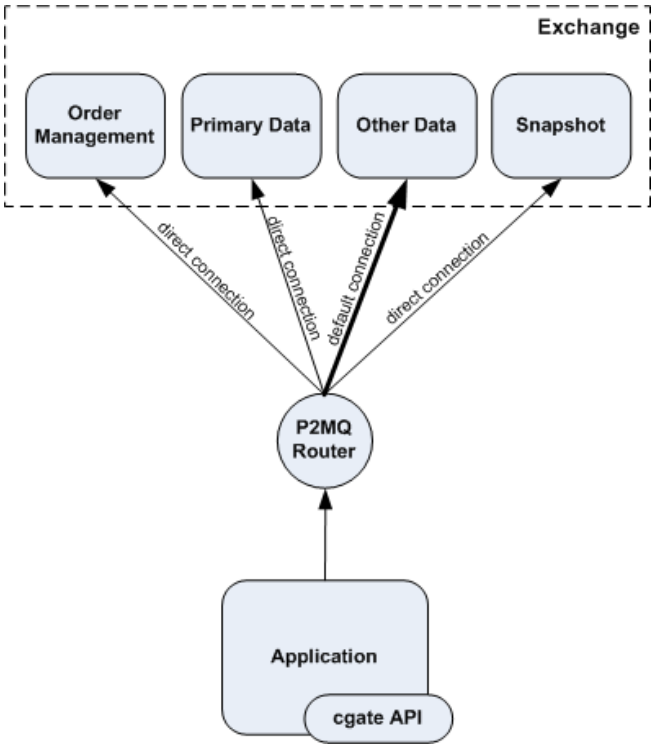

Figure 7. Gateway architecture

3.1.2. Hardware and software requirements

3.1.2.1. Hardware requirements

Hardware requirements may vary depending on usage of the PLAZA II gate.

The minimum system requirements for individual login without disk saving option are as follows:

- CPU: Core 2 duo 1 Ghz or better
- Memory: 2 GB or more for x32 systems, 4 GB or more for x64 systems.

The minimum system requirements for brokerage firm login without disk saving option are as follows:

- CPU: Intel Xeon 53xx 2 cores or better (or a similar AMD CPU 2 cores or better)
- Memory: 24 GB or more
- Separate SAS controller. Minimum 2 hard drives in RAID1. Two partitions, 30 GB each.

The minimum system requirements for brokerage firm login with disk saving option are as follows:

- CPU: Intel Xeon 53xx 2 cores or better (or a similar AMD CPU 2 cores or better)
- Memory: 4 GB or more
- Separate SAS controller powered with the write-back cache policy. Minimum 4 hard disks in RAID10. Two partitions, 30 GB each.

3.1.2.2. Software requirements

The following operation systems are supported by the gateway software:

- Microsoft Windows 10 (both 32 bit and 64 bit OS versions are supported)
- Microsoft Windows Server 2016/2019 (both 32 bit and 64 bit OS versions are supported)
- Linux RedHat/CentOS 7, AstraLinux SE v. 1.7, RedOS v. 7.3, AltLinux workstation 10.1. It is also possible to use other distributions

3.1.3. Installation for Windows

Download the latest gateway software version from https://ftp.moex.com/pub/ClientsAPI/Spectra/CGate/. The installation file's name is 'setup_SpectraCGate_x64_vx.x.x.msi', where x.x.x is the software version number, for example 7.0.0.

Run 'setup_SpectraCGate_x64_vx.x.x.msi'. The installation wizard will guide you through the installation process:

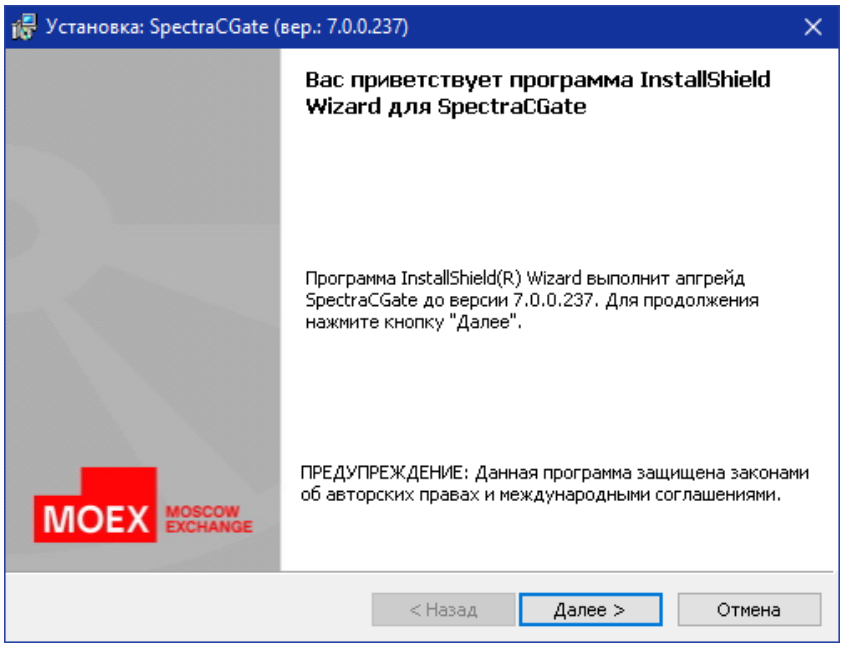

Figure 8. Installation start

Click the 'Далее' button to continue with installation:

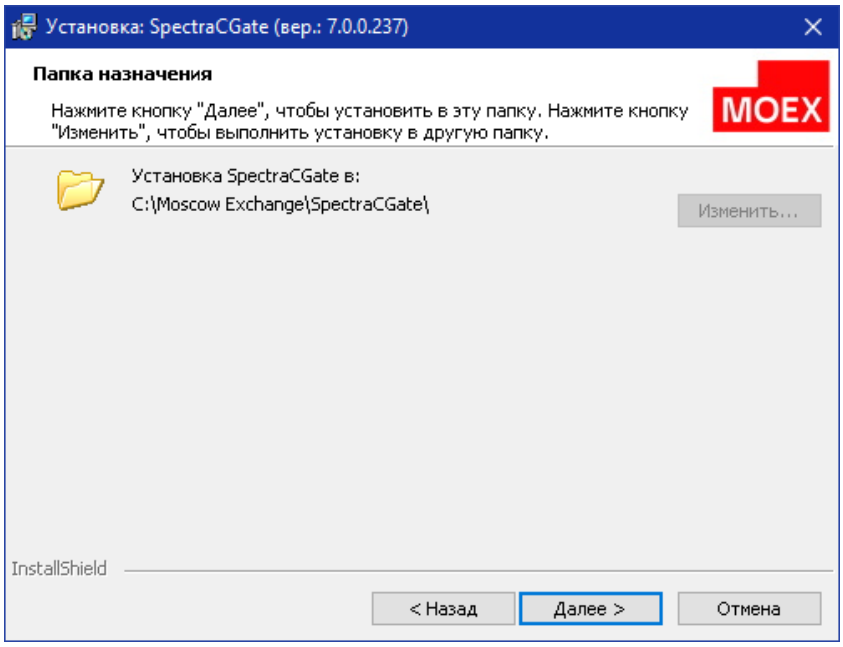

Figure 9. Destination folder

The default destination folder is C:\Moscow Exchange\SpectraCGate\. To confirm installation using the default folder and continue to the next step, click the 'Далее' button.

To change the destination folder, click the 'Изменить...' button. A new window appears in the screen; in this window, select a new destination folder using the "Поиск в папке" button; to move one level up in the folder tree use the button. Also, you can create a new destination folder using the button, or select an already existing one by manually typing the path in the "Имя папки" entry box in the lower part of the window. Click 'OK' to apply the changes you have made and close the current window. In the "Папка назначения" window, press the "Далее" button to continue to the next step.

Please be known that you will only be able to change the destination folder upon the initial installation, or when you are re-installing the software anew. In all other cases, you will not be able to change the destination folder (the 'Изменить...' button will be disabled).

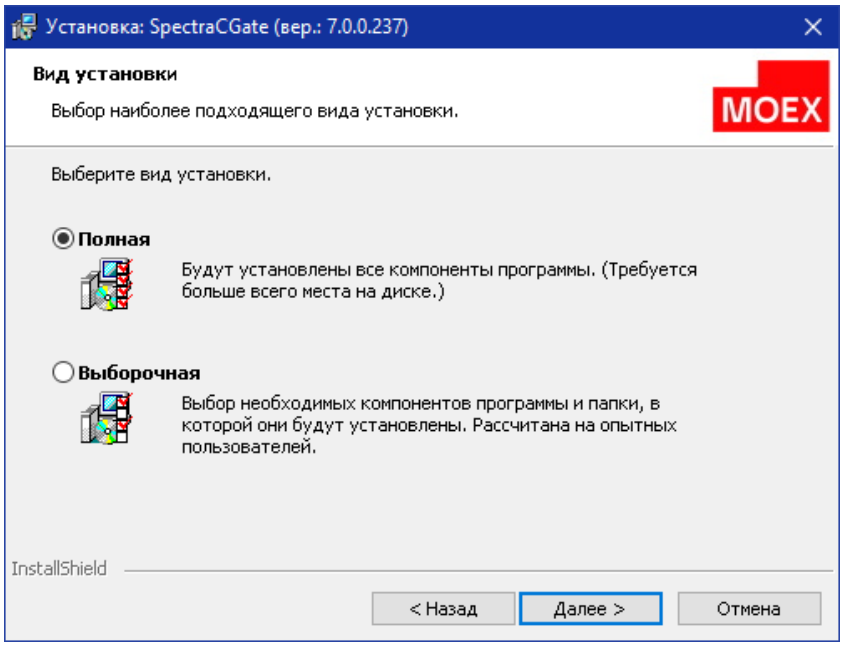

Figure 10. Select components to install

Select the installation mode you want to use, full or custom. The full install mode will install all the gateway components including module P2MQRouter, library cgate, additional utilities and the software development kit. The custom install mode allows you to manually select software components to install.

Click the 'Далее' button to continue with installation:

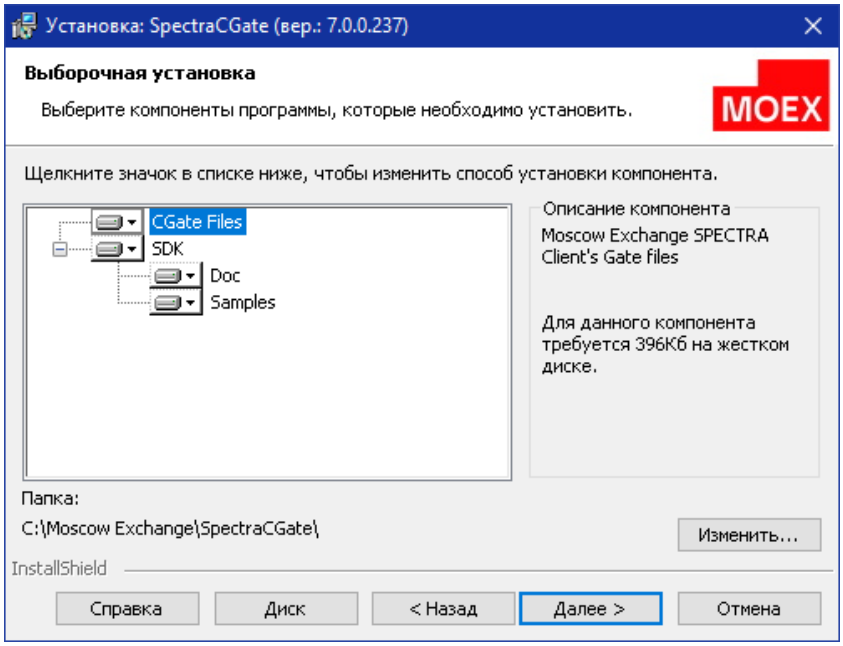

Figure 11. Custom install

Select the software components you need to install and the destination folder. The destination folder should be selected in accordance with the administrative recommendations.

Click the 'Далее' button to continue with installation:

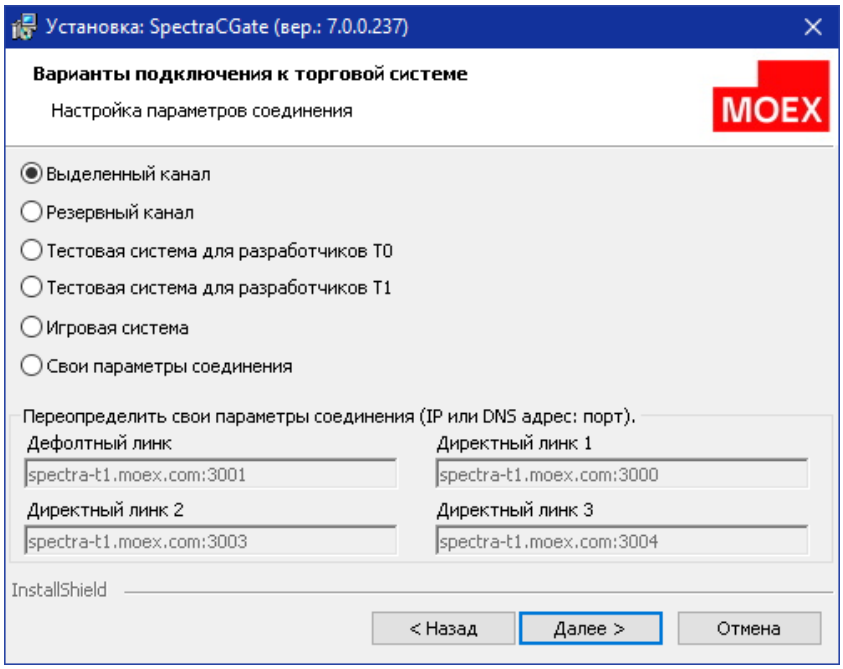

Figure 12. Select an address to connect

Select the trading system to which you want to connect (production, test, game, etc), or enter your parameters for connection to the exchange servers. The connection parameters are stored in a separate configuration file for each connection option. A configuration files are in the '\links' directory of the installation directory.

Connection option	Configuration file	Details
Выделенный канал	links_public.prod.ini	Connect to Spectra - production system
Резервный канал	links_public.rezerv.ini	Connect to Spectra - reserve system
Тестовая система для разработчиков T0	links_public.t0.ini	Connect to Spectra - public testing system - current version
Тестовая система для разработчиков T1	links_public.t1.ini	Connect to Spectra - public testing system - future version
Игровая система	links_public.game.ini	Connect to Spectra - gaming system
Свои параметры соединения	links_public.custom.ini	User defined connection

After the installation complete, a link to the corresponding file with connection parameters will be added to the ini-file of the 'P2MQRouter' module in the 'connections_ini' parameter. To change the connection type, just restart the installer and select the desired option. Please note that if you reinstall or uninstall the system, the '\links' directory and the file with user connection settings (links_public.custom.ini) are not deleted.

The fields in the user settings section display:

- if initial installation - default values (for example, parameters from links_public.t1.ini).
- if reinstalling - user connection options from links_public.custom.ini or client_router.ini. If there are no files, the default values are displayed.

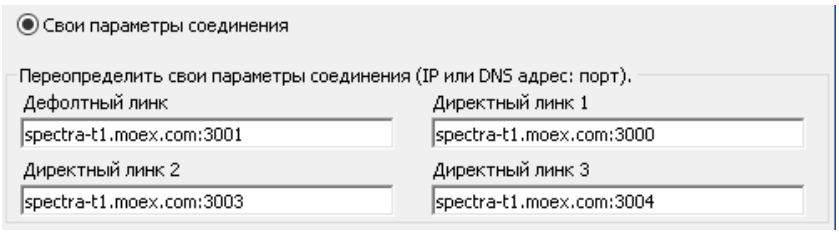

Figure 13. User connection settings section

For selecting the proper connection type, you should contact your brokerage firm and/or the Exchange technical support service.

Click the 'Далее' button to continue with installation:

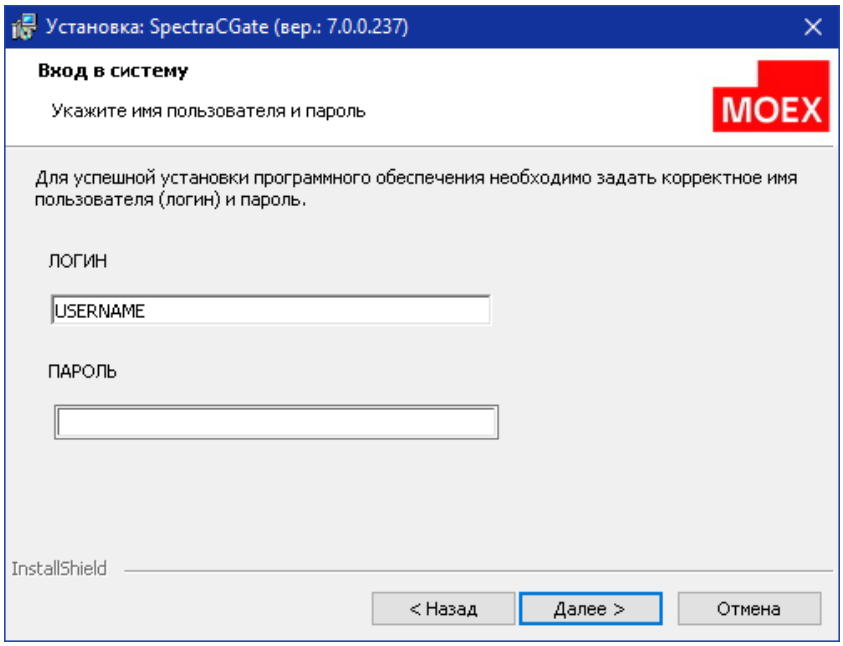

Figure 14. Enter username and password

Enter username and password for the connection selected in the previous step. Please note that usernames and passwords differ for each connection type (production, testing and gaming).

After the installation complete, the username and password will be added to a separate configuration file 'auth_client.ini', which will be created in the '\auth' directory of the installation directory, and a link to this file will be added to the 'auth_ini' parameter of the 'P2MQRouter' module ini-file.

When reinstalling, the username and password values specified in 'auth_client.ini' or 'client_router.ini' files are displayed in the form fields. Please note that if you reinstall or uninstall the system, the '\auth' directory and the file with identification data (auth_client.ini) are not deleted.

Click the 'Далее' button to continue with the installation:

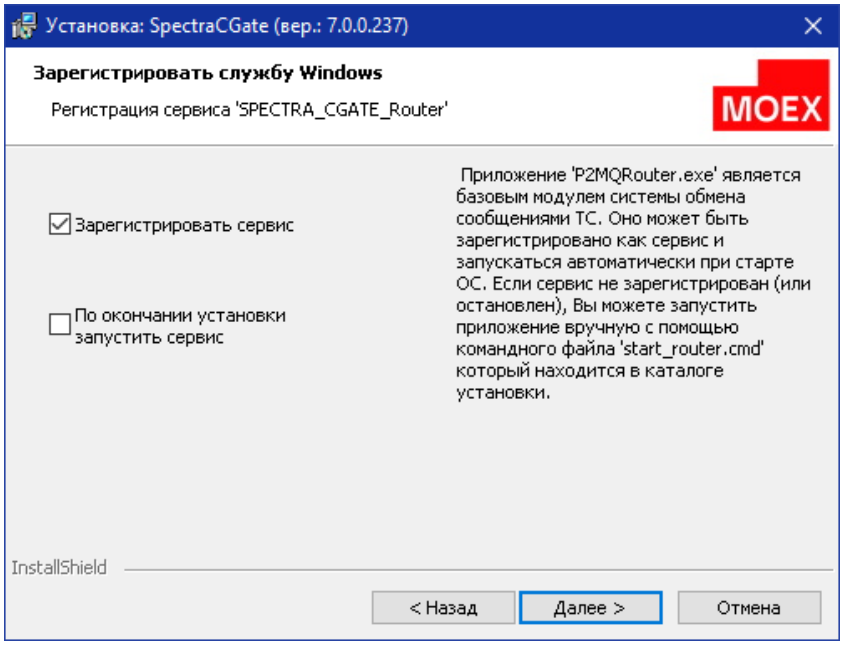

Figure 15. Registering router as OS service

If you need to install the Router as Windows service, check the appropriate checkbox and click the 'Далее' button to continue with the installation.

If you do not register the P2MQRouter as an OS service, you can do it later manually using the command file 'install_router.cmd (uninstall_router.cmd)'. The file is a part of distribution kit.

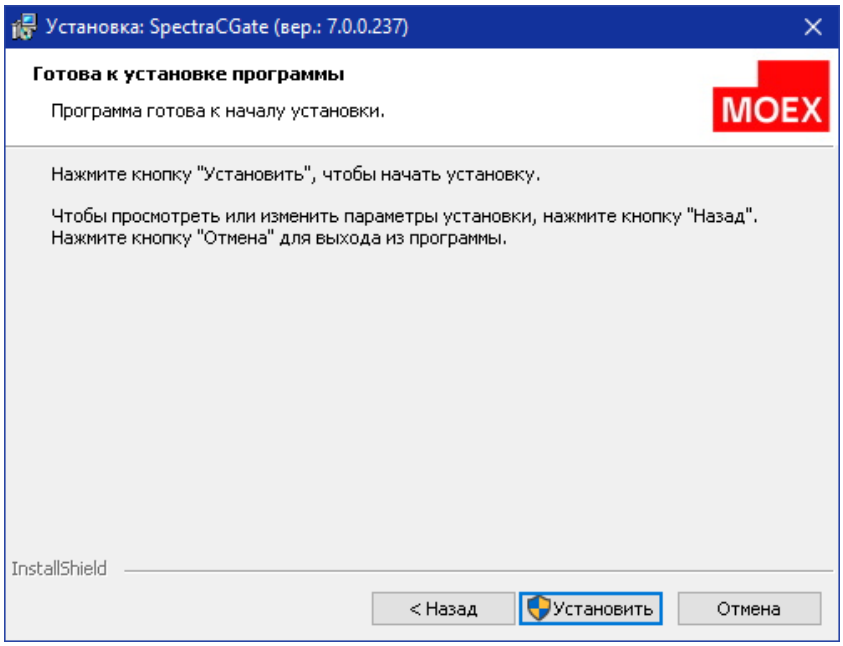

Figure 16. Starting installation

Click 'Установить' to begin installation.

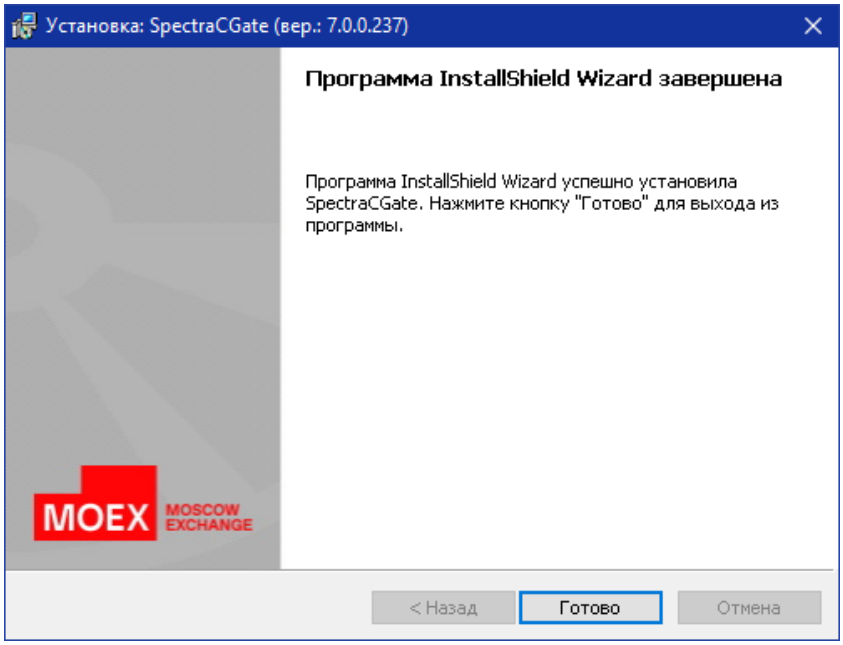

Figure 17. Finishing installation

Click 'Готово' to finish the installation.

3.1.4. Installation for Linux

3.1.4.1. Installation from zip archive

The distribution kit consists of installation script ('install.sh') and archive file ('tar.gz'); the archive file contains loadable modules 'cgate' and 'cgate_java', files 'include', documentation files and test examples. The distribution kit can be downloaded at https://ftp.moex.com/pub/ClientsAPI/Spectra/CGate/.

Installation order:

1. Execute the command:

```
chmod 755 ./install.sh
```

2. Execute the command:

```
./install.sh ./cgate_linux_amd64-7.12.0.103.zip
```

Please note that the archive file name depends on the software version, and may differ from the one shown above!

3. When you receive 'Please, enter cgate install path', specify the full path to the folder to decompress the cgate software.
4. When you receive 'Please, enter P2 login', specify the user's login.
5. When you receive 'Please, enter P2 password', specify the user's password.

3.1.4.2. Installation from a deb package or rpm package

Installation steps:

1. Download and install the gateway package by running:

```
dpkg -i cgate_<version>_amd64.deb
```

in case of installation from a deb package, or by running:

```
rpm -U cgate-<version>.x86_64.rpm
```

in case of installation from an rpm package.

'<version>' - distribution version number

Installation location:

Installation location	Description
/opt/moex/cgate	Binaries, schematics, gateway documentation
/etc/opt/moex/cgate	Configuration files, auth.ini - files
/var/opt/moex/cgate	Logging directory, trace files
/usr/share/doc	Copyright, installation documentation

2. Open the '/etc/opt/moex/cgate/auth' directory and specify the login/password for connection in the corresponding ini file:
   - prod.ini - to connect to the production system
   - t1.ini - to connect to the public test system T1
   - t0.ini - to connect to the public test system T0
   - game.ini - to connect to the gaming system

   Please note that if the package is upgraded, the '\auth' directory and files with connection settings are not deleted, so you do not need to reconfigure the login/password!

3. Start the service (router) with the command:

```
systemctl start cgate@<profile>
```

'<profile>' - connection option. Available options:

- prod - connection to production system
- rezerv - connection to reserve system
- t1 - connection to public test system T1
- t0 - connection to public test system T0
- game - connection to gaming system
- rfs.prod - connecting to a production system with additional access to RFS
- rfs.rezerv - connecting to a reserve system with additional access to RFS
- rfs.t1 - connection to public test system T1 with additional access to RFS
- rfs.t0 - connection to public test system T0 with additional access to RFS

Please note that when you upgrade a package, the running service stops, so after the upgrade, the service must be restarted!

3.1.5. Developer guidelines

3.1.5.1. Usage of test examples

In order to verify the installation accuracy you can compile and run the test examples included into the distribution kit.

The examples can be found either in folder Moscow Exchange\SpectraCGate\SDK\samples for Windows OS, or /usr/share/doc/cgate-examples (/opt/moex/cgate/samples) for Linux OS. To compile examples, you should run the special scripts, which may vary depending on the operation system and programming language used. For Linux OS, it is recommended to copy the examples into your login directory to compile them.

Description of examples:

1. Example 'aggrspy'

   'aggrspy' is an example which is used to build aggregated orderbook for to buy and sell a fixed instrument for the stream 'FORTS_AGGR50_REPL'. Press 'Enter' to display the orderbook snapshot.

   Execute:

   ```
   aggrspy ISIN_ID depth outfile [r]
   ```

   Input arguments:

   - 'isin_id' - instrument's ID;
   - 'depth' - depth of orderbook (up to 50);
   - 'outfile' - orderbook file for printing;
   - 'r' - reverse the sorting order (for instrument with reversed sorting order).

2. Example 'repl'

   'repl' is an example which is used for receiving replication data and printing all incoming messages into a log-file. When disconnected, the replica transfer process starts anew from the beginning. No input parameters required.

3. Example 'repl_resume'

   'repl_resume' is an example very similar to 'repl'. When disconnected, it allows to resume replica transfer process starting from the last message 'TN_COMMIT'. No input parameters required.

4. Example 'send'

   'send' is used to add order into the SPECTRA and write incoming replies into the trading system log. No input parameters required.

5. Example 'orderbook'

   'orderbook' is an example which is used to build aggregated orderbook for to buy and sell a fixed instrument for the online stream 'FORTS_ORDLOG_REPL' along with the snapshot stream 'FORTS_USERORDERBOOK_REPL'. It is recommended to use it for developing 'late join', and also for minimizing inactivity time when archival data is being downloaded. Press 'Enter' to display the orderbook snapshot.

   Execute:

   ```
   orderbook ISIN_ID depth outfile [r]
   ```

   Input arguments:

   - 'isin_id' - instrument's ID;
   - 'depth' - depth of orderbook (up to 50);
   - 'outfile' - orderbook file for printing;
   - 'r' - reverse the sorting order (for instrument with reversed sorting order).

6. Example 'p2sys'

   'p2sys' is an example, which is used for authorising the Router from cgate side. The following actions are executed cyclically:

   - Send erroneous login/pwd pair, get the 'logon failed' in reply;
   - Send the correct login/pwd pair;
   - Receive an 'authorisation successful' message, send 'logout' request;
   - Go back to the beginning.

7. Example 'send_mt'

   'send_mt' is an example of multi-thread order adding. (Please note, that only C++11compilers are supported!). Thread 1 is used for adding orders, while thread 2 is used for processing 'reply' messages received.

8. Example of instruments_state

   The instruments_state example demonstrates the independent calculation of instrument statuses based on the FORTS_SECURITYGROUPSTATE_REPL stream (for more details, see "Section 2.5.7, “Instrument status broadcast service"").

Before executing the examples, please make sure that 'P2MQRouter' has started and connected to the PLAZA II network (with touter massages analyzed), the INI files are accessible for the example file, as well as the PLAZA II libraries (it may be necessary to add 'Moscow Exchange\SpectraCGate\bin' folder into the PATH environment variable or specify Moscow Exchange\SpectraCGate\bin for your development environment).

Note: The examples above are not intended to be used with data other than test data! It is strictly prohibited to use these examples for working with the real logins!

3.1.5.2. Distributed configurations

The 'cgate' application and the 'P2MQRouter' module can be distributed to different computers. To distribute the modules in the brokerage firm network, you should do the following: a) install the 'Router 'module to the computer connected to the Exchange network; b) install 'cgate' to the client computer with the client application installed; c) specify the following settings:

- On the client side:
  - Specify the 'Host 'and 'Port' settings in accordance with that of your corporate network router.
  - Specify the Password settings (the local AppName application password for the router, which must be applied every time the application connects router from outside of the same computer). Please note that the local connection password is not the same as the PLAZA II authentication password!
- On the router side:
  - Add the '<AppName>=<local password>' string into the 'router.ini' file, [AS:Local] section, where 'AppName' (the application name) and 'Password' (the local password) should match the parameters transmitted by the client application.

3.1.5.3. Recommendations for third-party companies on including the Moscow Exchange runtimes into user application when distributing the user software

Users should copy the file set from the installation folder (Moscow Exchange\SpectraCGate\bin), as well as data and messages schemes (Moscow Exchange\SpectraCGate\SDK\scheme) into the folder containing user application. All these software parts should be distributed together.

It is not allowed to use different versions of 'P2MQRouter' and 'cgate' due to incompatibility. Before installing user application, please make sure that the 'P2MQRouter' version matches the one used in developing.

3.2. Market data structure

This section describes the structure of information sent by PLAZA II gateway.

All transmitted data is divided into the following logical groups:

- Reference information
- Trade information
- Recovery information
- Funds and limits information
- Clearing information
- Rates and indices information
- Auxiliary data streams

3.2.1. Reference information

The reference information includes the following data:

- Trading day status and schedule

  Trading day time information and all its components: evening and morning session, additional weekend trading session time are available in 'session' and 'trade_periods' tables of the 'FORTS_REFDATA_REPL' stream. The 'session' table also indicates the trading day status, that helps to track current trading day status.

- Instruments and underlying assets dictionary, properties

Futures Instruments assigned to the trading session are available in the 'fut_sess_contents' table of the 'FORTS_REFDATA_REPL' stream. Compound istruments are also listed in the table. Options instruments are sent in the 'opt_sess_content' table of the 'FORTS_REFDATA_REPL' stream. Dictionary of the futures' underlying assets is represented by the 'fut_vcb' table of the 'FORTS_REFDATA_REPL' stream.

These directories can be updated during the trading session, for example, as a result of the suspension of trading on any instrument or during the price limit enlargement procedure

- Companies and clients references

  Are sent in the 'dealer' and 'investor' tables from the 'FORTS_REFDATA_REPL' stream. Personal clients' information is available in these references.

- Bond references

  Bonds are desribed by a set of tables from the 'FORTS_REFDATA_REPL' stream: spot asset parameters references 'fut_bond_registry', bond's instruments references 'fut_bond_isin', ACI (Accrued Coupon Income) for coupon payment dates 'fut_bond_nkd', nominal payout value for a bond 'fut_bond_nominal'.

- Parametric volatility curve parameters

  Are sent in the 'volat_coeff' tables of the 'FORTS_RISKINFOBLACK_REPL' and 'FORTS_RISKINFOBACH_REPL' streams.

To carry out operations on all of the SPECTRA trading system markets user's system should receive at least the following reference information on-line:

- Sessions' schedule (session)
- Instruments dictionary (fut_sess_contents, opt_sess_contents)

3.2.2. Trade information

Trade information includes:

- Aggregate orderbooks

  Are generated on the basis of user system requests by adding up the volume for each instrument, the price level and the direction of an order. Updated online and comes to be the main way to get information by current prices and volumes. User can select the desired depth of an orderbook from 5, 20 or 50 of price levels in each direction; this choice is made when configuring a login and can not be changed during the trading session.

  Orderbooks are sent by multiple PLAZA II replication streams: 'FORTS_AGGR5_REPL', 'FORTS_AGGR20_REPL' and 'FORTS_AGGR50_REPL'.

- Market activity

  The best bid/ask price,opening price, closing price, current settlement prices, etc are sent as a part of market activity information. This information is sent in the 'FORTS_COMMON_REPL' stream.

- User's orders log (and full orders log in the trade system)

  The entire history of user's operarions with orders is sent in user's orders log. User's orders logs are available in 'orders_log' table of the 'FORTS_TRADE_REPL' stream for futures and options; the 'multileg_orders_log' table of the 'FORTS_TRADE_REPL' stream for and multileg instruments.

  In case the user configures his login with option to receive "full orders log", he will receive the complete log of all operations with orders on market (including own operations with orders) in anonymous mode. The log will be available in the table 'orders_log' of the stream FORTS_ORDLOG_REPL.

- User's deals log

  It contains a list of user's committed deals in the current session. User's deals log are available in the 'user_deal' table of the 'FORTS_TRADE_REPL' stream for futures and options; the table 'user_multileg_deal' of the 'FORTS_TRADE_REPL' stream contains logs for multileg instruments deals.

- All trade system deals log

  It contains a list of all committed deals from all users in the current session. Information of somebody else's deals is presented in anonymous mode. User's deals logs are available in the 'deal' table of the 'FORTS_DEALS_REPL' stream for futures and options; the table 'multileg_deal' contains logs for multileg instruments deals.

3.2.3. Recovery information

To ensure fast recovery of trade information receiving after losing connection with SPECTRA, and same with late start scenario connecting to exchange, PLAZA II gateway receives periodic snapshots from recent orderbooks in a non-aggregated form. This helps to receive the recent status of personal orders (in case when the 'full orders log' option is set - all orders in the trade system) at the current time.

Snapshots of active orders are sent with 2 minute interval in 'FORTS_USERORDERBOOK_REPL' stream.

3.2.4. Funds and limits information

Includes the following:

Position information

- Positions information

  Is sent in form of time snaps in the 'FORTS_POS_REPL' stream and last deal ID, included in position calculation by each position value, is available.

- User's funds and limits information

  Is sent in form of time snaps in the 'FORTS_PART_REPL' stream. Money amount (both money and pledge), money amount at the beginning of the trade session, also current and reserved funds - all of them are available for each value of the client's account.

3.2.5. Clearing information

Clearing information, sent by PLAZA II gateway, includes the following data:

- Clearing settlement prices

  Are formed by the time of clearing session and available in the 'fut_sess_settl' table of the 'FORTS_CLR_REPL' stream. The table with settlement prices also includes the instruments whose validity period has ended allowing this table to be used to receive right prices when delivery comes.

- Registries, containing orders rejected during the clearing session.

  Contain the orders, which were not replaced during the clearing session due to lack of funds. The registry is transmitted in the 'rejected_orders' table of the 'FORTS_REJECTEDORDERS_REPL' stream.

- Client funds based on clearing results

  Include information on the amount of funds in the accounts, account activity, fees, total initial and variation margin by the time of clearing. Are sent in the 'FORTS_CLR_REPL' stream.

- Option execution orders

3.2.6. Indices and rates information

The following information is sent as a part of this group:

- Current values of RTS indices

  Includes current values of RTS index, as well as all Exchange indices values. The values in this table are updated with 15 seconds intervals. The composition of the index information includes of USD rate value, which is used in index calculation. The data is sent in the 'RTS_INDEX_REPL' stream.

- Currencies rates values

  Contain rates of currencies used in the trading system for processing contracts, calculated in a currency other than rubles. The currencies values are available in the 'curr_online' table of the 'MOEX_RATES_REPL' stream.

3.2.7. Auxiliary information streams

That group includes the streams providing the following additional functions:

- Current values of the variation margin and premium indicative

  Are sent in the 'FORTS_VM_REPL' stream in the context of the positions of clients and SA.

- Current volatility values and theoretical prices for options

  Are sent in the 'FORTS_VOLAT_REPL' stream.

3.3. Gateway usage specifics

3.3.1. Service replication fields

Each replicated table contains three fields of the fixed type i8 in the top, which are used for replication purpose:

- **replID** — the unique record ID. When a new record appears in the table, the record is assigned with a unique ID. Even though a table may already have a primary ID-key, the one and only ID for replication purpose is the ID contained in the field 'replID'.

- **replRev** — when there is a change made in the table such as record insert, record edit or record deletion, this record will be assigned with a new value in the field 'replRev' (maximum previous 'replRev' value + 1).
- **replAct** — flag of a deleted record. If 'replAct' contains a value other than 0, then the record has been deleted. If 'replAct' contains 0, then the record is active.

3.3.2. Commands

To send a command, you should create a publisher with parameters 'NAME = FORTS_SRV', 'category = FORTS_MSG'. 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 type 'p2mqreply' listener.

In case of the message delivery and handling errors, the client receives either sending message function error or the 'system error' (msgid=100) message in return.

Field	Type	Description
code	i4	Return code
message	c255	Message body

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

3.3.3. Flood control

The control system of clients' application flood control is functioning in the SPECTRA trade 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 on SPECTRA with 30, 60, 90, etc. (but not more than 3000) trading transactions per second. Trade operations are all transactions associated with order managing. Amount of non-trade (all the rest) operations for any type of login is limited in 1000 transactions per second.

During nighttime technical maintenance of the SPECTRA trade system, the flood control threshold for non-trade operations is increased to 5,000 transactions per second.

If you exceed the limit of messages, the control system does not transmit a message into the trade system core, and sends the user a reply message with the notification of denial of service. It is msgid=99 and has the following structure:

Field	Type	Description
queue_size	i4	Number of messages for a single user
penalty_remain	i4	Time in milliseconds after which the next message from this user will be successfully received.
message	c128	Error text message

Please pay attention to the two details:

1. 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.
2. A reject message with 99 type can be sent in a reply to any user's message.

3.3.4. Cancel On Disconnect

The Trading System SPECTRA provides a client connection control feature ('Cancel On Disconnect' or 'COD'). This option allows to automatically cancel some client's orders (anonymous orders without specified expiration time) on disconnect.

To enable/disable the 'COD' option, a trading participant should apply the appropriate request to the Client Center. The 'COD' option will be enabled for the ID (p2login) belonging to the trading participant.

When an ID connects to the trading system having the 'Cancel On Disconnect' option enabled, the trading system starts to monitor its connection activity in the 'COD' mode.

The connection activity monitoring algorithm is as following:

- If the 'COD' mode is enabled for the client, the system monitors the client's activity on transaction layer. Each and every client's command or message registered by the Trading System is considered as activity, no matter whether it was complete or not.
- If the client does not send a single message, or does not reconnect to the Trading System after loosing connection within the time period specified (now is 20 seconds), all they active orders are automatically cancelled.

The order cancellation conditions are as following:

- A client has not sent any transaction within the specified time limit.
- Client application has lost connection to the Router. Orders will be cancelled after reaching the specified time limit.

- Router has lost connection to the Access Server. Active orders will be cancelled after reaching the specified time limit.
- Access Server has lost connection to the Trading System or become unable to operate properly due to an error. All active orders of all clients connected to this Access Server will be cancelled after reaching the specified time limit.
- There may occur an issue when FIX server or an API clients access server connected to the Trading System via gateway becomes unable to operate properly: it loses connections to a client but does not inform the Trading System about it. The Trading System cannot handle such issues; if occurs, the issue should be resolved on the client side.

All orders added by clients with COD-mode enabled are cancelled when the clearing session ends and when the Trading System has been restored after a failure.

The orders cancelled via the 'Cancel On Disconnect' option are marked with a special status (field 'xstatus') in the table.

If clients need to simulate their transaction activity, they should sent command ‘CODHeartbeat (msgid=10000) into the Trading System at least once per at least once per 10 seconds. The command structure is as following:

Field	Type	Details
seq_number	i4	Heartbeat-message number (not used in the current version).

The command is not included into transaction fee.

The connection control service does not send replies to the Heartbeat messages, so that clients should set 0 (no reply expected) when calling the message sending function: (cg_pub_post(pub, msgptr, 0). Any attempt to call the function 'cg_pub_post' with 'CG_PUB_NEEDREPLY' on sending a Heartbeat message will result the error 'CG_MSG_P2MQ_TIMEOUT'.

3.3.5. Replication stream sets for different login subtypes

Depending on the user login subtype (main, viewing, transactional), there are different sets of replication streams each login receives.

Replication streams set for the main login subtype:

- FORTS_CLR_REPL - Clearing information
- FORTS_FEERATE_REPL - Precise Exchange fee rates
- FORTS_BROKER_FEE_PARAMS_REPL - Parameters for calculating the brokerage fee
- FORTS_BROKER_FEE_REPL - Brokerage fees
- FORTS_FEE_REPL - Exchange fees and penalties
- FORTS_PROHIBITION_REPL - Prohibitions
- FORTS_REFDATA_REPL - Reference and session information
- FORTS_TRADE_REPL - User's orders and trades
- FORTS_MM_REPL - Information on MM's obligations
- FORTS_USERORDERBOOK_REPL - User orders: order book snapshot
- FORTS_FORECASTIM_REPL - Risk forecast after limits extension
- FORTS_INFO_REPL - Reference information
- FORTS_PART_REPL - Information about limits
- FORTS_POS_REPL - Information on positions
- FORTS_TNPENALTY_REPL - Information about Transaction fees
- FORTS_VM_REPL - Variation margin and premium
- FORTS_DEALS_REPL - Anonymous trades
- FORTS_COMMON_REPL - Market fundamentals
- FORTS_VOLAT_REPL - Online volatility information
- MOEX_RATES_REPL - Online currency rates
- RTS_INDEX_REPL - Online indices
- FORTS_RISKINFOBLACK_REPL - Risk parameters for the Black-Scholes model
- FORTS_RISKINFOBACH_REPL - Risk parameters for the Bachelier model

- FORTS_USER_REPL - Users
- FORTS_REJECTEDORDERS_REPL - Register of orders rejected during the clearing
- FORTS_RMT_REPL - Collateral without orders and current operational risk
- FORTS_SESSIONSTATE_REPL - Current trading day status.
- FORTS_INSTRUMENTSTATE_REPL - Statuses of instruments for the current trading day.
- FORTS_SECURITYGROUPSTATE_REPL - Group status of instruments for the current trading day.

Additional replication streams received (market data source: aggregated orderbook):

- FORTS_AGGR5_REPL, FORTS_AGGR20_REPL, FORTS_AGGR50_REPL - Aggregated order-book streams

Additional replication streams received (market data source: full orderbook):

- FORTS_ORDLOG_REPL - Anonymous orders
- FORTS_ORDBOOK_REPL - Depersonalized order book snapshot

Replication streams set for viewing login subtype:

- FORTS_CLR_REPL - Clearing information
- FORTS_FEERATE_REPL - Precise Exchange fee rates
- FORTS_BROKER_FEE_PARAMS_REPL - Parameters for calculating the brokerage fee
- FORTS_BROKER_FEE_REPL - Brokerage fees
- FORTS_FEE_REPL - Exchange fees and penalties
- FORTS_PROHIBITION_REPL - Prohibitions
- FORTS_REFDATA_REPL - Reference and session information
- FORTS_TRADE_REPL - User's orders and trades
- FORTS_MM_REPL - Information on MM's obligations
- FORTS_USERORDERBOOK_REPL - User orders: order book snapshot
- FORTS_FORECASTIM_REPL - Risk forecast after limits extension
- FORTS_INFO_REPL - Reference information
- FORTS_PART_REPL - Information about limits
- FORTS_POS_REPL - Information on positions
- FORTS_TNPENALTY_REPL - Information about Transaction fees
- FORTS_VM_REPL - Variation margin and premium
- FORTS_DEALS_REPL - Anonymous trades
- FORTS_COMMON_REPL - Market fundamentals
- FORTS_VOLAT_REPL - Online volatility information
- MOEX_RATES_REPL - Online currency rates
- RTS_INDEX_REPL - Online indices
- FORTS_RISKINFOBLACK_REPL - Risk parameters for the Black-Scholes model
- FORTS_RISKINFOBACH_REPL - Risk parameters for the Bachelier model
- FORTS_USER_REPL - Users
- FORTS_REJECTEDORDERS_REPL - Register of orders rejected during the clearing
- FORTS_RMT_REPL - Collateral without orders and current operational risk
- FORTS_SESSIONSTATE_REPL - Current trading day status.
- FORTS_INSTRUMENTSTATE_REPL - Statuses of instruments for the current trading day.

- FORTS_SECURITYGROUPSTATE_REPL - Group status of instruments for the current trading day.

Additional replication streams received (market data source: aggregated orderbook):

- FORTS_AGGR5_REPL, FORTS_AGGR20_REPL, FORTS_AGGR50_REPL - Aggregated order-book streams

Additional replication streams received (market data source: full orderbook):

- FORTS_ORDLOG_REPL - Anonymous orders
- FORTS_ORDBOOK_REPL - Depersonalized order book snapshot

Replication streams set for the transactional login subtype:

- FORTS_CLR_REPL - Clearing information
- FORTS_FEERATE_REPL - Precise Exchange fee rates
- FORTS_BROKER_FEE_PARAMS_REPL - Parameters for calculating the brokerage fee
- FORTS_BROKER_FEE_REPL - Brokerage fees
- FORTS_FEE_REPL - Exchange fees and penalties
- FORTS_PROHIBITION_REPL - Prohibitions
- FORTS_REFDATA_REPL - Reference and session information
- FORTS_TRADE_REPL - User's orders and trades
- FORTS_MM_REPL - Information on MM's obligations
- FORTS_USERORDERBOOK_REPL - User orders: order book snapshot
- FORTS_FORECASTIM_REPL - Risk forecast after limits extension
- FORTS_INFO_REPL - Reference information
- FORTS_PART_REPL - Information about limits
- FORTS_POS_REPL - Information on positions
- FORTS_TNPENALTY_REPL - Information about Transaction fees
- FORTS_VM_REPL - Variation margin and premium
- FORTS_USER_REPL - Users
- FORTS_REJECTEDORDERS_REPL - Register of orders rejected during the clearing
- FORTS_RMT_REPL - Collateral without orders and current operational risk
- FORTS_SESSIONSTATE_REPL - Current trading day status.
- FORTS_INSTRUMENTSTATE_REPL - Statuses of instruments for the current trading day.
- FORTS_SECURITYGROUPSTATE_REPL - Group status of instruments for the current trading day.

3.3.6. Changing user password for the Trading System

A user is able to change their authentication password for the Trading System by one of the following methods:

- using utility **change_password** (described below);
- create their own application for changing authentication password (for details, see the appropriate API object description in **cgate_en.pdf**, section 'Password change protocol objects') and send message **ChangePassword** (for details, see here) into the Trading System.

Requirements for password complexity and originality:

- The password must be at least 8 characters long and meet three of the four requirements:
  - contain latin lowercase letters (from ‘a’ to ‘z’);
  - contain latin capital letters (from ‘A’ to ‘Z’);
  - contain numerals (from ‘0’ to ‘9’);
  - contain special or non-alphanumeric characters (@,+,_, etc).

- Password must not contain 3 or more repeating characters.
- The password must not match one of the previously used passwords.

3.3.6.1. Utility 'change_password'

The 'change_password' utility is designed to change the user's password in the trading system. The utility receives the old and new password of the user, sends them to the TS Spectra, and receives a response about the successful (or not) change of the user's password in the trading system. The utility uses a protocol that provides secure data transmission, the password and user login in clear text are not transmitted over the network.

The utility is a console application launched from the command line using the 'change_password.exe' executable file. Location: for Windows environment - C:\Moscow Exchange\SpectraCGate\bin\change_password.exe; for Linux environment - \opt\moex\cgate\bin \change_password.

Possible parameters to run the utility:

--app_name	Application name. Optional setting;
--local_pass	Password for the local connection to the router. Optional setting;
--host	Router IP address. Optional setting, the default value is 127.0.0.1
--port	Router port. Optional setting, the default value is 4001
--ini	INI file containing logging settings. Optional setting. If no INI file specified, the data will be output to console.

Command line example:

```
C:\Moscow Exchange\SpectraCGate\bin\change_password.exe --port=4001
```

To change the password, follow these steps:

- Run the utility.
- Enter old and new password from console.
- Press 'Enter'.

The utility returns '0' if the password change succeed, and '1' in case of any error.

Please note that receiving a successful response means changing the user's password in the trading system, while the authorization of the current router connection does not change. To authorize the router with a new password, you need to change the password in the ini-file of the router and restart the router.

3.3.6.2. One-time passwords

When registering in the system or resetting an old password, the member is issued a login and a one-time password, which the member must change to a permanent one within 7 days using one of the listed methods. After this period, the one-time password becomes invalid. A member authorized in the system with a one-time password cannot perform any operations except for changing the password.

3.3.7. Partitioning of the matching

TS SPECTRA supports division (partitioning) of the traded instruments into groups and trading them separately on several independent orders matching modules. Wherein each matching module processes its own group of instruments.The belonging of an instrument to a group (matching) is determined by the underlying asset code (base_contract_code) of the instrument.

Broadcast trading data is also separate and independent, own replication streams are assigned to each of the matching modules. Matching affiliation of the replication streams is determined by the postfix _MATCH $ {id} in the stream name, where $ {id} is the ID of the matching module. For example, the 'FORTS_TRADE_REPL_MATCH1' stream is the user's orders and trades on futures instruments that are processed on MATCH1.

These streams are broadcast separately for each matching (have postfix _MATCH${id}):

- FORTS_DEALS_REPL
- FORTS_FEE_REPL
- FORTS_FORECASTIM_REPL
- FORTS_COMMON_REPL
- FORTS_TRADE_REPL
- FORTS_ORDBOOK_REPL
- FORTS_USERORDERBOOK_REPL

- FORTS_AGGR5_REPL, FORTS_AGGR20_REPL, FORTS_AGGR50_REPL
- FORTS_ORDLOG_REPL
- FORTS_POS_REPL
- FORTS_VM_REPL
- FORTS_VOLAT_REPL

The table 'instr2matching_map' is broadcasted to determine the correspondence between the instrument and the matching on which it is processed, in 'FORTS_REFDATA_REPL' stream. The table 'instr2matching_map' has following fields:

- base_contract_id - underlying contract ID;
- matching_id - matching ID.

Binding of instruments to matchings may change when trading day changes.

New algorithm for receiving trade data

- Define 'base_contract_code' for 'isin_id' according to the tables 'fut_sess_contents' / 'opt_sess_contents'.
- Define 'base_contract_id' for 'base_contract_code' according to the tables 'fut_vcb' / 'opt_vcb'.
- Define matching ID by 'base_contract_id' in the table 'instr2matching_map'.
- Open streams with matching _MATCH${id} for getting instrument trade data.

There is only one orders matching module In the TS SPECTRA version 6.3,and old replication streams (without partitioning by matchings) are left for backward compatibility. But old streams will be deleted in future versions of the system, So, we recommend that users rebuild their systems to work with new data streams. Also two new commands 'MoveOrder' (msgid=438) and 'DelOrder' (msgid=436) was added to TS version 6.3. These commands should be used to move and delete orders for futures and options in the trading system with several matchings.

3.3.8. Stream types

Types of data streams:

- **'Reliable' (R)** - The data published in such streams is relevant, reliable and not subject to change. Any change is a force majeure related to an emergency situation on the Exchange. The trading member can fully rely on data from such flows when making decisions.
- **'Almost Reliable' (AR)** - Data requires reconciliation with reports. Usually data in such streams is not subject to change, but there may be rare situations where the final values published in the reports differ from online data. For example, the settlement price can be adjusted by the NCC (this situation is provided for by regulatory documents). The trading member can rely on data from such flows, taking into account that it may be necessary to adjust the data obtained on the basis of automatic reconciliation with reports.
- **'Informational' (I)** - Data that the trading member cannot rely on as the sole source when he makes a decision. Data from such flows should be used with caution, if possible, by conducting a weighted comparison with similar data obtained in another way. An example of such data is volatility data, which is estimates and depends on the model used and calculation methodology.

The table below shows the gradation of streams by type:

Stream name	Description	Type
FORTS_TRADE_REPL	User's orders and trades	R
FORTS_ORDLOG_REPL	Anonymous orders	R
FORTS_DEALS_REPL	Anonymous trades	R
FORTS_USERORDERBOOK_REPL	User orders: order-book snapshot	R
FORTS_ORDBOOK_REPL	Depersonalized order-book snapshot	R
FORTS_PROHIBITION_REPL	Prohibitions	R
FORTS_REFDATA_REPL	Reference and session information	R
RTS_INDEX_REPL	Online indices	R
FORTS_INFO_REPL	Reference information	R
FORTS_USER_REPL	Users	R
FORTS_REJECTEDORDERS_REPL	Register of orders rejected during the clearing	R
FORTS_FEE_REPL	Exchange fees and penalties	AR
FORTS_FEERATE_REPL	Precise Exchange fee rates	AR
FORTS_CLR_REPL	Clearing information	AR

Stream name	Description	Type
FORTS_COMMON_REPL	Market fundamentals	I
FORTS_AGGR##_REPL	Aggregated order-book	I
FORTS_POS_REPL	Information on positions	I
FORTS_PART_REPL	Information about limits	I
FORTS_MM_REPL	Information on MM's obligations	I
FORTS_VM_REPL	Variation margin and premium	I
FORTS_VOLAT_REPL	Online volatility information	I
FORTS_TNPENALTY_REPL	Information about transaction fees	I
MOEX_RATES_REPL	Online currency rates	I
FORTS_FORECASTIM_REPL	Risk forecast after limits extension	I
FORTS_RISKINFOBLACK_REPL	Risk parameters for the Black-Scholes model	I
FORTS_RISKINFOBACH_REPL	Risk parameters for the Bachelier model	I
FORTS_RMT_REPL	Collateral without orders and current operational risk	I
FORTS_SESSIONSTATE_REPL	Current trading day status	I
FORTS_INSTRUMENTSTATE_REPL	Statuses of instruments for the current trading day	I
FORTS_SECURITYGROUPSTATE_REPL	Group status of instruments for the current trading day	I

3.3.9. Limiting the number of simultaneously open replication streams from one PLAZA II connection

The system has a limit on the number of simultaneous subscriptions to one PLAZA II (Cgate) stream from one gateway login - no more than 20. If this limit is exceeded, each subsequent attempt to subscribe to a stream will end with the error code ERROR:TOO MANY CONNECTIONS, which is reflected in the Cgate operation log.

3.3.10. Displaying information about treasurers' logins in trades/orders

In some cases, clients may see operations in their order/trade tables executed under unknown logins with the format **_admin4kazn**. These are treasurers' logins used by MOEX/NCC staff when performing procedures such as order cancellation, settlement trade execution, etc. The appearance of such orders/trades under a member’s account indicates that the member has issues with their financial status and should contact MOEX/NCC for assistance.

Two types of logins are used:

1. Login with the mask **ftxxrm_admin4forts** – a login with the role of Trading Administrator (Broker) at administrator level. It is used by MOEX staff. A login is applied in the following processes:
   - Setting constraints on brokerage firm, 7-digit client code.
   - Order cancellation (trading member's request, suspension, liquidation netting, settlement trades).
   - Other operations specified by MOEX procedures.
2. Login with the mask **ftxxrm_admin4kazn** – a login with the role of NCC Treasury at administrator level. It is used by NCC staff. A login is applied in the following processes:
   - Setting constraints on settlement account.
   - Executing trades on behalf of NCC (liquidation netting, settlement trades).
   - Other operations specified by NCC procedures.

3.4. Handling abnormal situations

3.4.1. Recovery on loss of connection with Exchange servers

In the standard configuration of PLAZA II gate, there are four TCP-connections to the Exchange servers:

- Connection for sending requests and commands
- Connection for receiving the main market data such as aggregated order-books streams and the streams 'FORTS_ORDLOG_REPL', 'FORTS_DEALS_REPL', 'FORTS_TRADE_REPL' and 'FORTS_COMMON_REPL'.
- Connection for receiving auxiliary and reference streams

- Connection for receiving snapshots (at the first connection or when recovering after loss of connection)

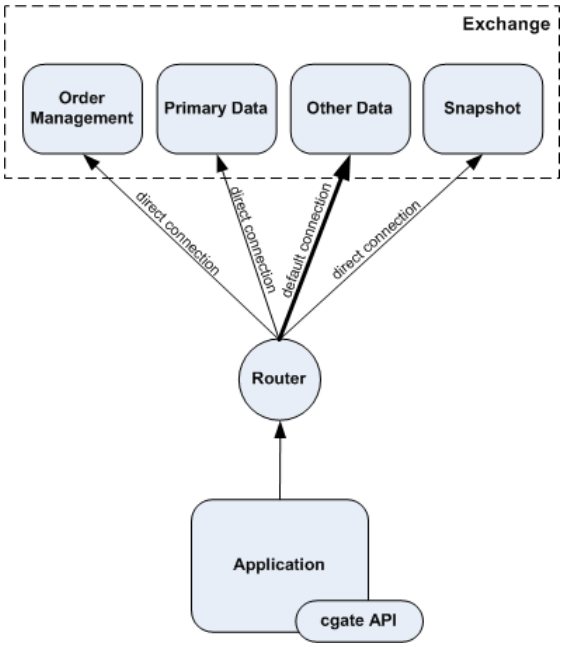

Figure 18. Connection scheme

In order to obtain stability, the trading system uses load balancing method to connect clients to the least loaded server at the moment.

3.4.1.1. Connection loss detection

P2MQRouter software handles all TCP-connections, with settings specified in the INI file where connection 'Other Data' is specified as the default outcoming connection and the other connections are specified as outcoming direct connections. This structure is used as the standard to establish connection directly to the Exchange server farm, but connection via a Brokerage Firm server server may require a different structure; in this case, clients should apply to the server's owners for more details about connection.

The P2MQRouter software also handles connection recovery in case of loss of connection. After disconnecting, P2MQRouter starts attempting to reestablish the connection periodically in accordance with the specified time value, while the client software is not able to interfere with the process. P2MQRouter status then changes from 'ROUTER_CONNECTED' to 'ROUTER_RECONNECTING' by receiving the appropriate notifications from object 'connection', and this is a way for client to check whether connection is still active or not.

The CGate library acts in the following way:

- When the loss of connection to the incoming request processing gateway occurs, it is detected directly on the moment of receiving the TCP-connection error. All the 'publisher' objects concerned go to the error state.
- When the loss of connection for receiving the main market data occurs, it is detected within 30 seconds. All the 'listener' objects concerned go to the error state.

All object in error state should be released. After that, it is necessary to try to reopen them anew periodically, for example, once in a few seconds.

3.4.1.2. Timeouts handling when commands/transactions are sent

When working with a 'p2mq' type publisher and 'p2mqreply' listener, client code can receive a 'CG_MSG_P2MQ_TIMEOUT' response code. This code signals that response has not been received to the message, a link to which is transmitted along with the response code within a user-specified time period. Response of "timeout" type indicates connection problems or other infrastructural problems on the user or the exchange side. This situation causes uncertainty: it is impossible to recognize whether the sent message was successfully processed by the system or not. To minimize the emergence of uncertainties, it is necessary to set large timeouts in the code - about 60 seconds.

When "timeout" response receiving, it is recommended to have a procedure for verifying the sent message:

1. It is recommended to verify that the 'p2mq' publisher and 'p2mqreply' listener pair have not went into an error state. If the publisher or the listener stay failed, perform their closure and deletion, then reinitialize them.
2. It is recommended to always specify the external order number (ext_id parameter) for order management commands. In case of "timeout" sent in response to the 'AddOrder' or 'MoveOrder' command, a possible safe reaction is to delete the order using the 'DelUserOrders' command with 'ext_id' specified.
3. If "timeout" occurred when deleting/canceling an order, it is assumed that the order was NOT deleted. It is necessary to repeat the deletion after the connection is re-established or use a CS connected to the alternative data center.

4. Analyze the data of 'orders_log' table of 'FORTS_TRADE_REPL' stream for availability or non-availability orders or cancelled orders.
5. In case of "timeout" sent in response to a non-trading command, it is recommended to analyze for applied changes a corresponding table state in the replication stream.

3.4.1.3. Recovery algorithm

In general, the connection recovery algorithm is as follows:

- After start-up, try to open connection to P2MQRouter periodically;
- When the router is reconnected to the PLAZA II network, the object 'connection' will go to the ACTIVE state;
- Open the necessary streams. To make it faster, it is recommended to receive data starting from the last update. When opening a stream, you should use the 'repl state' value received on closing the stream; also, you can directly specify revision numbers for tables and scheme life number by using that of the last received data.
- Recover the list of active orders (see below)
- Register 'publisher' for orders and commands.

The table below contains the recommended methods for recovering data depending on the stream:

Stream (table) name	Information type	Recovery method
FORTS_TRADE_REPL • orders_log	Own orders activity log (futures and options)	List of active orders: • use stream 'FORTS_USERORDERBOOK_REPL' to receive snapshot, then open stream 'FORTS_TRADE_REPL' using the revision number specified in snapshot Orders activity log: • open 'FORTS_TRADE_REPL' starting from the last received revision number
FORTS_TRADE_REPL • multileg_orders_log	Own orders activity log (multileg orders)	Orders activity log: • open 'FORTS_TRADE_REPL' starting from the last received revision number
FORTS_ORDLOG_REPL • orders_log	Complete anonymous orders activity log (futures and options)	List of active orders: • use stream 'FORTS_ORDRBOOK_REPL' to receive snapshot, then open stream 'FORTS_ORDLOG_REPL' using the revision number specified in the snapshot Orders activity log: • open 'FORTS_ORDLOG_REPL' starting from the last received revision number
FORTS_ORDLOG_REPL • multileg_orders_log	Complete anonymous orders activity log (multileg orders)	Orders activity log: • open 'FORTS_ORDLOG_REPL' starting from the last received revision number
FORTS_DEALS_REPL • deal • multileg_deal FORTS_TRADE_REPL • user_deal • multileg_deal	Orders log (futures, options, multileg instruments)	Reopen the appropriate stream using the last received revision number or 'repl state' value received on closing the stream.
FORTS_COMMON_REPL	Market fundamentals	Reopen the stream anew
FORTS_AGGR##_REPL	Order books for futures and options. (### - depth of order book)	Reopen the appropriate stream anew
FORTS_REFDATA_REPL	Reference and session information	Quick method: • Reopen the stream using the last received revision number or 'repl state' value received on closing the stream.

Stream (table) name	Information type	Recovery method
		Allowable method: • Reopen the stream anew
FORTS_PART_REPL	Information on limits	Reopen the stream anew
FORTS_POS_REPL	Information on positions	Reopen the stream anew
FORTS_VM_REPL	Information on variation margin and premium	Reopen the stream anew
FORTS_VOLAT_REPL	Information on volatility and theoretical prices on options	Reopen the stream anew
RTS_INDEX_REPL	Exchange indices values	Reopen the stream anew

Upon recovery, it is very important to receive the lists of the client's current orders:

1. List of orders which are active during the recovery procedure period
2. Orders activity log during the connection loss period.

For the first case, you should receive the order-book snapshot ('FORTS_USERORDERBOOK_REPL'). The orders missed in the snapshot have been either already matched or cancelled during the connection loss period.

For the second case, you should receive your own orders activity log (the table 'orders_log' of the streams 'FORTS_TRADE_REPL', also, the table 'multileg_orders_log' of the stream 'FORTS_TRADE_REPL') covering the connection loss period. To do this, you should open the appropriate stream using revision number of the last record actually received before the loss of connection occurred. Every order activity happened during the connection loss period will be recorded in these tables. Changing the stream state to 'ONLINE' indicates that all orders activity data have been successfully received.

Note: The recovering procedure described above can be also used for the late start connection.

3.4.1.4. General recommendations

In general case, to minimize possibility of loss of connection, the Exchange recommends to do the following:

- establish alternative connections
- obtain two client's IDs for the gateway, with the same user rights in order to have possibility to receive the same data by running two client applications simultaneously. Therefore, in case of any failure, you will be able to switch between two applications.

Alternatively, it is recommended to enable a feature in your application allowing to switch to another connection (a P2MQRouter connected to the Exchange servers using an alternative connection) in case of any failure.

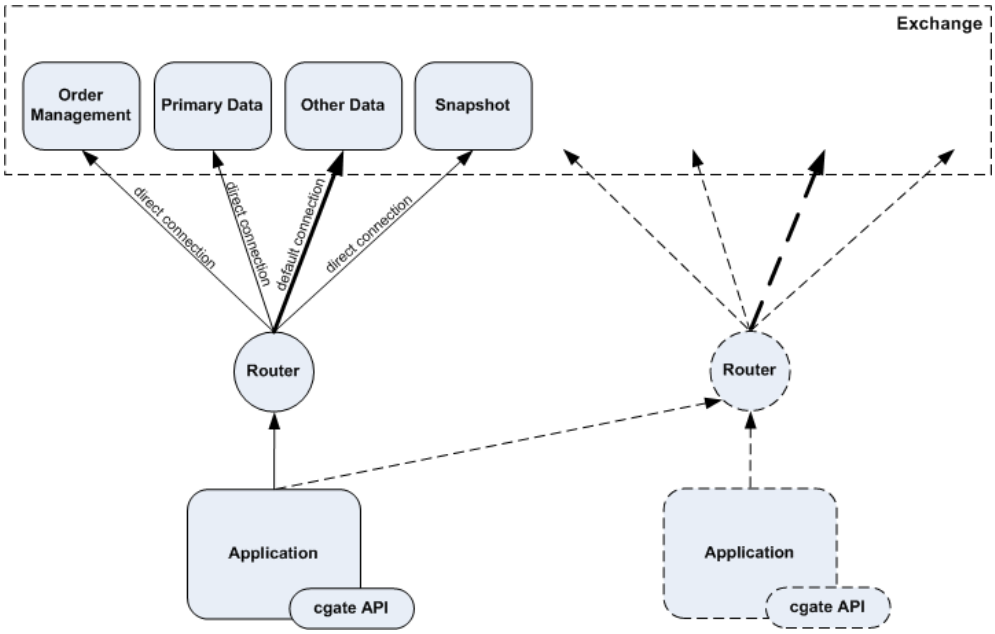

Figure 19. Channel duplication scheme

3.4.2. Recovery in case of the Exchange infrastructure failure

By the Exchange infrastructure failure we mean failures on the Exchange side caused by the Trading System kernel errors, or by errors in market data generating services. Then, as a rule, the services halt and restart.

3.4.2.1. Data cleanup by streams

In case of any routine maintenance, normal or abnormal service restarts on the Exchange side, or after reestablishing connection to a client, the publishing services send out notifications about obsolete data cleanup before sending the current snapshot to clients.

There are two types of data cleanup notifications:

- CG_MSG_P2REPL_CLEARDELETED - by every table, with use of revision number. The notification gives client the order to cleanup all records with the 'replRev' value smaller than the one in notification. In order to optimise data transfer, the notification may have a revision number value as 'MAX(int64)'. This means that client should cleanup all data from the specified table, as the entire table will be transferred anew.
- CG_MSG_P2REPL_LIFENUM - for the entire replication stream, using the new stream life number. This notification means, that data have been significantly changed since the last connection. Client should cleanup all data in all tables. All data will be transferred anew.

3.4.2.2. Possible data change in case of abnormal work of publishing services

In normal work mode, including routine works at non-trading time, when opening or reopening any replication stream except those related to history of orders and trades ('FORTS_TRADE_REPL', 'FORTS_ORDLOG_REPL' and 'FORTS_DEALS_REPL'), a client may receive both 'CG_MSG_P2REPL_CLEARDELETED' or 'CG_MSG_P2REPL_LIFENUM' notification types, and should process them correctly.

In normal work mode, for the streams related to history of orders and trades (see above), the notification CG_MSG_P2REPL_LIFENUM' is sent only in case of system version change, after the testing-mode trades, in order to make clients cleanup the user data. The notification 'CG_MSG_P2REPL_CLEARDELETED' has the 'replRev' value for the first available order or trade at the moment.

A 'CG_MSG_P2REPL_LIFENUM' with a new stream life number during a trading session indicates a severe failure in the Trading System, so the system is to resend data on orders and trades which could be already delivered to clients.

Additionally, there are some other information channels (the Exchange web site, etc.), where information about possible data issues (whether the data already delivered to clients were affected by the last data correction or not) will be posted. This includes information about possible system rollback to the state it was before the failure, along with the last number of order and trade available to client after the system restart.

4. Replication scheme FORTS_PUBLIC

4.1. Stream FORTS_TRADE_REPL - User's orders and trades (Type=R)

4.1.1. Data scheme

Tables:

- orders_log - Log of operations with orders
- multileg_orders_log - Log of operations with multileg orders
- user_deal - User trades
- user_multileg_deal - User’s multileg orders trades
- heartbeat - Server times table
- sys_events - table of events

4.1.1.1. Table orders_log: Log of operations with orders

Table 8. Fields of table orders_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
public_order_id	i8	Order ID (for iceberg order – ID of its visible part)
sess_id	i4	Trading day ID
isin_id	i4	Instrument unique ID
public_amount	i8	The number of contracts in the operation (for iceberg order - the number of contracts in the operation for its visible part)
public_amount_rest	i8	The remaining number of contracts in the order (for iceberg order – the remaining number of contracts in its visible part)
id_deal	i8	Deal ID for this operation
xstatus	i8	Extended order's status
xstatus2	i8	Extension for orders statuses (in addition to the ‘xstatus’ field)
price	d16.5	Price
moment	t	Order status changing time
moment_ns	u8	Order status changing time, nanoseconds since Unix epoch, UTC
dir	i1	Direction
public_action	i1	Type of operation with the order (for iceberg order – type of operation with its visible part)
deal_price	d16.5	Price of the deal
client_code	c7	Client code
login_from	c20	Login of the user who has entered the order
comment	c20	Trader's comment
ext_id	i4	External ID number. It is added to orders, trades
broker_to	c7	SPECTRA code of the company to whom the negotiated order is addressed
broker_to_rts	c7	RTS code of the company to whom the negotiated order is addressed
broker_from_rts	c7	RTS code of the company who has entered the order
date_exp	t	Order's expiration date
id_ord1	i8	ID number of the first order
aspref	i4	Client ID. For orders added by SMA login - MASTER login ID.
private_order_id	i8	Order ID (for iceberg order – ID of the entire order)
private_amount	i8	The number of contracts in the operation (for iceberg order - the number of contracts in the operation for the entire order)

Field	Type	Description
private_amount_rest	i8	The remaining number of contracts in the order (for iceberg order – the remaining number of contracts in the entire order)
variance_amount	i8	Variance amplitude for a random addition for the pop-up part (in contracts)
disclose_const_amount	i8	Number of instrument units in the pop-up part of the iceberg order
private_action	i1	Type of operation with the order (for iceberg order – type of operation with the entire order)
reason	i4	The flag (reason) of the order submitted for the making of the settlement trade of obligations.
match_ref	c10	Identical text values entered by both trading parties to match negotiated orders
compliance_id	c1	Order adding method
prevorder_id	i8	Order ID modified by the MoveOrder operation. Filled only in new orders generated upon successful execution of MoveOrder.

Notes:

- Field xstatus is a bit mask. For the complete list of all possible values of field 'status' please refer to section Flags applied to orders and trades.
- Field dir can take the following values:

  1 Buy

  2 Sell
- Field public_action can take the following values

  0 Order cancelled

  1 Order added

  2 Order is exercised in the trade
- Field id_ord1 contains the initial order ID number, i.e. the ID number which was assigned to order before the order has once been relisted
- Field 'private_action' ('action') can take the following values:

  0 Order cancelled

  1 Order added

  2 Order is exercised in the trade

  3 The order was added by appearance of a new visible part of the iceberg
- Field 'reason' can take the following values:

  0 Regular order

  4 Balancing Derivatives contracts entered into with the Non-defaulting Clearing Member without submitting orders

  6 Closing Derivatives contracts entered into under the cross-default procedure

  7 Closing Derivatives Contracts entered into upon non-fulfillment of the Margin Call

  8 Closing Derivatives contracts entered into in into upon non-fulfillment of the Delivery Obligation on the deliverable Derivatives contracts for precious metals.

  11 Balancing trade for the obligations of the Defaulting Clearing Member

  100 Other
- Field 'compliance_id' can take the following values:

  " " (space or empty string) Not filled in/Not specified

  M Manual input

  S As a result of the conditional request (stop-loss order)

  R As a result of the robot algorithm work

  A As a result of the auto-following algorithm

D	Covering a position as a result of an unexecuted Margin Call

4.1.1.2. Table multileg_orders_log: Log of operations with multileg orders

Table 9. Fields of table multileg_orders_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
public_order_id	i8	Order ID (for iceberg order – ID of its visible part)
sess_id	i4	Trading day ID
isin_id	i4	Multileg instrument ID
public_amount	i8	The number of contracts in the operation (for iceberg order - the number of contracts in the operation for its visible part)
public_amount_rest	i8	The remaining number of contracts in the order (for iceberg order – the remaining number of contracts in its visible part)
id_deal	i8	Deal ID for this operation
xstatus	i8	Extended order's status
xstatus2	i8	Extension for orders statuses (in addition to the ‘xstatus’ field)
price	d16.5	Price. *The field is not used.*
moment	t	Order status changing time
moment_ns	u8	Order status changing time, nanoseconds since Unix epoch, UTC
dir	i1	Direction
public_action	i1	Type of operation with the order (for iceberg order – type of operation with its visible part)
deal_price	d16.5	First leg price of a filled trade
rate_price	d16.5	Rate price. *The field is not used.*
swap_price	d16.5	Swap price
client_code	c7	Client code
login_from	c20	Login of the user who has entered the order
comment	c20	Trader's comment
ext_id	i4	External ID number. It is added to orders, trades
broker_to	c7	SPECTRA code of the company to whom the negotiated order is addressed
broker_to_rts	c7	RTS code of the company to whom the negotiated order is addressed
broker_from_rts	c7	RTS code of the company who has entered the order
date_exp	t	Order's expiration date
id_ord1	i8	ID number of the first order
aspref	i4	Client ID. For orders added by SMA login - MASTER login ID.
private_order_id	i8	Order ID (for iceberg order – ID of the entire order)
private_amount	i8	The number of contracts in the operation (for iceberg order - the number of contracts in the operation for the entire order)
private_amount_rest	i8	The remaining number of contracts in the order (for iceberg order – the remaining number of contracts in the entire order)
variance_amount	i8	Variance amplitude for a random addition for the pop-up part (in contracts)
disclose_const_amount	i8	Number of instrument units in the pop-up part of the iceberg order
private_action	i1	Type of operation with the order (for iceberg order – type of operation with the entire order)
reason	i4	The flag (reason) of the order submitted for the making of the settlement trade of obligations.
match_ref	c10	Identical text values entered by both trading parties to match negotiated orders
compliance_id	c1	Order adding method

Notes:

- Field xstatus is a bit mask. For the complete list of all possible values of field 'status' please refer to section Flags applied to orders and trades.
- Field dir can take the following values:

  1 Buy

  2 Sell
- Field public_action can take the following values

  0 Order cancelled

  1 Order added

  2 Order exercised in a trade
- Field 'private_action' ('action') can take the following values:

  0 Order cancelled

  1 Order added

  2 Order is exercised in the trade

  3 The order was added by appearance of a new visible part of the iceberg
- Field 'reason' can take the following values:

  0 Regular order

  4 Balancing Derivatives contracts entered into with the Non-defaulting Clearing Member without submitting orders

  6 Closing Derivatives contracts entered into under the cross-default procedure

  7 Closing Derivatives Contracts entered into upon non-fulfillment of the Margin Call

  8 Closing Derivatives contracts entered into in into upon non-fulfillment of the Delivery Obligation on the deliverable Derivatives contracts for precious metals.

  11 Balancing trade for the obligations of the Defaulting Clearing Member

  100 Other
- Field 'compliance_id' can take the following values:

  " " (space or empty string) Not filled in/Not specified

  M Manual input

  S As a result of the conditional request (stop-loss order)

  R As a result of the robot algorithm work

  A As a result of the auto-following algorithm

  D Covering a position as a result of an unexecuted Margin Call

4.1.1.3. Table user_deal: User trades

Table 10. 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 day ID
isin_id	i4	Instrument unique ID
id_deal	i8	Deal ID number
id_deal_multileg	i8	Deal ID number for multileg deals
id_repo	i8	Deal ID number of the other leg
xpos	i8	Number of positions in the instrument in the market after the trade
xamount	i8	Volume, number of units of the instrument

Field	Type	Description
public_order_id_buy	i8	The buyer's order ID (for iceberg order – ID of its visible part)
public_order_id_sell	i8	The seller's order ID (for iceberg order – ID of its visible part)
price	d16.5	Price
moment	t	Time when the deal was made
moment_ns	u8	Time when the deal was made, nanoseconds since Unix epoch, UTC
nosystem	i1	Sign of non-system deal
xstatus_buy	i8	Status of the trade from the buyer's side
xstatus_sell	i8	Status of the trade from the seller's side
xstatus2_buy	i8	Extension for trades statuses (in addition to the ‘xstatus_buy’ field)
xstatus2_sell	i8	Extension for trades statuses (in addition to the ‘xstatus_sell’ field)
ext_id_buy	i4	External ID number from the buyer's order
ext_id_sell	i4	External ID number from the seller's order
code_buy	c7	Buyer's code
code_sell	c7	Seller's code
comment_buy	c20	Comment from the buyer's order
comment_sell	c20	Comment from the seller's order
fee_buy	d26.2	Fee of the buyer's deal
fee_sell	d26.2	Fee of the seller's deal
login_buy	c20	Login of the buyer user
login_sell	c20	Login of the seller user
code_rts_buy	c7	RTS code of the buyer company
code_rts_sell	c7	RTS code of the seller company
private_order_id_buy	i8	The buyer's order ID (for iceberg order – ID of the entire order)
private_order_id_sell	i8	The seller's order ID (for iceberg order – ID of the entire order)
reason_buy	i4	The flag(reason) of the buyer's settlement trade.
reason_sell	i4	The flag(reason) of the seller's settlement trade.

Notes:

- Fields code_sell, comment_sell, ext_id_sell, login_sell, code_rts_sell, fee_sell, code_buy, comment_buy, ext_id_buy, login_buy, code_rts_buy, fee_buy, are filled with info only for "own" deals.
- Fields xstatus_sell and xstatus_buy are bit masks (for details see Flags applied to orders and trades)
- For technical trades that are results of trades with multileg instruments filed nosystem always equals 1, regardless the fact whether the trade is regular or negotiated one. To define whether the initial trade is regular the sign of the field nosystem should correspond to the record in the table multileg_deal.
- The field id_repo contains the deal ID of the other leg. It contains deal ID of the second leg for the first leg, and deal ID of the first leg for the second one.
- Field id_deal_multileg contains code of the trade with multileg intrument, if this record is about technical trade. the field equals 0 if the trade is with an ordinary instrument.
- For all other (not client-related) trades, fields ‘xstatus_buy’ and ‘xstatus_sell’ may contain flags ‘NonQuote’, ‘ClearingTrade’, ‘Address’, ‘Strategy’.
- In exercise trades, field private_order_id_buy contains the request ID (option call). In exercise trades, field private_order_id_sell contains the request ID (option put).
- The fields fee_buy and fee_sell contain the estimated size of the limit blocked for the trade fee. Fee size must be viewed in the FORTS_FEE_REPL stream.
- The reason_buy and reason_sell fields can take the following values:

  0 Regular trade

  4 Balancing Derivatives contracts entered into with the Non-defaulting Clearing Member without submitting orders

  6 Closing Derivatives contracts entered into under the cross-default procedure

7 Closing Derivatives Contracts entered into upon non-fulfillment of the Margin Call

8 Closing Derivatives contracts entered into in into upon non-fulfillment of the Delivery Obligation on the deliverable Derivatives contracts for precious metals.

11 Balancing trade for the obligations of the Defaulting Clearing Member

100 Other

4.1.1.4. Table user_multileg_deal: User's multileg orders trades

Table 11. 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 day ID
isin_id	i4	Multileg instrument ID
isin_id_rd	i4	Instrument ID of the first leg
isin_id_rb	i4	Instrument ID of the second leg
duration	i4	The difference in calendar days between the dates of execution of two futures
id_deal	i8	Deal ID number for multileg deals
id_deal_rd	i8	Deal ID of the first leg
id_deal_rb	i8	Deal ID of the second leg
public_order_id_buy	i8	The buyer's order ID (for iceberg order – ID of its visible part)
public_order_id_sell	i8	The seller's order ID (for iceberg order – ID of its visible part)
xamount	i8	Volume, number of units of the instrument
price	d16.5	Price of the first part of multileg trade
rate_price	d16.5	Rate price
swap_price	d16.5	Swap price
moment	t	Time when the deal was made
moment_ns	u8	Time when the deal was made, nanoseconds since Unix epoch, UTC
nosystem	i1	Sign of non-system deal
xstatus_buy	i8	Extended status of the trade from the buyer's side
xstatus_sell	i8	Extended status of the trade from the seller's side
xstatus2_buy	i8	Extension for trades statuses (in addition to the 'xstatus_buy' field)
xstatus2_sell	i8	Extension for trades statuses (in addition to the 'xstatus_sell' field)
ext_id_buy	i4	External ID number from the buyer's order
ext_id_sell	i4	External ID number from the seller's order
code_buy	c7	Buyer's code
code_sell	c7	Seller's code
comment_buy	c20	Comment from the buyer's order
comment_sell	c20	Comment from the seller's order
login_buy	c20	Login of the buyer user
login_sell	c20	Login of the seller user
code_rts_buy	c7	RTS code of the buyer company
code_rts_sell	c7	RTS code of the seller company
private_order_id_buy	i8	The buyer's order ID (for iceberg order – ID of the entire order)
private_order_id_sell	i8	The seller's order ID (for iceberg order – ID of the entire order)
reason_buy	i4	The flag(reason) of the buyer's settlement trade.
reason_sell	i4	The flag(reason) of the seller's settlement trade.

Notes:

- Fields code_sell, comment_sell, ext_id_sell, code_rts_sell, fee_sell, code_buy, comment_buy, ext_id_buy, code_rts_buy, fee_buy, are filled with info only for "own" deals.
- Field rate_price contains 0 for the instruments traded in swap-price.
- The reason_buy and reason_sell fields can take the following values:

  0 Regular trade

  4 Balancing Derivatives contracts entered into with the Non-defaulting Clearing Member without submitting orders

  6 Closing Derivatives contracts entered into under the cross-default procedure

  7 Closing Derivatives Contracts entered into upon non-fulfillment of the Margin Call

  8 Closing Derivatives contracts entered into in into upon non-fulfillment of the Delivery Obligation on the deliverable Derivatives contracts for precious metals.

  11 Balancing trade for the obligations of the Defaulting Clearing Member

  100 Other

4.1.1.5. Table heartbeat: Server times table

Records in this table are added periodically by the trading system's core. It can be used for synchronization purposes (e.g. to check whether all the trades were received at specified moment of time). The table is insert-only, no modifications or deletions occur during trading session.

Table 12. 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

4.1.1.6. Table sys_events: table of events

Table 13. 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	Unique ID of the event
sess_id	i4	Trading day ID
event_type	i4	Type of the event
message	c64	Description of the event
server_time	t	Event time

Notes:

- Possible types of events

  event_type = 1
  message = "session_data_ready"
  All data from the clearing system have been loaded into the trading system

  event_type = 5
  message = "clearing_started"
  Start of the clearing session

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

  event_type = 17
  message = "clearing_trades_inserted"
  Clearing trades have been generated

  event_type = 18

message = "extension_of_limits_started_for_core"
Limits have been extended

event_type = 22
message = "discrete_auction_matching_completed"
Order matching completed and opening auction trades have been published

event_type = 23
message = "discrete_auction_add_order_started"
The start of accepting orders in the opening auction

event_type = 24
message = "discrete_auction_add_order_finished"
The finish of accepting orders in the opening auction

event_type = 36
message = "start_of_calendar_date"
Beginning of the calendar day

event_type = 37
message = "settlement_session_started"
Beginning of the settlement session

event_type = 38
message = "settlement_session_finished"
End of the settlement session

event_type = 47
message = "settlpricecalc_finished"
Settlement price has been fixed

4.2. Stream FORTS_ORDLOG_REPL – anonymous orders (Type=R)

4.2.1. Data scheme

Tables:

- orders_log - Log of operations with orders
- multileg_orders_log - Log of operations with multileg orders
- heartbeat - Server times table
- sys_events - table of events

4.2.1.1. Table orders_log: Log of operations with orders

Table 14. Fields of table orders_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
public_order_id	i8	Order ID (for iceberg order – ID of its visible part)
sess_id	i4	Trading day ID
isin_id	i4	Instrument unique ID
public_amount	i8	The number of contracts in the operation (for iceberg order - the number of contracts in the operation for its visible part)
public_amount_rest	i8	The remaining number of contracts in the order (for iceberg order – the remaining number of contracts in its visible part)
id_deal	i8	Deal ID for this operation
xstatus	i8	Extended order's status
xstatus2	i8	Extension for orders statuses (in addition to the ‘xstatus’ field)
price	d16.5	Price
moment	t	Order status changing time
moment_ns	u8	Order status changing time, nanoseconds since Unix epoch, UTC
dir	i1	Direction

Field	Type	Description
public_action	i1	Type of operation with the order (for iceberg order – type of operation with its visible part)
deal_price	d16.5	Price of the deal

Notes:

- Field xstatus is a bit mask. For the complete list of all possible values of field 'status' please refer to section Flags applied to orders and trades.
- Field dir can take the following values:

  1 Buy

  2 Sell
- Field public_action can take the following values

  0 Order cancelled

  1 Order added

  2 Order is exercised in the trade

4.2.1.2. Table multileg_orders_log: Log of operations with multileg orders

Table 15. Fields of table multileg_orders_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
public_order_id	i8	Order ID (for iceberg order – ID of its visible part)
sess_id	i4	Trading day ID
isin_id	i4	Instrument unique ID
public_amount	i8	The number of contracts in the operation (for iceberg order - the number of contracts in the operation for its visible part)
public_amount_rest	i8	The remaining number of contracts in the order (for iceberg order – the remaining number of contracts in its visible part)
id_deal	i8	Deal ID for this operation
xstatus	i8	Extended order's status
xstatus2	i8	Extension for orders statuses (in addition to the ‘xstatus’ field)
price	d16.5	Price. *The field is not used.*
moment	t	Order status changing time
moment_ns	u8	Order status changing time, nanoseconds since Unix epoch, UTC
dir	i1	Direction
public_action	i1	Type of operation with the order (for iceberg order – type of operation with its visible part)
deal_price	d16.5	First leg price of a filled trade
rate_price	d16.5	Rate price. *The field is not used.*
swap_price	d16.5	Swap price

Notes:

- Field xstatus is a bit mask. For the complete list of all possible values of field 'status' please refer to section Flags applied to orders and trades.
- Field dir can take the following values:

  1 Buy

  2 Sell
- Field public_action can take the following values

  0 Order cancelled

1 Order added

2 Order exercised in a trade

4.2.1.3. Table heartbeat: Server times table

Records in this table are added periodically by the trading system's core. It can be used for synchronization purposes (e.g. to check whether all the trades were received at specified moment of time). The table is insert-only, no modifications or deletions occur during trading session.

Table 16. 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

4.2.1.4. Table sys_events: table of events

Table 17. 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	Unique ID of the event
sess_id	i4	Trading day ID
event_type	i4	Type of the event
message	c64	Description of the event
server_time	t	Event time

Notes:

- Possible types of events

  event_type = 1
  message = "session_data_ready"
  All data from the clearing system have been loaded into the trading system

  event_type = 5
  message = "clearing_started"
  Start of the clearing session

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

  event_type = 17
  message = "clearing_trades_inserted"
  Clearing trades have been generated

  event_type = 18
  message = "extension_of_limits_started_for_core"
  Limits have been extended

  event_type = 22
  message = "discrete_auction_matching_completed"
  Order matching completed and opening auction trades have been published

  event_type = 23
  message = "discrete_auction_add_order_started"
  The start of accepting orders in the opening auction

  event_type = 24
  message = "discrete_auction_add_order_finished"
  The finish of accepting orders in the opening auction

  event_type = 36
  message = "start_of_calendar_date"
  Beginning of the calendar day

event_type = 37
message = "settlement_session_started"
Beginning of the settlement session

event_type = 38
message = "settlement_session_finished"
End of the settlement session

event_type = 47
message = "settlpricecalc_finished"
Settlement price has been fixed

4.3. Stream FORTS_DEALS_REPL – anonymous trades (Type=R)

4.3.1. Data scheme

Tables:

- deal - Trades
- multileg_deal - Multileg trades
- heartbeat - Server times table
- sys_events - table of events

4.3.1.1. Table deal: Trades

Table 18. Fields of table 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 day ID
isin_id	i4	Instrument unique ID
id_deal	i8	Deal ID number
xpos	i8	Number of positions in the instrument in the market after the trade
xamount	i8	Volume, number of units of the instrument
public_order_id_buy	i8	The buyer's order ID (for iceberg order – ID of its visible part)
public_order_id_sell	i8	The seller's order ID (for iceberg order – ID of its visible part)
price	d16.5	Price
moment	t	Time when the deal was made
moment_ns	u8	Time when the deal was made, nanoseconds since Unix epoch, UTC
nosystem	i1	Sign of non-system deal
xstatus_buy	i8	Status of the trade from the buyer's side
xstatus_sell	i8	Status of the trade from the seller's side
xstatus2_buy	i8	Extension for trades statuses (in addition to the ‘xstatus_buy’ field)
xstatus2_sell	i8	Extension for trades statuses (in addition to the ‘xstatus_sell’ field)

Notes:

- In exercise trades, field public_order_id_sell contains the request ID (option trade). In exercise trades, field public_order_id_buy contains the request ID (futures trade for option call). In exercise trades, field public_order_id_sell contains the order ID (futures trade for option put).
- Fields xstatus_sell and xstatus_buy are bit masks (for details see Flags applied to orders and trades)

4.3.1.2. Table multileg_deal: Multileg trades

Table 19. Fields of table 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

Field	Type	Description
sess_id	i4	Trading day ID
isin_id	i4	Multileg instrument ID
id_deal	i8	Deal ID number
public_order_id_buy	i8	The buyer's order ID (for iceberg order – ID of its visible part)
public_order_id_sell	i8	The seller's order ID (for iceberg order – ID of its visible part)
xamount	i8	Volume, number of units of the instrument
price	d16.5	Price of the first part of multileg trade
rate_price	d16.5	Rate price
swap_price	d16.5	Swap price
moment	t	Time when the deal was made
moment_ns	u8	Time when the deal was made, nanoseconds since Unix epoch, UTC
nosystem	i1	Sign of non-system deal
xstatus_buy	i8	Status of the trade from the buyer's side
xstatus_sell	i8	Status of the trade from the seller's side
xstatus2_buy	i8	Extension for trades statuses (in addition to the ‘xstatus_buy’ field)
xstatus2_sell	i8	Extension for trades statuses (in addition to the ‘xstatus_sell’ field)

Notes:

- Fields xstatus_sell and xstatus_buy are bit masks (for details see Flags applied to orders and trades)

4.3.1.3. Table heartbeat: Server times table

Records in this table are added periodically by the trading system's core. It can be used for synchronization purposes (e.g. to check whether all the trades were received at specified moment of time). The table is insert-only, no modifications or deletions occur during trading session.

Table 20. 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

4.3.1.4. Table sys_events: table of events

Table 21. 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	Unique ID of the event
sess_id	i4	Trading day ID
event_type	i4	Type of the event
message	c64	Description of the event
server_time	t	Event time

Notes:

- Possible types of events

  event_type = 1
  message = "session_data_ready"
  All data from the clearing system have been loaded into the trading system

  event_type = 5
  message = "clearing_started"
  Start of the clearing session

  event_type = 6

message = "extension_of_limits_finished"
Limits have been extended

event_type = 17
message = "clearing_trades_inserted"
Clearing trades have been generated

event_type = 18
message = "extension_of_limits_started_for_core"
Limits have been extended

event_type = 22
message = "discrete_auction_matching_completed"
Order matching completed and opening auction trades have been published

event_type = 23
message = "discrete_auction_add_order_started"
The start of accepting orders in the opening auction

event_type = 24
message = "discrete_auction_add_order_finished"
The finish of accepting orders in the opening auction

event_type = 36
message = "start_of_calendar_date"
Beginning of the calendar day

event_type = 37
message = "settlement_session_started"
Beginning of the settlement session

event_type = 38
message = "settlement_session_finished"
End of the settlement session

event_type = 47
message = "settlpricecalc_finished"
Settlement price has been fixed

4.4. Stream FORTS_FEE_REPL - exchange fees and penalties (Type=AR)

4.4.1. Data scheme

Tables:

- adjusted_fee - exchange fees
- penalty - penalties
- FEE_info_currentday - snapshot information
- sys_events - table of events

4.4.1.1. Table adjusted_fee: exchange fees

Table 22. Fields of table adjusted_fee

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
id_deal	i8	Deal ID number
moment	t	Time when the deal was made
moment_ns	u8	Time when the deal was made, nanoseconds since Unix epoch, UTC
code_buy	c7	Buyer's code
code_sell	c7	Seller's code
initial_fee_buy	d26.2	Initial fee of the buyer's deal
initial_fee_sell	d26.2	Initial fee of the seller's deal
adjusted_fee_buy	d26.2	Adjusted fee of the buyer's deal

Field	Type	Description
adjusted_fee_trade_buy	d26.2	Adjusted exchange fee of the buyer's deal
adjusted_fee_clearing_buy	d26.2	Adjusted clearing fee of the buyer's deal
adjusted_fee_sell	d26.2	Adjusted fee of the seller's deal
adjusted_fee_trade_sell	d26.2	Adjusted exchange fee of the seller's deal
adjusted_fee_clearing_sell	d26.2	Adjusted clearing fee of the seller's deal
id_deal_multileg	i8	Deal ID number for multileg deals

4.4.1.2. Table penalty: penalties

Table 23. Fields of table penalty

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 day ID
id_deal	i8	Deal ID number
id_deal_multileg	i8	Deal ID number for multileg deals
moment	t	Time when the deal was made
moment_ns	u8	Time when the deal was made, nanoseconds since Unix epoch, UTC
code_buy	c7	Buyer's code
code_sell	c7	Seller's code
penalty_buy	d26.2	Penalty of the buyer's deal
penalty_sell	d26.2	Penalty of the seller's deal

4.4.1.3. Table FEE_info_currentday: Snapshot information

Service table used to obtain a snapshot of fees for the current day.

Table 24. Fields of table FEE_info_currentday

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
execlog_rev	i8	Revision of the 'start_of_calendar_date' event from the 'exec_log' stream
lifenum	i8	Life number of the 'FORTS_BROKER_FEE' stream

Notes:

- To obtain a snapshot of fees for the current day:
  - Retrieve only the FEE_info_currentday table and read <lifenum> and <execlog_rev> from it.
  - Then call the listener for the FORTS_BROKER_FEE stream: CG_RESULT cg_lsn_open(cg_listener_t* lsn, const char* settings),

    passing “mode=snapshot+online,lifenum=< lifenum >,rev.adjusted_fee=<execlog_rev >” in the settings string.

    The values of <lifenum> and <execlog_rev> are read from the FEE_info_currentday table.
  - This retrieves fees for the current day only.

4.4.1.4. Table sys_events: table of events

Table 25. 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

Field	Type	Description
event_id	i8	Unique ID of the event
sess_id	i4	Trading day ID
event_type	i4	Type of the event
message	c64	Description of the event
server_time	t	Event time

Notes:

- Possible types of events

  event_type = 1
  message = "session_data_ready"
  All data from the clearing system have been loaded into the trading system

  event_type = 5
  message = "clearing_started"
  Start of the clearing session

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

  event_type = 17
  message = "clearing_trades_inserted"
  Clearing trades have been generated

  event_type = 18
  message = "extension_of_limits_started_for_core"
  Limits have been extended

  event_type = 36
  message = "start_of_calendar_date"
  Beginning of the calendar day

  event_type = 37
  message = "settlement_session_started"
  Beginning of the settlement session

  event_type = 38
  message = "settlement_session_finished"
  End of the settlement session

  event_type = 47
  message = "settlpricecalc_finished"
  Settlement price has been fixed

4.5. Stream FORTS_FEERATE_REPL - Precise Exchange fee rates (Type=AR)

4.5.1. Data scheme

Tables:

- futures_rate - fee rates on futures and multi-leg instruments
- option_rate - fee rates on option contracts
- sys_events - table of events

4.5.1.1. Table futures_rate: fee rates on futures and multi-leg instruments

Table 26. Fields of table futures_rate

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
isin_id	i4	Instrument unique ID
sess_id	i4	Trading day ID

Field	Type	Description
exchange_fee_negdeal	d26.2	Precise exchange fee rate for negotiated trade
exchange_fee	d26.2	Precise exchange fee rate for anonymous trade
clearing_fee_negdeal	d26.2	Precise clearing fee rate for negotiated trade
clearing_fee	d26.2	Precise clearing fee rate for anonymous trade
exp_clearing_fee	d26.2	Precise clearing fee rate for contract exercising.

4.5.1.2. Table option_rate: fee rates on option contracts

Table 27. Fields of table option_rate

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
isin_id	i4	Instrument unique ID
sess_id	i4	Trading day ID
exchange_fee_negdeal	d26.2	Precise exchange fee rate for negotiated trade
exchange_fee	d26.2	Precise exchange fee rate for anonymous trade
clearing_fee_negdeal	d26.2	Precise clearing fee rate for negotiated trade
clearing_fee	d26.2	Precise clearing fee rate for anonymous trade
exp_clearing_fee	d26.2	Precise clearing fee rate for contract exercising.

4.5.1.3. Table sys_events: table of events

Table 28. 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	Unique ID of the event
sess_id	i4	Trading day ID
event_type	i4	Type of the event
message	c64	Description of the event
server_time	t	Event time

Notes:

- Possible types of events

  event_type = 1
  message = "session_data_ready"
  All data from the clearing system have been loaded into the trading system

  event_type = 5
  message = "clearing_started"
  Start of the clearing session

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

  event_type = 17
  message = "clearing_trades_inserted"
  Clearing trades have been generated

  event_type = 18
  message = "extension_of_limits_started_for_core"
  Limits have been extended

  event_type = 36

message = "start_of_calendar_date"
Beginning of the calendar day

event_type = 37
message = "settlement_session_started"
Beginning of the settlement session

event_type = 38
message = "settlement_session_finished"
End of the settlement session

event_type = 47
message = "settlpricecalc_finished"
Settlement price has been fixed

4.6. Stream FORTS_BROKER_FEE_REPL - Brokerage fees (Type=I)

4.6.1. Data scheme

Tables:

- broker_fee - brokerage fee
- BFEE_info_currentday - snapshot information
- sys_events - table of events

4.6.1.1. Table broker_fee: brokerage fee

Table 29. Fields of table broker_fee

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 day ID
id_deal	i8	Deal ID number
moment	t	Time when the deal was made
moment_ns	u8	Time when the deal was made, nanoseconds since Unix epoch, UTC
code_buy	c7	Buyer's code
code_sell	c7	Seller's code
broker_fee_buy	d26.2	Brokerage fee of the buyer's deal
broker_fee_sell	d26.2	Brokerage fee of the seller's deal
id_deal_multileg	i8	Deal ID number for multileg deals

4.6.1.2. Table BFEE_info_currentday: Snapshot information

Service table used to obtain a snapshot of fees for the current day.

Table 30. Fields of table BFEE_info_currentday

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
execlog_rev	i8	Revision of the 'start_of_calendar_date' event from the 'exec_log' stream
lifenum	i8	Life number of the 'FORTS_BROKER_FEE_REPL' stream

Notes:

- To obtain a snapshot of fees for the current day:
  - Retrieve only the BFEE_info_currentday table and read <lifenum> and <execlog_rev> from it.
  - Then call the listener for the FORTS_BROKER_FEE_REPL stream: CG_RESULT cg_lsn_open(cg_listener_t* lsn, const char* settings),

passing “mode=snapshot+online,lifenum=< lifenum >,rev.adjusted_fee=<execlog_rev >” in the settings string.

The values of <lifenum> and <execlog_rev> are read from the BFEE_info_currentday table.

- This retrieves fees for the current day only.

4.6.1.3. Table sys_events: table of events

Table 31. 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	Unique ID of the event
sess_id	i4	Trading day ID
event_type	i4	Type of the event
message	c64	Description of the event
server_time	t	Event time

Notes:

- Possible types of events

  event_type = 1
  message = "session_data_ready"
  All data from the clearing system have been loaded into the trading system

  event_type = 5
  message = "clearing_started"
  Start of the clearing session

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

  event_type = 17
  message = "clearing_trades_inserted"
  Clearing trades have been generated

  event_type = 18
  message = "extension_of_limits_started_for_core"
  Limits have been extended

  event_type = 36
  message = "start_of_calendar_date"
  Beginning of the calendar day

  event_type = 37
  message = "settlement_session_started"
  Beginning of the settlement session

  event_type = 38
  message = "settlement_session_finished"
  End of the settlement session

  event_type = 47
  message = "settlpricecalc_finished"
  Settlement price has been fixed

4.7. Stream FORTS_BROKER_FEE_PARAMS_REPL - Parameters for calculating the brokerage fee (Type=I)

4.7.1. Data scheme

Tables:

- broker_fee_params - Parameters for calculating the brokerage fee
- sys_events - table of events

4.7.1.1. Table broker_fee_params: Parameters for calculating the brokerage fee

Table 32. Fields of table 'broker_fee_params'

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 day ID
client_code	c7	Client code (broker code)
lower_fee	d26.2	Minimum possible amount of brokerage fee per contract
upper_fee	d26.2	Maximum possible amount of brokerage fee per contract
multiplier	d26.2	Multiplier to the amount of exchange and clearing fees
additive	d26.2	Constant addition per contract

Notes:

- The 'client_code' field may contain the client code or BF code. If a client code is specified, then the specified parameters are used to calculate the brokerage fee for the trades of this client. If a broker code is specified, then the parameters are used to calculate the brokerage fee for all BF clients.
- Field 'sess_id' can take the following values:

  sess_id Current calculation parameters.

  -1 Adding new calculation parameters. Parameters will be applied in the next trading day.

  -2 Deletion of current calculation parameters. Parameters will be deleted in the next trading day.

4.7.1.2. Table sys_events: table of events

Table 33. 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	Unique ID of the event
sess_id	i4	Trading day ID
event_type	i4	Type of the event
message	c64	Description of the event
server_time	t	Event time

Notes:

- Possible types of events

  event_type = 1
  message = "session_data_ready"
  All data from the clearing system have been loaded into the trading system

  event_type = 5
  message = "clearing_started"
  Start of the clearing session

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

  event_type = 17
  message = "clearing_trades_inserted"
  Clearing trades have been generated

  event_type = 18
  message = "extension_of_limits_started_for_core"
  Limits have been extended

  event_type = 36

message = "start_of_calendar_date"
Beginning of the calendar day

event_type = 37
message = "settlement_session_started"
Beginning of the settlement session

event_type = 38
message = "settlement_session_finished"
End of the settlement session

event_type = 47
message = "settlpricecalc_finished"
Settlement price has been fixed

4.8. Stream FORTS_USERORDERBOOK_REPL - User orders: order-book snapshot (Type=R)

The following data is published in the stream every 2 minutes: snapshot of active orders and a record in the 'info' table with the revision of the last processed transaction from 'orders_log', the stream's life number and the publication state of the snapshot ('publication_state' field). The 'publication_state' field is set to '0' in snapshot publication moment. After the snapshot is published, 'publication_state' field is set to '1'. The data in the 'orders' table may be inconsistent until 'publication_state' = 1.

4.8.1. Data scheme

Tables:

- orders - Current futures and options order-book
- multileg_orders - Table of active multileg orders
- info - Order-book snapshots information
- orders_currentday - Snapshot of active user orders at the start of the current day
- multileg_orders_currentday - Snapshot of active user multileg orders at the start of the current day
- info_currentday - Snapshot information

4.8.1.1. Table orders: Current futures and options order-book

Table 34. Fields of table orders

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
public_order_id	i8	Order ID (for iceberg order – ID of its visible part)
sess_id	i4	Trading day ID
client_code	c7	Client code
moment	t	Order status changing time
moment_ns	u8	Order status changing time, nanoseconds since Unix epoch, UTC
xstatus	i8	Extended order's status
xstatus2	i8	Extension for orders statuses (in addition to the ‘xstatus’ field)
public_action	i1	Type of operation with the order (for iceberg order – type of operation with its visible part)
isin_id	i4	Instrument unique ID
dir	i1	Direction
price	d16.5	Price
public_amount	i8	The number of contracts in the operation (for iceberg order - the number of contracts in the operation for its visible part)
public_amount_rest	i8	The remaining number of contracts in the order (for iceberg order – the remaining number of contracts in its visible part)
comment	c20	Trader's comment
ext_id	i4	External ID number. It is added to orders, trades

Field	Type	Description
login_from	c20	Login of the user who has entered the order
broker_to	c7	SPECTRA code of the company to whom the negotiated order is addressed
broker_to_rts	c7	RTS code of the company to whom the negotiated order is addressed
date_exp	t	Order's expiration date
id_ord1	i8	ID number of the first order
broker_from_rts	c7	RTS code of the company who has entered the order
aspref	i4	Client ID. For orders added by SMA login - MASTER login ID.
private_order_id	i8	Order ID (for iceberg order – ID of the entire order)
private_amount	i8	The number of contracts in the operation (for iceberg order - the number of contracts in the operation for the entire order)
private_amount_rest	i8	The remaining number of contracts in the order (for iceberg order – the remaining number of contracts in the entire order)
variance_amount	i8	Variance amplitude for a random addition for the pop-up part (in contracts)
disclose_const_amount	i8	Number of instrument units in the pop-up part of the iceberg order
private_action	i1	Type of operation with the order (for iceberg order – type of operation with the entire order)
private_init_moment	t	Placement order time (for iceberg order – placement time of the entire order)
private_init_amount	i8	The initial number of contracts in the order (for iceberg order - the initial number of contracts in the entire order)
reason	i4	The flag (reason) of the order submitted for the making of the settlement trade of obligations.
match_ref	c10	Identical text values entered by both trading parties to match negotiated orders
public_init_moment	t	Placement order time (for iceberg order – placement time of its visible part)
public_init_amount	i8	The initial number of contracts in the order (for iceberg order - the initial number of contracts in its visible part)
compliance_id	c1	Order adding method
prevorder_id	i8	Order ID modified by the MoveOrder operation. Filled only in new orders generated upon successful execution of MoveOrder.

Notes:

- Field xstatus is a bit mask. For the complete list of all possible values of field 'status' please refer to section Flags applied to orders and trades.
- Field dir can take the following values:

  1 Buy

  2 Sell
- Field public_action can take the following values:

  0 Order cancelled

  1 Order added

  2 Order is exercised in the trade
- Field 'private_action' ('action') can take the following values:

  0 Order cancelled

  1 Order added

  2 Order is exercised in the trade

  3 The order was added by appearance of a new visible part of the iceberg
- Field 'reason' can take the following values:

  0 Regular order

  4 Balancing Derivatives contracts entered into with the Non-defaulting Clearing Member without submitting orders

6 Closing Derivatives contracts entered into under the cross-default procedure

7 Closing Derivatives Contracts entered into upon non-fulfillment of the Margin Call

8 Closing Derivatives contracts entered into in into upon non-fulfillment of the Delivery Obligation on the deliverable Derivatives contracts for precious metals.

100 Other

- Field 'compliance_id' can take the following values:

" " (space or empty string)	Not filled in/Not specified
M	Manual input
S	As a result of the conditional request (stop-loss order)
R	As a result of the robot algorithm work
A	As a result of the auto-following algorithm
D	Covering a position as a result of an unexecuted Margin Call

4.8.1.2. Table multileg_orders: Table of active multileg orders

Table 35. Fields of table multileg_orders

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
public_order_id	i8	Order ID (for iceberg order – ID of its visible part)
sess_id	i4	Trading day ID
client_code	c7	Client code
moment	t	Order status changing time
moment_ns	u8	Order status changing time, nanoseconds since Unix epoch, UTC
xstatus	i8	Extended order's status
xstatus2	i8	Extension for orders statuses (in addition to the ‘xstatus’ field)
public_action	i1	Type of operation with the order (for iceberg order – type of operation with its visible part)
isin_id	i4	Multileg instrument ID
dir	i1	Direction
price	d16.5	Price. *The field is not used.*
public_amount	i8	The number of contracts in the operation (for iceberg order - the number of contracts in the operation for its visible part)
public_amount_rest	i8	The remaining number of contracts in the order (for iceberg order – the remaining number of contracts in its visible part)
comment	c20	Trader's comment
ext_id	i4	External ID number. It is added to orders, trades
login_from	c20	Login of the user who has entered the order
broker_to	c7	SPECTRA code of the company to whom the negotiated order is addressed
broker_to_rts	c7	RTS code of the company to whom the negotiated order is addressed
date_exp	t	Order's expiration date
id_ord1	i8	ID number of the first order
broker_from_rts	c7	RTS code of the company who has entered the order
aspref	i4	Client ID. For orders added by SMA login - MASTER login ID.
private_order_id	i8	Order ID (for iceberg order – ID of the entire order)
private_amount	i8	The number of contracts in the operation (for iceberg order - the number of contracts in the operation for the entire order)
private_amount_rest	i8	The remaining number of contracts in the order (for iceberg order – the remaining number of contracts in the entire order)

Field	Type	Description
variance_amount	i8	Variance amplitude for a random addition for the pop-up part (in contracts)
disclose_const_amount	i8	Number of instrument units in the pop-up part of the iceberg order
private_action	i1	Type of operation with the order (for iceberg order – type of operation with the entire order)
private_init_moment	t	Placement order time (for iceberg order – placement time of the entire order)
private_init_amount	i8	The initial number of contracts in the order (for iceberg order - the initial number of contracts in the entire order)
reason	i4	The flag (reason) of the order submitted for the making of the settlement trade of obligations.
match_ref	c10	Identical text values entered by both trading parties to match negotiated orders
public_init_moment	t	Placement order time (for iceberg order – placement time of its visible part)
public_init_amount	i8	The initial number of contracts in the order (for iceberg order - the initial number of contracts in its visible part)
compliance_id	c1	Order adding method
rate_price	d16.5	Rate price. *The field is not used.*
swap_price	d16.5	Swap price

Notes:

- Field xstatus is a bit mask. For the complete list of all possible values of field 'status' please refer to section Flags applied to orders and trades.
- Field dir can take the following values:

  1 Buy

  2 Sell
- Field public_action can take the following values

  0 Order cancelled

  1 Order added

  2 Order exercised in a trade
- Field 'private_action' ('action') can take the following values:

  0 Order cancelled

  1 Order added

  2 Order is exercised in the trade

  3 The order was added by appearance of a new visible part of the iceberg
- Field 'reason' can take the following values:

  0 Regular order

  4 Balancing Derivatives contracts entered into with the Non-defaulting Clearing Member without submitting orders

  6 Closing Derivatives contracts entered into under the cross-default procedure

  7 Closing Derivatives Contracts entered into upon non-fulfillment of the Margin Call

  8 Closing Derivatives contracts entered into in into upon non-fulfillment of the Delivery Obligation on the deliverable Derivatives contracts for precious metals.

  100 Other
- Field 'compliance_id' can take the following values:

  " " (space or empty string) Not filled in/Not specified

  M Manual input

  S As a result of the conditional request (stop-loss order)

R	As a result of the robot algorithm work
A	As a result of the auto-following algorithm
D	Covering a position as a result of an unexecuted Margin Call

4.8.1.3. Table info: Order-book snapshots information

Table 36. Fields of table info

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
infoID	i8	Unique key
moment	t	Snapshot time
publication_state	i1	State of the snapshot publication
trades_rev	i8	Last processed revision at the time of snapshot creation
trades_lifenum	i8	Life number of the stream

Notes:

- Field publication_state can take the following values:

  0 in progress

  1 done

4.8.1.4. Table orders_currentday: Snapshot of active user orders at the start of the current day

Table 37. Fields of table orders_currentday

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
public_order_id	i8	Order ID (for iceberg order – ID of its visible part)
sess_id	i4	Trading day ID
client_code	c7	Client code
moment	t	Order status changing time
moment_ns	u8	Order status changing time, nanoseconds since Unix epoch, UTC
xstatus	i8	Extended order's status
xstatus2	i8	Extension for orders statuses (in addition to the ‘xstatus’ field)
public_action	i1	Type of operation with the order (for iceberg order – type of operation with its visible part)
isin_id	i4	Instrument unique ID
dir	i1	Direction
price	d16.5	Price
public_amount	i8	The number of contracts in the operation (for iceberg order - the number of contracts in the operation for its visible part)
public_amount_rest	i8	The remaining number of contracts in the order (for iceberg order – the remaining number of contracts in its visible part)
comment	c20	Trader's comment
ext_id	i4	External ID number. It is added to orders, trades
login_from	c20	Login of the user who has entered the order
broker_to	c7	SPECTRA code of the company to whom the negotiated order is addressed
broker_to_rts	c7	RTS code of the company to whom the negotiated order is addressed
date_exp	t	Order's expiration date
id_ord1	i8	ID number of the first order

Field	Type	Description
broker_from_rts	c7	RTS code of the company who has entered the order
aspref	i4	Client ID. For orders added by SMA login - MASTER login ID.
private_order_id	i8	Order ID (for iceberg order – ID of the entire order)
private_amount	i8	The number of contracts in the operation (for iceberg order - the number of contracts in the operation for the entire order)
private_amount_rest	i8	The remaining number of contracts in the order (for iceberg order – the remaining number of contracts in the entire order)
variance_amount	i8	Variance amplitude for a random addition for the pop-up part (in contracts)
disclose_const_amount	i8	Number of instrument units in the pop-up part of the iceberg order
private_action	i1	Type of operation with the order (for iceberg order – type of operation with the entire order)
private_init_moment	t	Placement order time (for iceberg order – placement time of the entire order)
private_init_amount	i8	The initial number of contracts in the order (for iceberg order - the initial number of contracts in the entire order)
reason	i4	The flag (reason) of the order submitted for the making of the settlement trade of obligations.
match_ref	c10	Identical text values entered by both trading parties to match negotiated orders
public_init_moment	t	Placement order time (for iceberg order – placement time of its visible part)
public_init_amount	i8	The initial number of contracts in the order (for iceberg order - the initial number of contracts in its visible part)
compliance_id	c1	Order adding method
prevorder_id	i8	Order ID modified by the MoveOrder operation. Filled only in new orders generated upon successful execution of MoveOrder.

Notes:

- Field xstatus is a bit mask. For the complete list of all possible values of field 'status' please refer to section Flags applied to orders and trades.
- Field dir can take the following values:

  1 Buy

  2 Sell
- Field public_action can take the following values:

  0 Order cancelled

  1 Order added

  2 Order is exercised in the trade
- Field 'private_action' ('action') can take the following values:

  0 Order cancelled

  1 Order added

  2 Order is exercised in the trade

  3 The order was added by appearance of a new visible part of the iceberg
- Field 'reason' can take the following values:

  0 Regular order

  4 Balancing Derivatives contracts entered into with the Non-defaulting Clearing Member without submitting orders

  6 Closing Derivatives contracts entered into under the cross-default procedure

  7 Closing Derivatives Contracts entered into upon non-fulfillment of the Margin Call

  8 Closing Derivatives contracts entered into in into upon non-fulfillment of the Delivery Obligation on the deliverable Derivatives contracts for precious metals.

100 Other

- Field 'compliance_id' can take the following values:

" " (space or empty string)	Not filled in/Not specified
M	Manual input
S	As a result of the conditional request (stop-loss order)
R	As a result of the robot algorithm work
A	As a result of the auto-following algorithm
D	Covering a position as a result of an unexecuted Margin Call

4.8.1.5. Table multileg_orders_currentday: Snapshot of active user multileg orders at the start of the current day

Table 38. Fields of table multileg_orders_currentday

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
public_order_id	i8	Order ID (for iceberg order – ID of its visible part)
sess_id	i4	Trading day ID
client_code	c7	Client code
moment	t	Order status changing time
moment_ns	u8	Order status changing time, nanoseconds since Unix epoch, UTC
xstatus	i8	Extended order's status
xstatus2	i8	Extension for orders statuses (in addition to the ‘xstatus’ field)
public_action	i1	Type of operation with the order (for iceberg order – type of operation with its visible part)
isin_id	i4	Multileg instrument ID
dir	i1	Direction
price	d16.5	Price. *The field is not used.*
public_amount	i8	The number of contracts in the operation (for iceberg order - the number of contracts in the operation for its visible part)
public_amount_rest	i8	The remaining number of contracts in the order (for iceberg order – the remaining number of contracts in its visible part)
comment	c20	Trader's comment
ext_id	i4	External ID number. It is added to orders, trades
login_from	c20	Login of the user who has entered the order
broker_to	c7	SPECTRA code of the company to whom the negotiated order is addressed
broker_to_rts	c7	RTS code of the company to whom the negotiated order is addressed
date_exp	t	Order's expiration date
id_ord1	i8	ID number of the first order
broker_from_rts	c7	RTS code of the company who has entered the order
aspref	i4	Client ID. For orders added by SMA login - MASTER login ID.
private_order_id	i8	Order ID (for iceberg order – ID of the entire order)
private_amount	i8	The number of contracts in the operation (for iceberg order - the number of contracts in the operation for the entire order)
private_amount_rest	i8	The remaining number of contracts in the order (for iceberg order – the remaining number of contracts in the entire order)
variance_amount	i8	Variance amplitude for a random addition for the pop-up part (in contracts)
disclose_const_amount	i8	Number of instrument units in the pop-up part of the iceberg order
private_action	i1	Type of operation with the order (for iceberg order – type of operation with the entire order)

Field	Type	Description
private_init_moment	t	Placement order time (for iceberg order – placement time of the entire order)
private_init_amount	i8	The initial number of contracts in the order (for iceberg order - the initial number of contracts in the entire order)
reason	i4	The flag (reason) of the order submitted for the making of the settlement trade of obligations.
match_ref	c10	Identical text values entered by both trading parties to match negotiated orders
public_init_moment	t	Placement order time (for iceberg order – placement time of its visible part)
public_init_amount	i8	The initial number of contracts in the order (for iceberg order - the initial number of contracts in its visible part)
compliance_id	c1	Order adding method
rate_price	d16.5	Rate price. *The field is not used.*
swap_price	d16.5	Swap price

Notes:

- Field xstatus is a bit mask. For the complete list of all possible values of field 'status' please refer to section Flags applied to orders and trades.
- Field dir can take the following values:

  1 Buy

  2 Sell
- Field public_action can take the following values

  0 Order cancelled

  1 Order added

  2 Order exercised in a trade
- Field 'private_action' ('action') can take the following values:

  0 Order cancelled

  1 Order added

  2 Order is exercised in the trade

  3 The order was added by appearance of a new visible part of the iceberg
- Field 'reason' can take the following values:

  0 Regular order

  4 Balancing Derivatives contracts entered into with the Non-defaulting Clearing Member without submitting orders

  6 Closing Derivatives contracts entered into under the cross-default procedure

  7 Closing Derivatives Contracts entered into upon non-fulfillment of the Margin Call

  8 Closing Derivatives contracts entered into in into upon non-fulfillment of the Delivery Obligation on the deliverable Derivatives contracts for precious metals.

  100 Other
- Field 'compliance_id' can take the following values:

  " " (space or empty string) Not filled in/Not specified

  M Manual input

  S As a result of the conditional request (stop-loss order)

  R As a result of the robot algorithm work

  A As a result of the auto-following algorithm

  D Covering a position as a result of an unexecuted Margin Call

4.8.1.6. Table info_currentday: Snapshot information

Table 39. Fields of table info_currentday

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
publication_state	i1	State of the snapshot publication
trades_rev	i8	Revision of the start_of_calendar_date event
trades_lifenum	i8	Life number of the stream
server_time	t	Date and time of order snapshot creation

Notes:

- Field publication_state can take the following values:

  0 in progress

  1 done

4.9. Stream FORTS_ORDBOOK_REPL - Depersonalized order-book snapshot (Type=R)

The following data is published in the stream every 2 minutes: snapshot of active orders and a record in the 'info' table with the revision of the last processed transaction from 'orders_log', the stream's life number and the publication state of the snapshot ('publication_state' field). The 'publication_state' field is set to '0' in snapshot publication moment. After the snapshot is published, 'publication_state' field is set to '1'. The data in the 'orders' table may be inconsistent until 'publication_state' = 1.

4.9.1. Data scheme

Tables:

- orders - Current order-book
- multileg_orders - Table of active multileg orders
- info - Order-book snapshots information
- orders_currentday - Snapshot of active orders at the start of the current day
- multileg_orders_currentday - Snapshot of active multileg orders at the start of the current day
- info_currentday - Order-book snapshots information

4.9.1.1. Table orders: Current order-book

Table 40. Fields of table orders

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
public_order_id	i8	Order ID (for iceberg order – ID of its visible part)
sess_id	i4	Trading day ID
moment	t	Order status changing time
moment_ns	u8	Order status changing time, nanoseconds since Unix epoch, UTC
xstatus	i8	Extended order's status
xstatus2	i8	Extension for orders statuses (in addition to the ‘xstatus’ field)
public_action	i1	Type of operation with the order (for iceberg order – type of operation with its visible part)
isin_id	i4	Instrument unique ID
dir	i1	Direction
price	d16.5	Price
public_amount	i8	The number of contracts in the operation (for iceberg order - the number of contracts in the operation for its visible part)

Field	Type	Description
public_amount_rest	i8	The remaining number of contracts in the order (for iceberg order – the remaining number of contracts in its visible part)
public_init_moment	t	Placement order time (for iceberg order – placement time of its visible part)
public_init_amount	i8	The initial number of contracts in the order (for iceberg order - the initial number of contracts in its visible part)

Notes:

- Field xstatus is a bit mask. For the complete list of all possible values of field 'status' please refer to section Flags applied to orders and trades.
- Field dir can take the following values:

  1 Buy

  2 Sell

- Field public_action can take the following values

  1 Order added

  2 Order is exercised in the trade

4.9.1.2. Table multileg_orders: Table of active multileg orders

Table 41. Fields of table multileg_orders

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
public_order_id	i8	Order ID (for iceberg order – ID of its visible part)
sess_id	i4	Trading day ID
moment	t	Order status changing time
moment_ns	u8	Order status changing time, nanoseconds since Unix epoch, UTC
xstatus	i8	Extended order's status
xstatus2	i8	Extension for orders statuses (in addition to the ‘xstatus’ field)
public_action	i1	Type of operation with the order (for iceberg order – type of operation with its visible part)
isin_id	i4	Multileg instrument ID
dir	i1	Direction
price	d16.5	Price. *The field is not used.*
public_amount	i8	The number of contracts in the operation (for iceberg order - the number of contracts in the operation for its visible part)
public_amount_rest	i8	The remaining number of contracts in the order (for iceberg order – the remaining number of contracts in its visible part)
public_init_moment	t	Placement order time (for iceberg order – placement time of its visible part)
public_init_amount	i8	The initial number of contracts in the order (for iceberg order - the initial number of contracts in its visible part)
rate_price	d16.5	Rate price. *The field is not used.*
swap_price	d16.5	Swap price

Notes:

- Field xstatus is a bit mask. For the complete list of all possible values of field 'status' please refer to section Flags applied to orders and trades.
- Field dir can take the following values:

  1 Buy

  2 Sell

- Field public_action can take the following values

0 Order cancelled

1 Order added

2 Order exercised in a trade

- Field 'private_action' ('action') can take the following values:

0 Order cancelled

1 Order added

2 Order is exercised in the trade

3 The order was added by appearance of a new visible part of the iceberg

- Field 'reason' can take the following values:

0 Regular order

4 Balancing Derivatives contracts entered into with the Non-defaulting Clearing Member without submitting orders

6 Closing Derivatives contracts entered into under the cross-default procedure

7 Closing Derivatives Contracts entered into upon non-fulfillment of the Margin Call

8 Closing Derivatives contracts entered into in into upon non-fulfillment of the Delivery Obligation on the deliverable Derivatives contracts for precious metals.

100 Other

- Field 'compliance_id' can take the following values:

" " (space or empty string) Not filled in/Not specified

M Manual input

S As a result of the conditional request (stop-loss order)

R As a result of the robot algorithm work

A As a result of the auto-following algorithm

D Covering a position as a result of an unexecuted Margin Call

4.9.1.3. Table info: Order-book snapshots information

Table 42. Fields of table info

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
infoID	i8	Unique key
moment	t	Snapshot time
publication_state	i1	State of the snapshot publication
trades_rev	i8	Last processed revision at the time of snapshot creation
trades_lifenum	i8	Life number of the stream

Notes:

- Field publication_state can take the following values:

0 in progress

1 done

4.9.1.4. Table orders_currentday: Snapshot of active orders at the start of the current day

Table 43. Fields of table orders_currentday

Field	Type	Description
replID	i8	Service field of the replication subsystem

Field	Type	Description
replRev	i8	Service field of the replication subsystem
replAct	i8	Service field of the replication subsystem
public_order_id	i8	Order ID (for iceberg order – ID of its visible part)
sess_id	i4	Trading day ID
moment	t	Order status changing time
moment_ns	u8	Order status changing time, nanoseconds since Unix epoch, UTC
xstatus	i8	Extended order's status
xstatus2	i8	Extension for orders statuses (in addition to the ‘xstatus’ field)
public_action	i1	Type of operation with the order (for iceberg order – type of operation with its visible part)
isin_id	i4	Instrument unique ID
dir	i1	Direction
price	d16.5	Price
public_amount	i8	The number of contracts in the operation (for iceberg order - the number of contracts in the operation for its visible part)
public_amount_rest	i8	The remaining number of contracts in the order (for iceberg order – the remaining number of contracts in its visible part)
public_init_moment	t	Placement order time (for iceberg order – placement time of its visible part)
public_init_amount	i8	The initial number of contracts in the order (for iceberg order - the initial number of contracts in its visible part)

Notes:

- Field xstatus is a bit mask. For the complete list of all possible values of field 'status' please refer to section Flags applied to orders and trades.
- Field dir can take the following values:

  1 Buy

  2 Sell
- Field public_action can take the following values

  1 Order added

  2 Order is exercised in the trade

4.9.1.5. Table multileg_orders_currentday: Snapshot of active multileg orders at the start of the current day

Table 44. Fields of table multileg_orders_currentday

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
public_order_id	i8	Order ID (for iceberg order – ID of its visible part)
sess_id	i4	Trading day ID
moment	t	Order status changing time
moment_ns	u8	Order status changing time, nanoseconds since Unix epoch, UTC
xstatus	i8	Extended order's status
xstatus2	i8	Extension for orders statuses (in addition to the ‘xstatus’ field)
public_action	i1	Type of operation with the order (for iceberg order – type of operation with its visible part)
isin_id	i4	Multileg instrument ID
dir	i1	Direction
price	d16.5	Price. *The field is not used.*
public_amount	i8	The number of contracts in the operation (for iceberg order - the number of contracts in the operation for its visible part)

Field	Type	Description
public_amount_rest	i8	The remaining number of contracts in the order (for iceberg order – the remaining number of contracts in its visible part)
public_init_moment	t	Placement order time (for iceberg order – placement time of its visible part)
public_init_amount	i8	The initial number of contracts in the order (for iceberg order - the initial number of contracts in its visible part)
rate_price	d16.5	Rate price. *The field is not used.*
swap_price	d16.5	Swap price

Notes:

- Field xstatus is a bit mask. For the complete list of all possible values of field 'status' please refer to section Flags applied to orders and trades.
- Field dir can take the following values:

  1 Buy

  2 Sell
- Field public_action can take the following values

  0 Order cancelled

  1 Order added

  2 Order exercised in a trade
- Field 'private_action' ('action') can take the following values:

  0 Order cancelled

  1 Order added

  2 Order is exercised in the trade

  3 The order was added by appearance of a new visible part of the iceberg
- Field 'reason' can take the following values:

  0 Regular order

  4 Balancing Derivatives contracts entered into with the Non-defaulting Clearing Member without submitting orders

  6 Closing Derivatives contracts entered into under the cross-default procedure

  7 Closing Derivatives Contracts entered into upon non-fulfillment of the Margin Call

  8 Closing Derivatives contracts entered into in into upon non-fulfillment of the Delivery Obligation on the deliverable Derivatives contracts for precious metals.

  100 Other
- Field 'compliance_id' can take the following values:

  " " (space or empty string) Not filled in/Not specified

  M Manual input

  S As a result of the conditional request (stop-loss order)

  R As a result of the robot algorithm work

  A As a result of the auto-following algorithm

  D Covering a position as a result of an unexecuted Margin Call

4.9.1.6. Table info_currentday: Snapshot information

Table 45. Fields of table info_currentday

Field	Type	Description
replID	i8	Service field of the replication subsystem
replRev	i8	Service field of the replication subsystem

Field	Type	Description
replAct	i8	Service field of the replication subsystem
publication_state	i1	State of the snapshot publication
trades_rev	i8	Revision of the start_of_calendar_date event
trades_lifenum	i8	Life number of the stream
server_time	t	Date and time of order snapshot creation

Notes:

- Field publication_state can take the following values:

  0 in progress

  1 done

4.10. Stream FORTS_COMMON_REPL - Market fundamentals (Type=I)

4.10.1. Data scheme

Tables:

- common - Market fundamentals
- sys_events - Table of events

4.10.1.1. Table common: Market fundamentals

The table contains market fundamentals data (best buy/sell orders, opening/closing price values, etc).

Table 46. Fields of table common

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 day ID
isin_id	i4	Instrument unique ID
best_buy	d16.5	Best bid (subject to the synthetic liquidity)
xamount_buy	i8	Size of the best bid (subject to the synthetic liquidity)
orders_buy_qty	i4	Number of bid orders (subject to the synthetic liquidity)
xorders_buy_amount	i8	Total number of contracts in bid (subject to the synthetic liquidity)
best_sell	d16.5	Best offer (subject to the synthetic liquidity)
xamount_sell	i8	Size of the best offer (subject to the synthetic liquidity)
orders_sell_qty	i4	Number of offer orders (subject to the synthetic liquidity)
xorders_sell_amount	i8	Total number of contracts in offer (subject to the synthetic liquidity)
open_price	d16.5	Opening price
close_price	d16.5	Closing price
opening_auction_price	d16.5	Opening auction price
price	d16.5	Price of the last trade
trend	d16.5	Price trend (difference between the prices of the last two trades)
xamount	i8	Size of the last trade
deal_time	t	Date and time of the last trade
deal_time_ns	u8	Date and time of the last trade, nanoseconds since Unix epoch, UTC
min_price	d16.5	The low price
max_price	d16.5	The high price
avr_price	d16.5	Average weighted price
xcontr_count	i8	Total number of contracts in the trades (volume).Takes into account all trades (negotiated and addressless).

Field	Type	Description
capital	d26.2	Total volume of trades in Russian rubles (turnover). Shown without taking into account negotiated trades.
total_premium_volume	d26.2	Total volume of option premium
deal_count	i4	Number of trades
settlement_price	d16.5	Settlement price at the start of the trading day.
xpos	i8	Current open interest
mod_time	t	Date and time of changing the entry in the table
mod_time_ns	u8	Date and time of changing the entry in the table, nanoseconds since Unix epoch, UTC
market_price	d16.5	Current market price.
price_assigned_by_admin	i1	Attribute of manual current market price setting by trades Administrator.
local_time	t	Time stamp for monitoring purposes
best_buy_native	d16.5	Best bid (excluding synthetic liquidity)
xamount_buy_native	i8	Size of the best bid (excluding synthetic liquidity)
xorders_buy_amount_native	i8	Total number of contracts in bid (excluding synthetic liquidity)
best_sell_native	d16.5	Best offer (excluding synthetic liquidity)
xamount_sell_native	i8	Size of the best offer (excluding synthetic liquidity)
xorders_sell_amount_native	i8	Total number of contracts in offer (excluding synthetic liquidity)
swap_rate	d16.5	Current indicative funding by instrument (for perpetual futures)
swap_rate_curr	d16.5	Current indicative funding by instrument (for perpetual futures). In points.
index_div	d18.4	Current dividend amendment for perpetual futures on an index or share

Notes:

- Field open_price contains the price of the first transaction in the current trading day, and if not, then 0.
- Field close_price, closing price, is determined by the price of the last trade of the trading day executed in the double continuous bilateral auction mode. A non-zero closing price is published at the start of the evening M-t-M clearing session. Closing price information is available until the start of the servicing period following the completed M-t-M session.
- Field 'price_assigned_by_admin' can take the following values:

  1 The value of the current market price in the 'market_price' field is set by the Trading Administrator.

  0 The value of the current market price in the 'market_price' field is calculated by the system.

4.10.1.2. Table sys_events: table of events

Table 47. 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	Unique ID of the event
sess_id	i4	Trading day ID
event_type	i4	Type of the event
message	c64	Description of the event
server_time	t	Event time

Notes:

- Possible types of events

  event_type = 1
  message = "session_data_ready"
  All data from the clearing system have been loaded into the trading system

  event_type = 5
  message = "clearing_started"

Start of the clearing session

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

event_type = 17
message = "clearing_trades_inserted"
Clearing trades have been generated

event_type = 18
message = "extension_of_limits_started_for_core"
Limits have been extended

event_type = 36
message = "start_of_calendar_date"
Beginning of the calendar day

event_type = 37
message = "settlement_session_started"
Beginning of the settlement session

event_type = 38
message = "settlement_session_finished"
End of the settlement session

event_type = 47
message = "settlpricecalc_finished"
Settlement price has been fixed

4.11. Aggregated order-book streams (Type=I)

There are several streams of aggregated order-books with different depths.

- FORTS_AGGR50_REPL – with a depth of 50 price levels
- FORTS_AGGR20_REPL – with a depth of 20 price levels
- FORTS_AGGR5_REPL – with a depth of 5 price levels

4.11.1. Data scheme

Tables:

- orders_aggr - Aggregated order-books
- sys_events - Table of events

4.11.1.1. Table orders_aggr: Aggregated order-books

Aggregated order-books are formed by summing up volumes of active orders with the same instrument, price and direction.

Table 48. Fields of table orders_aggr

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
isin_id	i4	Instrument unique ID
price	d16.5	Price level
volume	i8	The volume subject to the synthetic liquidity
moment	t	Moment of the last record update
moment_ns	u8	Moment of the last record update, nanoseconds since Unix epoch, UTC
dir	i1	Direction
synth_volume	i8	The volume of synthetic liquidity

Note:

- Data in the aggregated order-book table orders_aggr are considered valid only after receiving a synchronous event session_data_ready in the sys_events table.

- The order-book for an instrument may contain records with zero values. This means that the number of orders for the instrument (price levels) is not enough to fill the entire fixed depth of the order-book. Such records should be ignored. Records with zeros can be updated with values with a new price level, when new orders for the instrument appearance in the system.
- The records in the order-book for an instrument can be updated (change price/volume/dir). This means that the previous price level "out" the order-book, and the new one "come in" the order-book.
- Zeroing (volume = 0) of an existing record in the order-book means that the price level has "out" the order-book (for example, the only order that formed the price level was deleted), and there are no other hidden price levels (orders) for the instrument in the system.
- The 'moment' (moment_ns) field value in the table is not monotonically increasing. As 'replRev' increases, records with an earlier value of the 'moment' field may appear in the stream of aggregated order-books. This behavior of the system is expected and can occur in different situations, when the previously formed price level was hidden for some reason, but then began to be displayed. The 'moment' field contains the time of the event that led to the formation of the price level (adding, canceling, exercising an order). Examples of similar system behavior:
  - The simplest case: in the streams of the aggregated order-books, a price-limited number of liquidity levels is displayed. For example,in FORTS_AGGR20_REPL the top 20 price levels is shown only. A hidden, but already formed level with a price outside the displayed range may appear if one of the displayed price levels "disappeared" (for example, the only order that formed the visible price level was deleted).
  - A similar, but slightly more complicated situation is associated with indicative synthetic liquidity. In aggregated order-books, regardless of their depth, always no more than 5 price levels formed by indicative synthetic orders are broadcast (see Section 2.9.2, “Synthetic liquidity in aggregated order-books”). If some price levels are represented by indicative synthetic liquidity only, and if such a price level "disappears", the previously hidden price level of indicative synthetic liquidity will be displayed. But if these levels also contain liquidity created by real orders for this instrument, then there will be no such effect. For more information on synthetic matching and indicative synthetic liquidity, see Section 2.9, “Synthetic matching”.

An example of building an aggregated order-book:

To simplify, we choose the case when the depth of the glass is equal to 2.

1. Time 12:00:01. A buy order appeared with a price of 5 and a volume of 10 for a new instrument with isin_id=12345. Four new orders are coming. The order-book was completely filled with zero values.

```
replID=1   replRev=1   isin_id=12345   price=0   volume=0   moment='12:00:01'   dir=1
replID=2   replRev=2   isin_id=12345   price=0   volume=0   moment='12:00:01'   dir=1
replID=3   replRev=3   isin_id=12345   price=0   volume=0   moment='12:00:01'   dir=2
replID=4   replRev=4   isin_id=12345   price=0   volume=0   moment='12:00:01'   dir=2
```

   The record was updated. In one of the records, the price and volume was changed. Note that when a new record added, when null records are searched, the system may choose a random replID, i.e.it can be an update of a record with both replID=1 and replID=2 or some other.

```
replID=1   replRev=5   isin_id=12345   price=5   volume=10   moment='12:00:01'   dir=1
```

   Order-book:

12345		
volume	price	volume
10	5	

2. Time 12:00:02. A buy order appeared with a price of 4 and a volume of 10. The record was updated.

```
replID=2   replRev=6   isin_id=12345   price=4   volume=10   moment='12:00:02'   dir=1
```

   Order-book:

12345		
volume	price	volume
10	5	
10	4	

3. Time 12:00:03. A sell order appeared with a price of 8 and a volume of 10. The record was updated.

```
replID=3   replRev=7   isin_id=12345   price=8   volume=10   moment='12:00:03'   dir=2
```

Order-book:

12345		
volume	price	volume
	8	10
10	5	
10	4	

4. Time 12:00:04. A sell order appeared with a price of 7 and a volume of 10. The record was updated.

```
replID=4   replRev=8   isin_id=12345   price=7   volume=10   moment='12:00:04'   dir=2
```

Order-book:

12345		
volume	price	volume
	8	10
	7	10
10	5	
10	4	

5. Time 12:00:05. A buy order appeared with a price of 4 and a volume of 5. The record was updated. For an order with a price of 4, i.e. with replID=2, volume was changed .

```
replID=2   replRev=9   isin_id=12345   price=4   volume=15   moment='12:00:05'   dir=1
```

Order-book:

12345		
volume	price	volume
	8	10
	7	10
10	5	
15	4	

6. Time 12:00:06. A buy order with a price of 5 and a volume of 10 was deleted/matched. The record was updated. The price and volume values were zero out for replID=1, as there is no such order anymore.

```
replID=1   replRev=10   isin_id=12345   price=0   volume=0   moment='12:00:06'   dir=1
```

Order-book:

12345		
volume	price	volume
	8	10
	7	10
15	4	

7. Time 12:00:07. A buy order appeared with a price of 5 and a volume of 8. The record was updated.

```
replID=1   replRev=11   isin_id=12345   price=5   volume=8   moment='12:00:07'   dir=1
```

Order-book:

12345		
volume	price	volume
	8	10
	7	10
8	5	
15	4	

8. Time 12:00:08. A buy order appeared with a price of 6 and a volume of 10. The record was updated. The entry with price 4 is no longer included in the order-book, it was replaced with price 6.

```
replID=2   replRev=12   isin_id=12345   price=6   volume=10   moment='12:00:08'   dir=1
```

Order-book:

12345		
volume	price	volume
	8	10
	7	10
10	6	
8	5	
15	4	

9. Time 12:00:09. A buy order with a price of 6 and a volume of 10 was deleted/matched. The record was updated. The order with a price of 6 was deleted, so the order with a price of 4 is returned to the order-book.

```
replID=2   replRev=13   isin_id=12345   price=4   volume=15   moment='12:00:09'   dir=1
```

Order-book:

12345		
volume	price	volume
	8	10
	7	10
8	5	
15	4	

4.11.1.2. Table sys_events: table of events

Table 49. 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_type	i4	Type of the event
event_id	i8	Unique ID of the event
sess_id	i4	Trading day ID
message	c64	Description of the event
server_time	t	Event time

Notes:

- Possible types of events

  event_type = 1
  message = "session_data_ready"
  All data from the clearing system have been loaded into the trading system

  event_type = 5
  message = "clearing_started"
  Start of the clearing session

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

  event_type = 17
  message = "clearing_trades_inserted"
  Clearing trades have been generated

  event_type = 18
  message = "extension_of_limits_started_for_core"
  Limits have been extended

event_type = 22
message = "discrete_auction_matching_completed"
Order matching completed and opening auction trades have been published

event_type = 23
message = "discrete_auction_add_order_started"
The start of accepting orders in the opening auction

event_type = 24
message = "discrete_auction_add_order_finished"
The finish of accepting orders in the opening auction

event_type = 36
message = "start_of_calendar_date"
Beginning of the calendar day

event_type = 37
message = "settlement_session_started"
Beginning of the settlement session

event_type = 38
message = "settlement_session_finished"
End of the settlement session

event_type = 47
message = "settlpricecalc_finished"
Settlement price has been fixed

4.12. Stream FORTS_POS_REPL - information on positions (Type=I)

4.12.1. Data scheme

Tables:

- position - Client and BRokerage Firm positions
- position_sa - Settlement Account positions
- info - Snapshot information
- sys_events - table of events

4.12.1.1. Table position: Client positions

The table contains information on clients and BF positions.

Table 50. Fields of table position

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
isin_id	i4	Instrument's unique ID
xpos	i8	Current position
xbuys_qty	i8	Number of contracts bought during the trading day
xsells_qty	i8	Number of contracts sold during the trading day
xday_open_qty	i8	Number of positions at the start of the calendar day
xday_open_buys_qty	i8	Number of contracts bought at the start of the calendar day
xday_open_sells_qty	i8	Number of contracts sold at the start of the calendar day
xopen_qty	i8	Number of positions at the start of the trading day
waprice	d16.5	Volume-weighted average price
net_volume_rur	d26.2	Nett volume per trading day, in Rubles. Positive value indicates credited funds, negative value indicates debited funds
last_deal_id	i8	ID of the last deal
last_quantity	i8	Position volume at the end of the clearing session

Field	Type	Description
account_type	i1	• 1 - BF's account • 2 - client's account

4.12.1.2. Table position_sa: Settlement Account positions

The table contains information on Settlement Account positions.

Table 51. Fields of table position_sa

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	c12	Settlement Account code
isin_id	i4	Instrument's unique ID
xpos	i8	Current position
xbuys_qty	i8	Number of contracts bought during the trading day
xsells_qty	i8	Number of contracts sold during the trading day
xday_open_qty	i8	Number of positions at the start of the calendar day
xday_open_buys_qty	i8	Number of contracts bought at the start of the calendar day
xday_open_sells_qty	i8	Number of contracts sold at the start of the calendar day
xopen_qty	i8	Number of positions at the start of the trading day
waprice	d16.5	Volume-weighted average price
net_volume_rur	d26.2	Nett volume per trading trading day, in Rubles. Positive value indicates credited funds, negative value indicates debited funds
last_deal_id	i8	ID of the last deal
last_quantity	i8	Position volume at the end of the clearing session

4.12.1.3. Table info: Snapshot information

Table 52. Fields of table info

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
trades_rev	i8	Revision of the synchronous event start_of_calendar_date
trades_lifenum	i8	Life number of the stream corresponding to the synchronous event start_of_calendar_date
server_time	t	Date and time of position snapshot creation

Notes:

- Field publication_state can take the following values:

  0 in progress

  1 done

4.12.1.4. Table sys_events: table of events

Table 53. 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	Unique ID of the event
sess_id	i4	Trading day ID

Field	Type	Description
event_type	i4	Type of the event
message	c64	Description of the event
server_time	t	Event time

Notes:

- Possible types of events

  event_type = 1
  message = "session_data_ready"
  All data from the clearing system have been loaded into the trading system

  event_type = 5
  message = "clearing_started"
  Start of the clearing session

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

  event_type = 17
  message = "clearing_trades_inserted"
  Clearing trades have been generated

  event_type = 18
  message = "extension_of_limits_started_for_core"
  Limits have been extended

  event_type = 36
  message = "start_of_calendar_date"
  Beginning of the calendar day

  event_type = 37
  message = "settlement_session_started"
  Beginning of the settlement session

  event_type = 38
  message = "settlement_session_finished"
  End of the settlement session

  event_type = 47
  message = "settlpricecalc_finished"
  Settlement price has been fixed

4.13. Stream FORTS_PART_REPL - Information about limits (Type=I)

4.13.1. Data scheme

Tables:

- part - Limits on clients and brokerage firms
- part_sa - Limits by Settlement Account
- sys_events - table of events

4.13.1.1. Table part: Limits on clients and brokerage firms

The table contains information about funds, limits, and settings for automatic limit changes for clients and brokerage firms.

Table 54. Fields of table part

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 or brokerage code
money_free	d26.2	Free limit. Amount of free cash in rubles and pledges, discounted to ruble, available for opening positions. (money_free=money_amount +

Field	Type	Description
		net_option_value - money_blocked - vm_reserve - fee - broker_fee - penalty - premium_blocked)
money_blocked	d26.2	Assets pledged as initial margin.
vm_reserve	d26.2	Variation margin on closed positions, and FX risk.
fee	d26.2	Debited fee
limits_set	i1	Flag of set limits: 1 - limit is set (checked), 0 - limit is not set (not checked)
money_old	d26.2	Total amount of rubles and pledges discounted to rubles at the end of the previous trading day
money_amount	d26.2	Trading limit. Total amount of rubles and pledges discounted to rubles
money_pledge_amount	d26.2	Total amount of pledges, discounted to rubles
actual_amount_of_base_currency	d26.2	Actual amount in rubles (Only for Brokerage Firm, for clients actual_amount_of_base_currency is always 0)
is_auto_update_limit	i1	Flag of automatic adjustment of the limit by the amount of income during downloading after clearing: 0-no, 1-adjust.
moment	t	Moment of the last record update
broker_fee	d26.2	Assets blocked as brokerage fees.
penalty	d26.2	Penalty for settlement trades made into during the procedure of forced closing of positions of the Defaulting Clearing Member
net_option_value	d26.2	Total estimated value of premium options in the portfolio.
premium_blocked	d26.2	Premium reserve. Contains the accumulated financial result of daily turnover of premium option, reserve for orders and currency risk for orders/trades for 7CC with not_use_nov=1.

4.13.1.2. Table part_sa: Limits by Settlement Account

Table 55. Fields of table part_sa

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
settlement_account	c12	Settlement Account
money_old	d26.2	Total amount of rubles and pledges discounted to rubles at the end of the previous trading day
money_amount	d26.2	Trading limit. Total amount of rubles and pledges discounted to rubles
money_free	d26.2	Free limit. Amount of free cash in rubles and pledges, discounted to ruble, available for opening positions. (money_free=money_amount + net_option_value - money_blocked - vm_reserve - fee - blocked_tax)
money_blocked	d26.2	Assets pledged as initial margin.
money_pledge_amount	d26.2	Total amount of pledges, discounted to rubles
actual_amount_of_base_currency	d26.2	Actual amount in rubles
vm_reserve	d26.2	Variation margin on closed positions, and FX risk.
fee	d26.2	Debited fee
blocked_tax	d26.2	Assets blocked for tax payments.
net_option_value	d26.2	Total estimated value of premium options in the portfolio.
moment	t	Moment of the last record update
margin_call_moment	t	Date and time of margin call fixation.
margin_call_fix	d26.2	Fixed margin call.
margin_call_current	d26.2	Current status of fixed margin call.

4.13.1.3. Table sys_events: table of events

Table 56. Fields of table sys_events

Field	Type	Description
replID	i8	Service field of the replication subsystem

Field	Type	Description
replRev	i8	Service field of the replication subsystem
replAct	i8	Service field of the replication subsystem
event_type	i4	Type of the event
event_id	i8	Unique ID of the event
sess_id	i4	Trading day ID
message	c64	Description of the event
server_time	t	Event time

Notes:

- Possible types of events

  event_type = 1
  message = "session_data_ready"
  All data from the clearing system have been loaded into the trading system

  event_type = 5
  message = "clearing_started"
  Start of the clearing session

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

  event_type = 17
  message = "clearing_trades_inserted"
  Clearing trades have been generated

  event_type = 18
  message = "extension_of_limits_started_for_core"
  Limits have been extended

  event_type = 36
  message = "start_of_calendar_date"
  Beginning of the calendar day

  event_type = 37
  message = "settlement_session_started"
  Beginning of the settlement session

  event_type = 38
  message = "settlement_session_finished"
  End of the settlement session

  event_type = 47
  message = "settlpricecalc_finished"
  Settlement price has been fixed

4.14. Stream FORTS_PROHIBITION_REPL - Prohibitions (Type=R)

4.14.1. Data scheme

Tables:

- prohibition - Prohibitions
- sys_events - Table of events

4.14.1.1. Table prohibition: Prohibitions

Table 57. Fields of table prohibition

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
section	c50	Section name

Field	Type	Description
base_contract_code	c25	Underlying asset code.
isin_id	i4	Instrument unique ID
group_mask	i8	Bitmask of groups for which there is a prohibition
xprohibition_id	i8	Number of prohibition
section_id	i4	Section ID
initiator	i4	Prohibition originator
base_contract_id	i4	Underlying contract ID
client_code	c7	Client code
is_legacy	i1	Prohibition originator type
priority	i4	Priority of prohibition
type	i4	Type of prohibition

Notes:

- Field 'initiator' - Initiator of the prohibition:

  1 CF Chief trader;

  2 CC Administrator;

  3 TS Administrator.

- Field 'type' - Prohibition type

  0 No prohibitions. Used for pinpoint permission in case of a broader prohibition;

  1 Prohibited to open positions;

  2 Prohibited to add any orders;

  3 Prohibited to open sell positions;

  0x08 BF prohibition to add requests for exercising;

  0x10 Chief Trader prohibition to add requests for exercising; but to himself - it is possible;

  0x20 Prohibition of requests without auto-confirmation (RFS);

  0x40 Prohibition to request liquidity stream (RFS);

  0x80 Prohibition to perform trades with insufficient number of quotes (RFS);

  0x100 Prohibition to request liquidity stream with limited lifetime of quotes (RFS).

- Field 'group_mask' - Instrument type bitmask:

  0x40000000 Futures.

  0x80000000 Options.

- Field 'priority' - From high to low

  High custom priority 12

  Medium custom priority 11

  Low custom priority 10

  Client code, instrument 9

  Client code, UA 8

  Client code, all UAs 7

  BF code, instrument 6

  BF code, UA 5

  BF code, all UAs 4

  CF code, instrument 3

CF code, UA 2

CF code, all UAs 1

- Field 'section' - Section name:

  1 Securities

  2 Commodities

  3 Money

- Field 'is_legacy' - Prohibition originator type:

  0 indicates the prohibition set by the Trading Administrator/Clearing Administrator; these prohibitions cannot be changed by traders.

  1 indicates the prohibition set by a trader; these prohibitions can be changed by traders.

4.14.1.2. Table sys_events: table of events

Table 58. 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	Unique ID of the event
sess_id	i4	Trading day ID
event_type	i4	Type of the event
message	c64	Description of the event
server_time	t	Event time

Notes:

- Possible types of events

  event_type = 1
  message = "session_data_ready"
  All data from the clearing system have been loaded into the trading system

  event_type = 5
  message = "clearing_started"
  Start of the clearing session

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

  event_type = 17
  message = "clearing_trades_inserted"
  Clearing trades have been generated

  event_type = 18
  message = "extension_of_limits_started_for_core"
  Limits have been extended

  event_type = 36
  message = "start_of_calendar_date"
  Beginning of the calendar day

  event_type = 37
  message = "settlement_session_started"
  Beginning of the settlement session

  event_type = 38
  message = "settlement_session_finished"
  End of the settlement session

  event_type = 47
  message = "settlpricecalc_finished"
  Settlement price has been fixed

4.15. Stream FORTS_REFDATA_REPL - Reference and session information (Type=R)

4.15.1. Data scheme

Tables:

- rates - Currency rates dictionary
- fut_sess_contents - Traded instruments directory (futures)
- fut_vcb - Traded assets directory (futures)
- fut_instruments - Instruments dictionary
- fut_bond_registry - Spot asset parameters directory
- dealer - Companies directory
- sys_messages - Trading system messages
- opt_sess_contents - Traded instruments directory (options)
- opt_vcb - Traded assets directory (options)
- multileg_dict - Multileg instruments dictionary
- opt_exp_orders - Register of requests for exercise of option
- fut_bond_nkd - Accrued interest as of the bond futures contract expiration date
- fut_bond_nominal - Payment of bonds' face value
- fut_bond_isin - Directory of compliance of instruments with spot assets
- user - System users
- sess_option_series - Option series
- investor - Clients directory
- fut_margin_type - Type of margining
- fut_settlement_account - Settlement Account
- sma_master - SMA login binding to MASTER login
- sma_pre_trade_check - SMA login pre-trade verification settings.
- clearing_members - Clearing Members
- instr2matching_map - Instrument binding to Matching ID
- fut_exec_orders - Exercise requests of daily futures contracts with auto-prolongation
- discrete_auction - Parameters of assigned opening auctions
- discrete_auction_base_contract - Underlying contracts assigned to the opening auction
- trade_periods - Trading period parameters
- session - Information about a trading day
- brokers_base_contracts_params - Individual coefficient of IM in the context of the underlying contract and BF
- sys_events - Table of events

4.15.1.1. Table rates: Currency rates dictionary

Table 59. Fields of table rates

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
rate_id	i4	Payment currency identifier
curr_base	c15	Base currency code

Field	Type	Description
curr_coupled	c15	Linked currency code
radius	d16.5	Price indicator change radius (in percent)

4.15.1.2. Table fut_sess_contents: Traded instruments directory (futures)

The table contains dictionary of instruments which are traded in specified trading day.

Table 60. Fields of table fut_sess_contents

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 day ID
isin_id	i4	Instrument unique ID
short_isin	c25	Short symbol code of the instrument for information systems
isin	c25	Symbol code of the instrument
name	c75	Instrument name
inst_term	i4	Shift from spot instruments
base_contract_code	c25	Underlying asset code.
limit_up	d16.5	Upper price limit
limit_down	d16.5	Lower price limit
buy_deposit	d16.2	Collateral of the buyer
sell_deposit	d16.2	Collateral of the seller
roundto	i4	Number of decimal places after the decimal point for the price
min_step	d16.5	Minimum price increment
lot_volume	i4	Lot, i.e. number of units of the underlying asset in the instrument. The field is deprecated and will be removed in version 10.3.
xlot_volume	d16.8	Lot, i.e. number of units of the underlying asset in the instrument. Fractional values are possible.
step_price	d16.5	Value of the minimum price increment
last_trade_date	t	Expiration date.
is_spread	i1	Flag of the futures contract's being part of an intermonth spread 1 – spread; 0 – no spread.
d_exp_start	t	Opening date of instrument expiration
percent_rate	d6.2	Variation margin rate for interest rate futures
settlement_price	d16.5	Settlement price after the last clearing session.
signs	i4	Flags field
ticker	i4	Unique ID number of the primary RTS standard instruments. The field is deprecated and will be removed in version 9.12.
state	i4	State of trading in the instrument
multileg_type	i4	Type of multileg instrument
legs_qty	i4	Number of instruments for multileg instrument
step_price_clr	d16.5	Value of the minumum increment for the clearing session
step_price_curr	d16.5	Value of the minimum increment in currency. Used for contracts with settlements in foreign currency, for ruble contracts the value is the same as 'step_price'.
pctyield_coeff	d16.5	Coef. for yield calculation on percent rates futures
pctyield_total	d16.5	Sum of rates for yield calculation on percent rates futures
d_exp_end	t	Closing date of instrument expiration
enforce_ims_half_netting	i1	Flag - consider the risks of intermonth spread according to the "half-netto" rule: "1" - yes; "0" - no.
tas_base_fut_isin_id	i4	The UA-futures identifier (isin_id) for the TAS-futures. (*Not available in the current version*)

Field	Type	Description
trade_mode_id	i4	The trading mode to which an instrument belongs. One instrument can only belong to one trading mode. The bit number, counting from 1.
group_mask	i8	Bitmask of groups to which the instrument belongs
d_sess_assign	t	Trading start date (calendar date from which orders for the instrument may be submitted)
trade_period_access	i8	The instrument is traded during a specific period of the trading session (bitmask)
buy_deposit_erc	d6.2	Basic size of collateral of the buyer for exceptional-risk client (Russian rubles)
sell_deposit_erc	d6.2	Basic size of collateral of the seller for exceptional-risk client (Russian rubles)
buy_deposit_hrc	d6.2	Basic size of collateral of the buyer for high-risk client (Russian rubles)
sell_deposit_hrc	d6.2	Basic size of collateral of the seller for high-risk client (Russian rubles)
buy_deposit_lrc	d6.2	Basic size of collateral of the buyer for low-risk client (Russian rubles)
sell_deposit_lrc	d6.2	Basic size of collateral of the seller for low-risk client (Russian rubles)
buy_deposit_mrc	d6.2	Basic size of collateral of the buyer for medium-risk client (Russian rubles)
sell_deposit_mrc	d6.2	Basic size of collateral of the seller for medium-risk client (Russian rubles)
settlement_price_clr	d16.5	Settlement price for the future clearing session.

Notes:

- Trading day state has priority over instrument state. That is, if a trading day is in "suspended" or "finished" state, then all instruments can't be traded regardless their states.
- Field state can take the following values:

  0 Trading day for this instrument is scheduled. One can cancel orders for this instrument

  1 Trading day for this instrument is running. One can both add and cancel orders for this instrument.

  2 Trading in all instruments has been suspended. One can cancel orders for each instrument.

  4 Trading day for this instrument has been completed because the time is up. Orders can be neither added nor cancelled

  5 Trading in this instrument has been suspended. One can cancel orders for this instrument

  6 The opening auction for this instrument started. You can put and delete orders for this instrument.

  7 The opening auction for this instrument is completed

  8 Trading day for this instrument is running. You can put orders to close position for this instrument.

  9 The opening auction for this instrument is running. You can put orders to close position for this instrument.
- Field signs is a bit mask and defines the following values:

  0x10 Sign of anonymous trading

  0x20 Sign of non-anonymous trading

  0x100 Sign of multileg-instrument

  0x200 Futures

  0x4000 Daily futures contract with automatic prolongation (CFD - Contract for difference)

  0x10000 Calendar Spread

  0x40000 Collateral

  0x80000 Scheduled execution or execution in clearing session:

  ○ 0 - clearing session

  ○ 1 - scheduled execution

  0x100000 TAS futures (*Not available in the current version*)
- Field multileg_type can take the following values:

0 Ordinary instrument, not the multileg one

3 The instrument is calendar futures spread

- The price precision in clearing trades may exceed the value specified in the 'roundto' field for the instrument, and the price of such a trade may not be a multiple of the minimum price step.
- Field trade_mode_id can take the following values:

  1 Futures, anonymous

  3 Options, anonymous

  14 Calendar spreads, anonymous

  17 Collateral, negotiated

  18 One-day futures with auto-prolongation, anonymous
- Field group_mask is a bitmask and can take the following values:

  0x1 Free

  0x2 HaltGroupMask overflow

  0x4 Prohibition on trading in the additional trading session

  0x10 Scheduled execution

  0x20 Halts related to discrete auctions
- Field trade_period_access is a bitmask and can take the following values:

  0x1 Main

  0x2 Evening

  0x4 Additional weekend trading session

  0x8 Morning

4.15.1.3. Table fut_vcb: Traded assets directory (futures)

The table contains directory of base contracts for instruments.

Table 61. Fields of table fut_vcb

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.
name	c75	Name
exec_type	c1	Settlement type
curr	c3	Quotation currency
trade_scheme	c1	Trading mode
section	c50	Market section. 'Securities', 'Commodities', 'Money'
rate_id	i4	Rate ID
base_contract_id	i4	Underlying contract ID
SECCODE	c12	Code 'SECCODE' of table 'SECURITIES' of ASTS. Default value is NULL.
signs	i4	Flags field
negative_prices	i1	Sign of restriction of negative prices.
option_model	i1	Options pricing model.
asset_class	i4	Underlying asset type.
board_md	c4	'SECBOARD' trading board ID from ASTS gateway.
section_id	i4	Section identifier

Field	Type	Description
expiration_time	t	Expiration time in trades

Notes:

- Field exec_type can take the following values:

  I Cash-settled

  T Delivery via ASTS

  D Delivery by other way (not used)

- Field trade_scheme can take the following values:

  F With 100% collateral

  G With pledge

- Field signs is a bit mask and defines the following values:

  0x1 Scheduled execution or execution in clearing session: 0 - clearing session; 1 - scheduled execution

  0x2 Foreign instrument: 0 - not foreign; 1 - foreign

  0x4 TAS futures: 0 - not TAS futures; 1 - TAS futures (*Not available in the current version*)

  0x8 For qualified investors

- Field negative_prices can take the following values:

  0 Futures prices, price limits and options strikes are limited to be positive only.

  1 Futures prices and options strikes are not limited.

- Field option_model can take the following values:

  0 Black-Scholes model.

  1 Bachelier model.

- Field 'asset_class' may contain the following values:

  1 Share

  2 Currency

  3 Bond

  4 Index

  5 Commodity

  6 Interest rate

- Field 'section_id' may contain the following values:

  1 Stock section

  2 Commodity Section

  3 Money Section

4.15.1.4. Table fut_instruments: Instruments dictionary

Table 62. Fields of table fut_instruments

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
isin_id	i4	Instrument unique ID
short_isin	c25	Short symbol code of the instrument for information systems

Field	Type	Description
isin	c25	Symbol code of the instrument
name	c75	Instrument name
inst_term	i4	Shift from RTS standard instruments
base_contract_code	c25	Underlying asset code.
roundto	i4	Number of decimal places after the decimal point for the price
min_step	d16.5	Minimum price increment
lot_volume	i4	Lot, i.e. number of units of the underlying asset in the instrument. The field is deprecated and will be removed in version 10.3.
xlot_volume	d16.8	Lot, i.e. number of units of the underlying asset in the instrument. Fractional values are possible.
step_price	d16.5	Value of the minimum price increment
last_trade_date	t	Expiration date.
is_spread	i1	Flag of the futures contract's being part of an intermonth spread '1' – spread; '0' – no spread.
d_exp_start	t	Start date of instrument expiration.
percent_rate	d6.2	Variation margin rate for interest rate futures
settlement_price	d16.5	Settlement price after the last clearing session.
signs	i4	Flags field
multileg_type	i4	Type of multileg instrument
legs_qty	i4	Number of instruments for multileg instrument
step_price_clr	d16.5	Value of the minumum increment for the clearing session
step_price_curr	d16.5	Value of the minimum increment in currency. Used for contracts with settlements in foreign currency, for ruble contracts the value is the same as 'step_price'.
pctyield_coeff	d16.5	Coef. for yield calculation on percent rates futures
pctyield_total	d16.5	Sum of rates for yield calculation on percent rates futures
series_type	c1	Futures maturity type. 'M' - monthly; 'Q' - quarterly.
enforce_ims_half_netting	i1	Flag - consider the risks of intermonth spread according to the "half-netto" rule: '1' - yes; '0' - no.
tas_base_fut_isin_id	i4	The UA-futures identifier (isin_id) for the TAS-futures. (*Not available in the current version*)
trade_mode_id	i4	The trading mode to which an instrument belongs. One instrument can only belong to one trading mode. The bit number, counting from 1.
group_mask	i8	Bitmask of groups to which the instrument belongs
trade_period_access	i8	The instrument is traded during a specific period of the trading day (bitmask)

Notes:

- The price precision in clearing trades may exceed the value specified in the 'roundto' field for the instrument, and the price of such a trade may not be a multiple of the minimum price step.
- Field trade_mode_id can take the following values:

  1 Futures, anonymous

  3 Options, anonymous

  14 Calendar spreads, anonymous

  17 Collateral, negotiated

  18 Contract for difference, anonymous
- Field group_mask is a bitmask and can take the following values:

  0x1 Free

  0x2 HaltGroupMask overflow

  0x4 Prohibition on trading in the additional trading session

0x10 Scheduled execution

0x20 Halts related to discrete auctions

- Field trade_period_access is a bitmask and can take the following values:

0x1 Day

0x2 Evening

0x4 Additional weekend trading session

0x8 Morning

4.15.1.5. Table fut_bond_registry: Spot asset parameters directory

Table 63. Fields of table fut_bond_registry

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
bond_id	i4	Spot asset unique ID.
small_name	c25	Spot asset symbol code
short_isin	c25	ISIN code of share/bond issue/currency code
name	c75	Spot asset name
date_redempt	t	Bond’s maturity date (NULL for others)
nominal	d16.5	Bond/share par value
bond_type	i4	Type: share/bond/currency
year_base	i2	Calculation base (conditional number of days in a year)

Notes:

- Field bond_type is a bit mask and defines the following values:

0x1 Share

0x2 Bond

0x4 Amortized bond

0x800000 Currency

4.15.1.6. Table dealer: Companies directory

Table 64. Fields of table dealer

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
name	c200	Company name
rts_code	c50	RTS code of the company
signs	i4	Lock mode. 4 - locked by the Trading System Administrator. 8 - locked by Clearing Firm's Chief Trader.
status	i4	Sign of segregated account
transfer_code	c7	Account code for position transfer
exp_weight	d3.2	Expiration scenario weight for BF, in total collateral. Applied during the clearing session.
num_clr_2delivery	i4	Number of clearing sessions before expiration to start BF expiration scenarios calculation. Applied during the clearing session.
margin_type	i1	Margin type, according to BF's sections, applied during the evening clearing session:

Field	Type	Description
		• 3 - half nett • 4 - nett
calendar_spread_margin_type	i1	Margin type for calendar spreads, for BF portfolio, applied during the evening clearing session: • 3 - half nett • 4 - nett
num_clr_2delivery_client_default	i4	Number of clearing sessions before expiration to start clients expiration scenarios calculation (default value). Applied during the clearing session.
exp_weight_client_default	d3.2	Expiration scenario weight for clients, in total collateral (default value). Applied during the clearing session.
coeff_im	d16.5	Total collateral ratio value, for BF. Applied during the clearing session.
check_limit_on_withdrawal	i1	Verify collateral sufficiency, for BF, upon funds depositing/withdrawal, applied during the evening clearing session: • 1 - Verify • 0 - Do not verify
limit_tied_money	i1	BF trading limit accordance with the BF's total funds amount (all sections): • 1 - maintain accordance • 0 - virtual (independent) limit. The value may change according to the profit/loss values only, resulting from the clearing session. Applied during the clearing session.
limits_set	i1	Verify collateral sufficiency, for BF, upon adding orders: • 1 - Verify • 0 - Do not verify
no_fut_discount	i1	Discount on futures for BF portfolio, applied during the evening clearing session: • 1 - Discount prohibited • 0 - Discount allowed
no_fut_discount_client_default	i1	Discount on futures for BF's clients, default value: • 1 - DIscount prohibited • 0 - DIscount allowed Applied during the clearing session.
firm_id	c12	Trading Member's code for Derivatives Market
tm_name	c200	Trading Member's name
short_option_minimum_charge_ratio	d5.3	Individual coefficient of SOMC scenario weight.
ics_margin_type	i1	Margin type for cross-contract spreads: • 3 - half nett • 4 - nett
order_allowed_in_morning_session	i1	Access to trading during the morning trading session.
order_allowed_in_weekend_session	i1	Access to trading during an additional weekend trading session.
not_use_nov_client_default	i1	Default status of using the special mode for accounting of Premium option for clients of the Brokerage Firm.

Notes:

- Status field is a bit mask:
  - 0x01 - Brokerage Firm (Trust Management type)
  - 0x02 - Segregated Brokerage Firm
  - 0x100 - BF for a client - legal entity

- 0x200 - BF for non-resident client
- 0x10000 - NCC
- 0x20000 – Own Brokerage Firm
- 0x40000 – Client Brokerage Firm
- 0x80000 - Special Brokerage Firm

Other bits contain technical information

- Field order_allowed_in_morning_session can take the following values:

  0 Access to trading during the morning trading session is limited. Trading operations are prohibited, except for orders cancellation operations.

  1 Access to trading during the morning trading session is allowed.

- Field order_allowed_in_weekend_session can take the following values:

  0 Access to trading during an additional weekend trading session is limited. Trading operations are prohibited, except for orders cancellation operations.

  1 Access to trading during an additional weekend trading session is allowed.

- The not_use_nov_client_default field can take the following values:

  0 Standard method for accounting of options.

  1 Special mode (without Initial margin and NOV, with a prohibition on negative positions in premium option).

4.15.1.7. Table sys_messages: Trading system messages

Table 65. Fields of table sys_messages

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
msg_id	i4	Unique message ID
lang_code	c8	Message language
type_id	i4	Message type
moment	t	Message date and time
text	c255	Short text
urgency	i1	Urgency
status	i1	Message status
message_body	c4000	Full text

Message types:

type_id	Message template text
4	Dear Clients, please note that if #БФ# futures price remains at the current level during the next #ShiftTimeMin# minutes, trading will be halted for the #НомерРасширения# expansion of #ГРАНИЦА# bound of the price band and #РИСКОВ#.
5	Trading in #БФ# futures has been halted for the #НомерРасширения# expansion of #ГРАНИЦА# bound of the price band and #РИСКОВ#.
6	Trading in #БФ# futures has been resumed. New #ГРАНИЦА# bounds of the price band: #UpCotir/DownCotir#, new #NEWRATE#.
7	Initial margin are #AmountGO# #valuta#, change to the previous day #DiffAmountGO# #valuta#.
8	Trading in #БФ# futures has been halted.
9	Trading in #БФ# futures has been resumed.
11	Contacts for liquidity providers and block trade enabling Brokers - https://www.moex.com/s3005
12	Please, pay attention that today is the last trading day of #ТЕКУЩМЕСЯЦ# #НЕДЕЛЬНЫЕ?weekly##АМЕРИКАНСКИХ/ЕВРОПЕЙСКИХ# #МАРЖИРУЕМЫХ/ПРЕМИАЛЬНЫХ# #РАСЧЕТНЫХ/ПОСТАВОЧНЫХ# equity options (#БФ#).

type_id	Message template text
13	Orders can be cancelled now.
18	Currency rates for the Variation margin and the Initial margin calculation during the clearing session: CAD=51.88280, CHF=76.18050, EUR=76.22500, GBP=84.79190, HKD=9.03970, JPY=0.53061, TRY=3.75870, USR=72.08270.
19	Exercise price calculated…
500	Free form text field for standard messages
501	Free form text field for non-standard messages

4.15.1.8. Table opt_sess_contents: Traded instruments directory (options)

The table contains dictionary of instruments which are traded in specified trading day.

Table 66. Fields of table opt_sess_contents

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 day ID
isin_id	i4	Instrument unique ID
isin	c25	Symbol code of the instrument
short_isin	c25	Short symbol code of the instrument for information systems
name	c75	Instrument name
base_contract_code	c25	Underlying asset code.
fut_isin_id	i4	ID of the futures instrument
base_im_covered_sell	d16.2	Basic size of the collateral to be posted on one open position of the option writer (Russian rubles).
base_im_sell	d16.2	Basic size of collateral to be posted on one unsecured position of the option writer (Russian rubles).
put	i1	Option's type. 0 - Call option, 1 - Put option
strike	d16.5	Strike price
roundto	i4	Number of decimal places after the decimal point for the price
last_trade_date	t	Expiration date.
signs	i4	Flags field
settlement_price	d16.5	Settlement price (theoretical price of the option) after the last clearing session.
base_im_buy	d16.2	Basic size of Collateral requested in order to buy a futures-style option.
option_series_id	i4	Series of Options ID
state	i4	The state of trading for the instrument.
trade_mode_id	i4	The trading mode to which an instrument belongs. One instrument can only belong to one trading mode. The bit number, counting from 1.
group_mask	i8	Bitmask of groups to which the instrument belongs
trade_period_access	i8	The instrument is traded during a specific period of the trading day (bitmask)
base_im_covered_sell_erc	d16.2	Basic size of collateral to be posted on one open position of the option writer for exceptional-risk client (Russian rubles)
base_im_covered_sell_hrc	d16.2	Basic size of collateral to be posted on one open position of the option writer for high-risk client (Russian rubles)
base_im_covered_sell_lrc	d16.2	Basic size of collateral to be posted on one open position of the option writer for low-risk client (Russian rubles)
base_im_covered_sell_mrc	d16.2	Basic size of collateral to be posted on one open position of the option writer for medium-risk client (Russian rubles)
base_im_sell_erc	d16.2	Basic size of collateral to be posted on one unsecured position of the option writer for exceptional-risk client (Russian rubles)
base_im_buy_erc	d16.2	Basic size of collateral requested in order to buy a futures-style option for exceptional-risk client (Russian rubles)

Field	Type	Description
base_im_sell_hrc	d16.2	Basic size of collateral to be posted on one unsecured position of the option writer for high-risk client (Russian rubles)
base_im_buy_hrc	d16.2	Basic size of collateral requested in order to buy a futures-style option for high-risk client (Russian rubles)
base_im_sell_lrc	d16.2	Basic size of collateral to be posted on one unsecured position of the option writer for low-risk client (Russian rubles)
base_im_buy_lrc	d16.2	Basic size of collateral requested in order to buy a futures-style option for low-risk client (Russian rubles)
base_im_sell_mrc	d16.2	Basic size of collateral to be posted on one unsecured position of the option writer for medium-risk client (Russian rubles)
base_im_buy_mrc	d16.2	Basic size of collateral requested in order to buy a futures-style option for medium-risk client (Russian rubles)
settlement_price_clr	d16.5	Settlement price for the future clearing session.
d_sess_assign	t	Trading start date (calendar date from which orders for the instrument may be submitted)

Notes:

- The price precision in clearing trades may exceed the value specified in the 'roundto' field for the instrument, and the price of such a trade may not be a multiple of the minimum price step.
- Field signs is a bit mask and defines the following values:

  0x10 Sign of anonymous trading

  0x20 Sign of non-anonymous trading
- Field state can take the following values:

  0 Trading day for this instrument is scheduled. One can cancel orders for this instrument

  1 Trading day for this instrument is running. One can both add and cancel orders for this instrument

  2 Trading in all instruments has been suspended. One can cancel orders for each instrument.

  4 Trading day for this instrument has been completed because the time is up. Orders can be neither added nor cancelled

  5 Trading in this instrument has been suspended. One can cancel orders for this instrument

  6 The opening auction for this instrument started. You can put and delete orders for this instrument.

  7 The opening auction for this instrument is completed

  8 Trading day for this instrument is running. You can put orders to close position for this instrument.

  9 The opening auction for this instrument is running. You can put orders to close position for this instrument.
- Field trade_mode_id can take the following values:

  1 Futures, anonymous

  3 Options, anonymous

  14 Calendar spreads, anonymous

  17 Collateral, negotiated

  18 One-day futures with auto-prolongation, anonymous
- Field group_mask is a bitmask and can take the following values:

  0x1 Free

  0x2 HaltGroupMask overflow

  0x4 Prohibition on trading in the additional trading session

  0x10 Scheduled execution

  0x20 Halts related to discrete auctions
- Field trade_period_access is a bitmask and can take the following values:

0x1 Main

0x2 Evening

0x4 Additional weekend trading session

0x8 Morning

4.15.1.9. Table opt_vcb: Traded assets directory (options)

The table contains dictionary of underlying contracts for instruments.

Table 67. Fields of table opt_vcb

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.
name	c75	Name
exec_type	c1	Settlement type
curr	c3	Quotation currency
trade_scheme	c1	Trading mode
rate_id	i4	Rate ID
base_contract_id	i4	Underlying contract ID
negative_prices	i1	Sign of restriction of negative prices.
option_model	i1	Options pricing model.
settlement_currency	c3	Settlement currency
section_id	i4	Section identifier

Notes:

- Field negative_prices can take the following values:

  0 Futures prices, price limits and options strikes are limited to be positive only.

  1 Futures prices and options strikes are not limited.

- Field option_model can take the following values:

  0 Black-Scholes model.

  1 Bachelier model.

- Field 'section_id' may contain the following values:

  1 Stock section

  2 Commodity Section

  3 Money Section

4.15.1.10. Table multileg_dict: Multileg instruments dictionary

Table 68. Fields of table multileg_dict

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 day ID
isin_id	i4	Multileg instrument ID
isin_id_leg	i4	ID of the instrument which is a component of specified multileg instrument
qty_ratio	i4	Quantity ratio
leg_order_no	i1	Leg order in a multileg instrument. The default value is 0.

Notes:

- The meaning of the filed qty_ratio is specifying the number and direction of the multileg instrument: If the value equals qty_ratio > 0 then this instrument is a multileg instrument with the same direction with which is the multileg order, if qty_ratio < 0 – with opposite. Absolute value of qty_ratio specifies the coefficient by which the number of multileg instruments in the order should be multiplied in order to get the number of instruments isin_id_leg.

4.15.1.11. Table opt_exp_orders: Register of requests for exercise of option

Table 69. Fields of table opt_exp_orders

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
exporder_id	i8	ID of the request to exercise the options
client_code	c7	Client code
isin_id	i4	Instrument unique ID
xamount	i8	Number of expiring positions
sess_id	i4	Trading day ID
date	t	Date and time

4.15.1.12. Table fut_bond_nkd: Accrued interest as of the bond futures contract expiration date

Table 70. Fields of table fut_bond_nkd

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
bond_id	i4	ID of the bond
date	t	Coupon payment date
nkd	d16.7	Accrued interest as of the coupon payment date
is_cupon	i1	Flags: 0 - accrued interest as of the bond futures contract settlement date, 2 - accrued interest as of the bond settlement date

4.15.1.13. Table fut_bond_nominal: Payment of bonds’ face value

Table 71. Fields of table fut_bond_nominal

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
bond_id	i4	ID of the bond
date	t	Coupon payment date
nominal	d16.5	payment of bonds’ face value
face_value	d16.5	Payment of bonds’ rest face value
coupon_nominal	d8.5	Coupon value in % of face value
is_nominal	i1	Type of record in the table: 0 - Remaining face value as of bond futures contract expiration date, 2 - Remaining face value as of bond settlement date

4.15.1.14. Table fut_bond_isin: Directory of compliance of instruments with spot assets

Table 72. Fields of table fut_bond_isin

Field	Type	Description
replID	i8	Service field of the replication subsystem
replRev	i8	Service field of the replication subsystem

Field	Type	Description
replAct	i8	Service field of the replication subsystem
isin_id	i4	Instrument unique ID
bond_id	i4	Spot asset unique ID
coeff_conversion	d5.4	Conversion ratio

4.15.1.15. Table user: System users

Table 73. Fields of table user

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
login	c20	Trading participant's login
start_date	t	Login start time
end_date	t	Login end time
client_code	c7	7-symbol client code
operation_mask	i4	Bitmask: • 2 - Limit open positions for BF • 8 - Limit BF on funds transfer. The setting is available to Clearing Firm operator or Trading Administrator only. • 16 - Money back. • 32 - Limit client positions. • 128 - Client restrictions. • 1024 - Orders-related restrictions for SMA logins.
lang	i2	Message language code
sma_flags	i4	Bitmask (see Notes below): • 1st bit - Cancel on Disconnect • 2nd bit - Cancel on DropCopy Disconnect • 3rd bit - SMA login.
sma_status	i4	Bitmask (see Notes below): • 1st bit - enable/disable trading transactions for the login. • 2nd bit - cancel/do not cancel orders when trading transactions are disabled for the login.
aspref	i4	Client ID. For orders added by SMA login - MASTER login ID.
user_level	i1	User login level: • 1 - CF • 2 - BF • 3 - Client
password_expiration_date	t	Password expiration date.

Notes:

- Field 'sma_flags' is bitmask:
  - 1st bit: 0 - Cancel on Disconnect is disabled for the login, 1 - Cancel on Disconnect is enabled for the login
  - 2nd bit: 0 - Cancel on Drop-Copy Disconnect is disabled for the login, 1 - Cancel on Drop-Copy Disconnect is enabled for the login
  - 3rd bit: 0 - SMA mode is disabled for the login, 1 - SMA mode is enabled for the login.
- Field 'sma_status' is bitmask::

- 1st bit: 0 - trading transactions are enabled for the login, 1 - trading transactions are disabled for the login
- 2nd bit: 0 - do not cancel orders when trading transactions are disabled for the login, 1 - cancel orders when trading transactions are disabled for the login.

4.15.1.16. Table sess_option_series: Option series

Table 74. Fields of table sess_option_series

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
series_id	i4	Series of Options ID
underlying_id	i4	Futures ID
base_contract_id	i4	Underlying contract ID
expiration_date	t	Expiration period closing date
expiration_anchor_date	t	Anchor date (expiration date of series of Options)
days_to_expiration	i4	Number of business days (local calendar) before expiration date
years_to_expiration	f	Time before Option expiration date in fractions of year (calendar days before exercise date/number of days in year)
series_type	c1	Option maturity type . W-weekly; M-monthly; Q-quarterly.
small_name	c25	Symbol code
strike_step	d16.5	Strike step
use_null_volat	i1	1 - Zero volatility calculation mode is on, 0 - Off
sub_risk	i1	1 - Risk accounting by risk sub-points mode is on, 0 - Off
volat_min	d20.15	Minimum volatility limit
volat_max	d20.15	Maximum volatility limit
volatility_risk	f	Current volatility risk rate, in fractions
volatility_mismatch_risk	f	Current volatility mismatch risk rate, in fractions
signs	i4	Flags field
a_black	f	Volatility curve calculation parameter for Black-Scholes model.
b_black	f	Volatility curve calculation parameter for Black-Scholes model.
c_black	f	Volatility curve calculation parameter for Black-Scholes model.
d_black	f	Volatility curve calculation parameter for Black-Scholes model.
e_black	f	Volatility curve calculation parameter for Black-Scholes model.
s_black	f	Volatility curve calculation parameter for Black-Scholes model.
a_bach	f	Volatility curve calculation parameter for Bachelier model.
b_bach	f	Volatility curve calculation parameter for Bachelier model.
c_bach	f	Volatility curve calculation parameter for Bachelier model.
d_bach	f	Volatility curve calculation parameter for Bachelier model.
e_bach	f	Volatility curve calculation parameter for Bachelier model.
s_bach	f	Volatility curve calculation parameter for Bachelier model.
m_bach	f	Volatility curve calculation parameter for Bachelier model.
r	f	Risk-free interest rate.
fixed_spot_discount	f	Sum of discounted values of declared cash flows. CF calculated on the next day T+1
projected_spot_discount	f	Sum of discounted values of forecasted cash flows.
margin_style	i4	Option margin method. 0 - Futures-style option; 1 - Equity-style option.
settlement_type	i4	Option type. 0 - Cash-settled; 1 - Deliverable.
exercise_style	i4	Exercise style of option. 0 - American; 1 - European.
min_step	d16.5	The minimum price movement.
step_price	d16.5	Price step cost.

Field	Type	Description
lot_coefficient	i4	Coefficient indicating the volume of the underlying asset in the contract quote and strikes of option series.
interest_rate_risk_up	f	Interest rate mismatch rate in the scenario of rate movement ‘r’ upward.
interest_rate_risk_down	f	Interest rate mismatch rate in the downward rate movement scenario ‘r’.
step_price_curr	d16.5	Price step cost in currency. Used for contracts with settlements in foreign currency, for ruble contracts the value is the same as 'step_price'.
underlying_price	d16.5	The current spot price of the instrument, or on expiration date, the price of the underlying asset according to which the options will be exercised (see the relevant contract specification for details).
lot_volume	i4	Lot, i.e. number of units of the underlying asset in the instrument.
sess_id	i4	Trading day ID
step_price_clr	d16.5	The cost of the clearing session price step.
r2	f	Risk-free interest rate FX2 of the currency pair FX2/FX1 (for premium options on the currency); dividend yield rate ‘q’ (for premium options on the index).
interest_rate2_risk_up	f	Interest rate mismatch rate in the scenario of rate movement ‘r2’ upward.
interest_rate2_risk_down	f	Interest rate mismatch rate in the downward rate movement scenario ‘r2’.
d_sess_assign	t	Trading start date (calendar date from which orders for the instrument may be submitted)
expiration_time	t	Expiration time in trades
fixed_spot_discount_clr	f	Sum of discounted values of declared cash flows. CF calculated at the beginning of the trading day T

Notes:

- Field signs is a bit mask and defines the following values:

  0x1 Scheduled execution or execution in clearing session: 0 - clearing session; 1 - scheduled execution

4.15.1.17. Table investor: Clients directory

Table 75. Fields of table investor

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.
name	c200	Client name.
calendar_spread_margin_type	i1	Margin type for client calendar spread, applied during the evening clearing session: • 3 - half nett • 4 - nett
is_blank	i4	The account-blank flag for online registration
short_option_minimum_charge_ratio	d5.3	Individual coefficient of SOMC scenario weight.
ics_margin_type	i1	Margin type for cross-contract spreads: • 3 - half nett • 4 - nett
coeff_im	d16.5	Total collateral ratio value.
no_fut_discount	i1	Discount on futures: • 1 - Discount prohibited • 0 - Discount allowed
num_clr_2delivery	i4	Number of clearing sessions before expiration to start expiration scenarios calculation.
exp_weight	d3.2	Expiration scenario weight, in total collateral.

Field	Type	Description
xstatus	i8	Client's flags, extended.
client_risk_level	i4	Client risk level category.
not_use_nov	i4	Status of using the special mode for accounting of premium option (without Initial margin and NOV, with a prohibition on negative positions in premium option).

Notes:

- Xstatus field is a bit mask:
  - 0x1 - Trust Management
  - 0x2 - Separated
  - 0x4 - Brokerage Firm (Trust Management type)
  - 0x80 - Private entity
  - 0x100 - Legal entity
  - 0x200 - Non-resident
  - 0x2000 - Individual investment account
  - 0x4000 - Flag for allowing cross-trades for negotiated orders. 1 - cross-trades allowed; 0 - cross-trades prohibited
  - 0x8000 - Stateless person
  - 0x20000 - Own
  - 0x40000 - Client
  - 0x80000 - Special BF
  - 0x10000000 - Additional own account
  - 0x10000000000 - Qualified investor
  - 0x40000000000 - Cancel a passive order in a cross-trade
- Field client_risk_level can take the following values:
  - 0 - Not specified (default value)
  - 1 - Low-risk client
  - 2 - Medium-risk client
  - 3 - High-risk client
  - 4 - Exceptional-risk client
- The not_use_nov field can take the following values:
  - -1 - "Same as for BF" (inherited from dealer.not_use_nov_client_default)
  - 0 - Standard method for accounting of options
  - 1 - Special mode (without Initial margin and NOV, with a prohibition on negative positions in Premium option)

4.15.1.18. Table fut_margin_type: Type of margining

Table 76. Fields of table fut_margin_type

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
code	c12	Settlement Account or Brokerage Firm Code
type	i1	Type of Code. Settlement Account - 0, Brokerage Firm - 1.
margin_type	i1	Type of margining. 2 - Gross, 3 - Half nett, 4 - Nett.
UCP_type	i1	UCP-related Settlement Account.

Field	Type	Description
prohibit_coeff	d16.2	Debt coefficient value for Settlement Account/Brokerage Firm/section. The value defines the maximum correlation between negative free limit volume and trading limit volume. As the value exceeded, the system prohibits operations. The prohibition mode is specified by field 'prohibit_type'.
prohibit_type	i4	Type of automatic prohibition for Settlement Account: • 1 - prohibited to open positions • 2 - prohibited to add orders.
settlement_account_type	i1	Settlement Account Type. 0 - own SA, 1 - client SA, 2 - SA (Trust Management type).
operator_input	i1	Settlement account blocking set by the TS Administrator: • 0 - blocking off • 1 - blocking on.
spf	i1	spf - special flag Indicator that the Settlement Account is open for a non-resident (citizen of an unfriendly country): 0 - standard Settlement Account; 2 - Settlement Account "Ин"

Notes:

- Possible 'operator_input' field values: 0 - blocking off, 1 - blocking on. When the blocking mode is turned on, orders placed from all BF clearing accounts linked to the blocked SA are automatically cancelled. The cancelled orders in the 'xstatus' field are marked with a special sign - 'OperatorInputSA' (0x1000000000000). In the blocking mode, any trading commands with the indication of BF clearing accounts linked to this SA are prohibited, and the positions transfer between BFs is also prohibited. Orders and trades formed for SA by the Trading Administrator in blocking mode, have a special sign in the 'xstatus' field (in orders) and 'xstatus_sell' or 'xstatus_buy' fields (in trades) - 'OperatorInputSA' (0x1000000000000).

4.15.1.19. Table fut_settlement_account: Settlement Account

Table 77. Fields of table fut_settlement_account

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
code	c7	Brokerage Firm Code or Client Code
type	i1	Brokerage Firm - 1, Client - 2
settlement_account	c12	Settlement Account

4.15.1.20. Table sma_master: SMA login binding to MASTER login

The table contains information on how SMA login is binding to MASTER login.

Table 78. Fields of table sma_master

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
sma_asp	c20	SMA login
sma_aspref	i4	SMA login ID
master_asp	c20	MASTER login
master_aspref	i4	MASTER login ID

4.15.1.21. Table sma_pre_trade_check: SMA login pre-trade verification settings

The table contains information on SMA login pre-trade verification settings.

Table 79. Fields of table sma_pre_trade_check

Field	Type	Description
replID	i8	Service field of the replication subsystem

Field	Type	Description
replRev	i8	Service field of the replication subsystem
replAct	i8	Service field of the replication subsystem
check_id	i8	Unique record ID.
sma_asp	c20	SMA login
sma_aspref	i4	SMA login ID
check_number	i1	Verification number (1 - 7).
base_contract_code	c25	Underlying asset code.
instrument_type	i1	Instrument type: • 0 - Futures • 1 - Option • 3 - Calendar spread
client_code_check	c7	Client code under verification.
value	d26.2	Verification number.

4.15.1.22. Table clearing_members: Clearing Members

The table contains information about blocking of members.

Table 80. Fields of table clearing_members

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
code	c2	Member code
lock_type	i1	Blocking type
lock_date	t	Blocking date
name	c200	Member name

Notes:

- Field lock_type can take the following values:

  0 No blocking

  2 Liquidation netting in respect of the Clearing Member

  11 Suspension of clearing services for the Clearing Member due to cancellation of the license to carry out professional activities in the securities market

4.15.1.23. Table instr2matching_map: Instrument binding to Matching ID

The table contains information on how instrument is binding to Matching ID.

Table 81. Fields of table instr2matching_map

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_id	i4	Underlying contract ID
matching_id	i1	Matching ID

4.15.1.24. Table fut_exec_orders: Exercise requests of daily futures contracts with auto-prolongation

Table 82. Fields of table fut_exec_orders

Field	Type	Description
replID	i8	Service field of the replication subsystem
replRev	i8	Service field of the replication subsystem

Field	Type	Description
replAct	i8	Service field of the replication subsystem
exec_order_id	i8	ID of the request to exercise the futures
client_code	c7	Client code
isin_id	i4	Instrument unique ID
xamount	i8	Number of contracts to be exercised (including the sign)
sess_id	i4	Trading day ID
date	t	Date and time of apply/ change of the request

4.15.1.25. Table discrete_auction: Parameters of assigned opening auctions

Table 83. Fields of table discrete_auction

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
discrete_auction_id	i8	Opening auction ID
sess_id	i4	Trading day ID
add_order_start	t	The start time for accepting orders to the opening auction
add_order_finish_from	t	The beginning of the time interval during which the acceptance of orders to the opening auction will stop
add_order_finish_till	t	The end of the time interval during which the acceptance of orders to the opening auction will stop
synchro_event_id_start	i4	Sync event ID 'discrete_auction_add_order_started'
synchro_event_id_finish	i4	Sync event ID 'discrete_auction_add_order_finished'
trade_period_id	i8	Trading period ID
trade_period_type	i8	Trading period type

Notes:

- Field trade_period_type can take the following values:
  - 1 Main trading session.
  - 2 Evening additional trading session.
  - 4 Additional weekend trading session.
  - 8 Morning additional trading session.

4.15.1.26. Table discrete_auction_base_contract: Underlying contracts assigned to the opening auction

Table 84. Fields of table discrete_auction_base_contract

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
discrete_auction_id	i8	Opening auction ID
base_contract_id	i4	Underlying contract ID
base_contract_code	c25	Underlying contract code.

4.15.1.27. Table trade_periods: Trading period parameters

The table contains trading periods timetable.

Table 85. Fields of table trade_periods

Field	Type	Description
replID	i8	Service field of the replication subsystem

Field	Type	Description
replRev	i8	Service field of the replication subsystem
replAct	i8	Service field of the replication subsystem
trade_period_id	i8	Trading period ID
sess_id	i4	Trading day ID
trade_period_type	i8	Trading period type
trade_period_begin	t	Date and time of the start of the trading period
trade_period_end	t	Date and time of end of trading period
trade_period_name	c255	Trading period name
trade_period_flags	i8	Additional properties' flags of the trading period

Notes:

- Field trade_period_type can take the following values:

  1 Main trading session.

  2 Evening additional trading session.

  4 Additional weekend trading session.

  8 Morning additional trading session.

- Field trade_period_flags can take the following values:

  0x01 Pseudo extension was performed before the trading period.

4.15.1.28. Table session: Information about a trading day

The table contains trading days timetable.

Table 86. Fields of table session

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 day ID
begin	t	Opening time
end	t	Closing time
state	i4	Trading day status
eve_on	i1	Flag of holding an additional evening session
eve_begin	t	Time when the additional evening session starts
eve_end	t	Time when the additional evening session is over
mon_on	i1	Flag of holding an additional morning session
mon_begin	t	Time when the additional morning session starts
mon_end	t	Time when the additional morning session is over
settl_sess_begin	t	Date and time of the start of the settlement session
clr_sess_begin	t	Date and time of the start of the clearing session
settl_price_calc_time	t	Date and time of the settlement price determination process
settl_sess_t1_begin	t	Planned date and time of the start of the settlement session in trading day T+1
margin_call_fix_schedule	t	Date and time of automatic margin call fixation

Notes:

- Field state can take the following values:

  0 Trading day is scheduled. Orders can't be placed but can be cancelled.

  1 Trading day is running. Orders can be both placed and cancelled.

  2 Trading with all instruments is suspended. Orders can't be placed but can be cancelled.

4 Trading day is completed because the time is up. Orders can be neither added nor cancelled.

- Field inter_cl_state is a bit mask:

  0x0 It is not defined. Orders can be both placed and cancelled.

  0x01 It is scheduled today. Orders can be placed and cancelled.

  0x02 It is cancelled. Orders can be placed and cancelled.

  0x04 Current, i.e.it is running, nothing can be done. Orders can't be placed and cancelled.

  0x08 Current, i.e. it is running (due to time schedule), but actually it is over and intraday clearing data is already available. Orders can't be placed but can be cancelled.

  0x10 It is successfully over (due to time schedule as well). Orders can be placed and cancelled.

4.15.1.29. Table brokers_base_contracts_params: Individual coefficient of IM in the context of the underlying contract and BF

Table 87. Fields of table brokers_base_contracts_params

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
broker_code	c4	Brokerage Firm code
base_contract_code	c25	Underlying contract code.
coeff_im	d16.5	Coefficient of IM under the underlying contract for all clients of this BF. Applied during the clearing session.

4.15.1.30. Table spectra_environment: Description of the current environment

The table provides the label of the current production environment to which the client is connected.

Table 88. Fields of table spectra_environment

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
environment_id	i4	Environment ID
environment_name	c8	Environment name

Notes:

- The spectra_environment table contains exactly one record:

  Either 0x01 | MXFOPROD - Production (PROD)

  Or 0x02 | MXFOPREP - Test (PRE-PROD)

4.15.1.31. Table sys_events: table of events

Table 89. 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_type	i4	Type of the event
event_id	i8	Unique ID of the event
sess_id	i4	Trading day ID
message	c64	Description of the event
server_time	t	Event time

Notes:

- Possible types of events

  event_type = 1
  message = "session_data_ready"
  All data from the clearing system have been loaded into the trading system

  event_type = 5
  message = "clearing_started"
  Start of the clearing session

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

  event_type = 17
  message = "clearing_trades_inserted"
  Clearing trades have been generated

  event_type = 18
  message = "extension_of_limits_started_for_core"
  Limits have been extended

  event_type = 22
  message = "discrete_auction_matching_completed"
  Order matching completed and opening auction trades have been published

  event_type = 23
  message = "discrete_auction_add_order_started"
  The start of accepting orders in the opening auction

  event_type = 24
  message = "discrete_auction_add_order_finished"
  The finish of accepting orders in the opening auction

  event_type = 36
  message = "start_of_calendar_date"
  Beginning of the calendar day

  event_type = 37
  message = "settlement_session_started"
  Beginning of the settlement session

  event_type = 38
  message = "settlement_session_finished"
  End of the settlement session

  event_type = 47
  message = "settlpricecalc_finished"
  Settlement price has been fixed

4.16. Stream FORTS_MM_REPL - information on MM's obligations (Type=I)

4.16.1. Data scheme

Tables:

- mm_agreement_filter - Table with numbers, types of contracts and codes of programs for the provision of market-making services
- fut_MM_info - MM's obligations in futures
- opt_MM_info - MM's obligations in options
- cs_mm_rule - Instruments for recalculating the central strike price.

4.16.1.1. Table mm_agreement_filter: Table with numbers, types of contracts and codes of programs for the provision of market-making services

Table 90. Fields of table mm_agreement_filter

Field	Type	Description
replID	i8	Service field of the replication subsystem
replRev	i8	Service field of the replication subsystem

Field	Type	Description
replAct	i8	Service field of the replication subsystem
agmt_id	i4	Identifier of the agreement
is_fut	i1	Type of obligation
agreement	c50	Number of the agreement
client_code	c7	Client code
program_code	c50	Program code

4.16.1.2. Table fut_MM_info: MM's obligations in futures

Table 91. Fields of table fut_MM_info

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
isin_id	i4	Instrument unique ID
sess_id	i4	Trading day ID
spread	d16.5	Spread in points
price_edge_sell	d16.5	Price of the worst sell order included in the spread
xamount_sells	i8	Number of contracts in the sell order included in the spread
price_edge_buy	d16.5	Price of the worst buy order included in the spread
xamount_buys	i8	Number of contracts in the buy order included in the spread
mm_spread	d16.5	Agreed spread
xmm_amount	i8	Number in accordance with the agreement
spread_sign	i1	Sign: 1-spread is not maintained, 0-spread is maintained
amount_sign	i1	Sign: 1- number is not maintained, 0- number is maintained
percent_time	d6.2	% of fulfilled obligations
period_start	t	Start of the period of MM rules coming into force
period_end	t	End of the period of MM rules coming into force
client_code	c7	Client code
active_sign	i4	Sign: 1-note is deleted (stopped being active), 0-is active
agmt_id	i4	Identifier of the MM agreement
fulfil_min	d6.2	Minimum percentage of the liabilities for the trading day
fulfil_partial	d6.2	Percentage of partial fulfillment of the obligations of the trading day
fulfil_total	d6.2	Percentage of fulfillment of obligations of the trading day
is_fulfil_min	i1	Minimum sign of the liabilities for the trading session
is_fulfil_partial	i1	Sign of partial fulfillment of the obligations of the trading
is_fulfil_total	i1	Sign of fulfillment of obligations of the trading session
is_rf	i1	Sign of clearing member market-maker requirement
id_group	i4	ID of market-maker group of instrument

Notes: The 'fut_MM_info' table of the 'FORTS_MM_REPL' stream contains market-makers obligations accurate to 7-symbol client code.

4.16.1.3. Table opt_MM_info: MM's obligations in options

Table 92. Fields of table opt_MM_info

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
isin_id	i4	Instrument unique ID
sess_id	i4	Trading day ID

Field	Type	Description
spread	d16.5	Spread in points
price_edge_sell	d16.5	Price of the worst sell order included in the spread
xamount_sells	i8	Number of contracts in the sell order included in the spread
price_edge_buy	d16.5	Price of the worst buy order included in the spread
xamount_buys	i8	Number of contracts in the buy order included in the spread
mm_spread	d16.5	Agreed spread
xmm_amount	i8	Number in accordance with the agreement
spread_sign	i1	Sign: 1-spread is not maintained, 0-spread is maintained
amount_sign	i1	Sign: 1- number is not maintained, 0- number is maintained
percent_time	d6.2	% of fulfilled obligations
period_start	t	Start of the period of MM rules coming into force
period_end	t	End of the period of MM rules coming into force
client_code	c7	Client code
cstrike_offset	d16.5	Central Strike offset
active_sign	i4	Sign: 1-note is deleted (stopped being active), 0-is active
agmt_id	i4	Identifier of the MM agreement
fulfil_min	d6.2	Minimum percentage of the liabilities for the trading day
fulfil_partial	d6.2	Percentage of partial fulfillment of the obligations of the trading day
fulfil_total	d6.2	Percentage of fulfillment of obligations of the trading day
is_fulfil_min	i1	Minimum sign of the liabilities for the trading session
is_fulfil_partial	i1	Sign of partial fulfillment of the obligations of the trading
is_fulfil_total	i1	Sign of fulfillment of obligations of the trading session
is_rf	i1	Sign of clearing member market-maker requirement
id_group	i4	ID of market-maker group of instrument

Notes: The 'opt_MM_info' table of the 'FORTS_MM_REPL' stream contains market-makers obligations accurate to 7-symbol client code.

4.16.1.4. Table cs_mm_rule: Instruments for recalculating the central strike price.

Table 93. Fields of table cs_mm_rule

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 day ID
client_code	c4	Client code
option_series_id	i4	Series of Options ID

4.17. Stream FORTS_CLR_REPL - clearing information (Type=AR)

4.17.1. Data scheme

Tables:

- money_clearing - Balances for clients and brokerage firms
- money_clearing_sa - Balances by Settlement Account
- clr_rate - Currency and Index rates
- fut_pos - Positional state in futures as a result of clearing session
- opt_pos - Positional state in options as a result of clearing session
- fut_sess_settl - Futures settlement prices
- opt_sess_settl - Options settlement prices

- pledge_details - Pledgs details table
- fut_pos_sa - Positional state of SA on futures as a result of clearing session
- opt_pos_sa - Positional state of SA on options as a result of clearing session
- option_series_settl - Settlement prices for option series
- sys_events - table of events

4.17.1.1. Table money_clearing: Balances for clients and brokerage firms

Table 94. Fields of table money_clearing

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
amount_beg	d16.2	Actual balance of funds at the start of the trading day
vm	d16.2	Actually calculated (in Settlement Session) variation margin, including variation margin for marginable options
premium	d16.2	Actually calculated (in Settlement Session) premium for options received/paid in the clearing session, converted into rubles. Includes the financial result from option exercise
fee_fut	d16.2	Actually calculated (in Settlement Session) amount of exchange and clearing fees for anonymous trades on futures, included in charges
fee_opt	d16.2	Actually calculated (in Settlement Session) amount of exchange and clearing fees for anonymous trades on options, included in charges
go	d16.2	Total initial margin on futures and options
amount_end	d21.2	Current balance of funds at the time of balance formation during the Clearing Session or Settlement Session
free	d22.2	Available funds
date_money_clearing	t	Date and time of balance formation during the Clearing Session or Settlement Session
settl_datetime	t	Scheduled settlement date for obligations
in_out_netto	d26.2	Current net balance of deposits/withdrawals
vm_mtm	d26.2	Scheduled obligation (defined in the Clearing Session) for variation margin, including variation margin for marginable options
premium_mtm	d26.2	Scheduled obligation (defined in the Clearing Session) for option premiums, including the financial result from option exercise
charges	d16.2	Actually calculated (in the Settlement Session) obligations for payment of fees and penalties
charges_mtm	d16.2	Scheduled obligations (defined in the Clearing Session) for payment of fees and penalties
fee_fut_mtm	d26.2	Scheduled obligation (defined in the Clearing Session) for the amount of exchange and clearing fees for anonymous futures trades
fee_opt_mtm	d26.2	Scheduled obligation (defined in the Clearing Session) for the amount of exchange and clearing fees for anonymous options trades

4.17.1.2. Table money_clearing_sa: Balances by Settlement Account

Table 95. Fields of table money_clearing_sa

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
settlement_account	c12	Settlement Account
amount_beg	d26.2	Actual balance of funds at the start of the trading day
vm	d26.2	Actually calculated (in Settlement Session) variation margin, including variation margin for marginable options

Field	Type	Description
premium	d26.2	Actually calculated (in Settlement Session) premium for options received/paid in the clearing session, converted into rubles. Includes the financial result from option exercise
fee_fut	d26.2	Actually calculated (in Settlement Session) amount of exchange and clearing fees for anonymous trades on futures, included in charges
fee_opt	d26.2	Actually calculated (in Settlement Session) amount of exchange and clearing fees for anonymous trades on options, included in charges
go	d26.2	Total initial margin on futures and options
amount_end	d26.2	Current balance of funds at the time of balance formation during the Clearing Session or Settlement Session
free	d26.2	Available funds
blocked_tax	d26.2	Funds blocked for tax payments.
date_money_clearing	t	Date and time of balance formation during the Clearing Session or Settlement Session
settl_datetime	t	Scheduled settlement date for obligations
in_out_netto	d26.2	Current net balance of deposits/withdrawals
vm_mtm	d26.2	Scheduled obligation (defined in the Clearing Session) for variation margin, including variation margin for marginable options
premium_mtm	d26.2	Scheduled obligation (defined in the Clearing Session) for option premiums, including the financial result from option exercise
charges	d16.2	Actually calculated (in the Settlement Session) obligations for payment of fees and penalties
charges_mtm	d16.2	Scheduled obligations (defined in the Clearing Session) for payment of fees and penalties
fee_fut_mtm	d26.2	Scheduled obligation (defined in the Clearing Session) for the amount of exchange and clearing fees for anonymous futures trades
fee_opt_mtm	d26.2	Scheduled obligation (defined in the Clearing Session) for the amount of exchange and clearing fees for anonymous options trades

4.17.1.3. Table clr_rate: Currency and Index rates

Table 96. Fields of table clr_rate

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
rate	d16.5	Index value
moment	t	Date and time value was fixed
signs	i1	Sign, that corresponds to the current value
sess_id	i4	Trading day ID
rate_id	i4	Rate ID

4.17.1.4. Table fut_pos: Positional state in futures as a result of clearing session

Table 97. Fields of table fut_pos

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
isin_id	i4	Instrument unique ID
sess_id	i4	Trading day ID
isin	c25	Symbol code of the instrument
client_code	c7	Client code
account_type	i1	Account type (0 - CF; 1 - BF; 2 - client).
xpos_beg	i8	Position on trading session start

Field	Type	Description
xpos_end	i8	Position on trading session end
vm	d16.2	Total variation margin at clearing time. Accrued calculated amount.
fee	d16.2	Total fee. Accrued calculated amount.
accum_go	d16.2	Accumulated Collateral Deposit. Accrued calculated amount.
fee_ex	d16.2	Exchange fee. Accrued calculated amount.
fee_cc	d16.2	Clearing fee. Accrued calculated amount.
charge_exec	d16.2	Execution fee. Accrued calculated amount.
sbor_nosys	d16.2	Sum total of the exchange fee / fee of the settlement organization and the commission fee of the Clearing Center for OTC-trades (RUB). Accrued calculated amount.
pos_enforcement_exec	i8	Number of positions forcedly executed for exiting perpetual futures
pay_enforcement_exec	d16.2	Payment amount (with sign) for forced execution of perpetual futures. Accrued calculated amount.
xpos_exec	i8	Number of executed positions.
vm_settl_price	d16.2	Part of total variation margin related to the change in settlement price, as calculated after the clearing session for the client/firm and instrument.
vm_swap_rate	d16.2	Part of total variation margin related to funding (for perpetual futures), as calculated after the clearing session for the client/firm and instrument.
vm_index_div	d16.2	Part of total variation margin related to dividend adjustment (for perpetual futures on index or share), as calculated after the clearing session for the client/firm and instrument.
date_money_clearing	t	Clearing date
settl_datetime	t	Settlement date

4.17.1.5. Table opt_pos: Positional state in options as a result of clearing session

Table 98. Fields of table opt_pos

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
isin_id	i4	Instrument unique ID
sess_id	i4	Trading day ID
isin	c25	Symbol code of the instrument
client_code	c7	Client code
account_type	i1	Account type (0 - CF; 1 - BF; 2 - client).
xpos_beg	i8	Position on trading session start
xpos_end	i8	Position on trading session end
vm	d16.2	Total VM after the clearing session per client/firm and instrument. Accrued calculated amount.
fee	d16.2	Total fee of the client/firm and instrument. Accrued calculated amount.
fee_ex	d16.2	Exchange fee. Accrued calculated amount.
fee_cc	d16.2	Clearing fee. Accrued calculated amount.
charge_exec	d16.2	Execution fee. Accrued calculated amount.
sbor_nosys	d16.2	Sum total of the exchange fee / fee of the settlement organization and the commission fee of the Clearing Center for OTC-trades (RUB). Accrued calculated amount.
premium	d26.2	Ruble premium under the option received/withdrawn at clearing session. Include the financial result of exercising the position. Accrued calculated amount.
premium_in_settl_currency	d26.2	Currency premium under the option received/withdrawn at clearing session. Include the financial result of exercising the position. Accrued calculated amount.
xpos_exec	i8	Number of executed positions.

Field	Type	Description
date_money_clearing	t	Clearing date
settl_datetime	t	Settlement date

4.17.1.6. Table fut_sess_settl: Futures settlement prices

Table 99. Fields of table fut_sess_settl

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 day ID
date_clr	t	Clearing date
isin	c25	Symbol code of the instrument
isin_id	i4	Instrument unique ID
settl_price	d16.5	Settlement price
swap_rate	d16.5	Funding when the contract price deviates from the underlying asset price is greater than the level specified on the contract (for perpetual futures)
index_div	d18.4	Dividend amendment for perpetual futures on the index or the stock-equities
exp_enforcement_pay	d16.2	Payment amount (in rubles) per contract for settlement of perpetual futures positions, in case of forced execution, from the party initiating the exit to the party forcibly closed out
step_price	d16.5	The cost of the clearing session price step
step_price_curr	d16.5	The cost of the clearing session price step in currency
swap_rate_curr	d16.5	Funding when the contract price deviates from the underlying asset price beyond the specified level on the contract (for perpetual futures). In points.

4.17.1.7. Table opt_sess_settl: Options settlement prices

Table 100. Fields of table opt_sess_settl

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 day ID
date_clr	t	Clearing date
isin	c25	Symbol code of the instrument
isin_id	i4	Instrument ID number
volat	d16.5	Option’s volatility
theor_price	d16.5	Option’s theoretical price
step_price	d16.5	The cost of the clearing session price step
step_price_curr	d16.5	The cost of the clearing session price step in currency

4.17.1.8. Table pledge_details: Pledgs details table

Table 101. Fields of table pledge_details

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
pledge_name	c10	Foreign currency/security code
xamount_beg	d26.2	Foreign currencies/securities amount at trading day opening
xpay	d26.2	Amount of foreign currencies/securities deposited or withdrawn, in units

Field	Type	Description
xamount	d26.2	Current amount of foreign currencies/securities
rate	d16.5	Assessed value of foreign currency/security unit (in Russian roubles)
xamount_beg_money	d26.2	Foreign currency/securities amount at trading day opening (in Russian rubles)
xpay_money	d26.2	Amount of foreign currencies/securities deposited or withdrawn, in units (in Russian roubles)
xamount_money	d26.2	Current amount of foreign currencies/securities (in Russian rubles)

Notes:

- Field 'amount_money' - Current amount of foreign currencies/securities (in Russian rubles) (calculated as 'amount' * 'rate')
- Field 'amount_beg_money' - Foreign currencies/securities amount at trading day opening (in Russian rubles) (in Russian roubles) (calculated as 'amount_beg' * 'rate')
- Field 'pay_money' - Amount of foreign currencies/securities deposited or withdrawn, in units (in Russian roubles) (calculated as 'pay' * 'rate')

4.17.1.9. Table fut_pos_sa: Positional state of SA on futures as a result of clearing session

Table 102. Fields of table fut_pos_sa

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
isin_id	i4	Instrument unique ID
sess_id	i4	Trading day ID
isin	c25	Symbol code of the instrument
settlement_account	c12	Settlement Account
xpos_beg	i8	Position on trading session start
xpos_end	i8	Position on trading session end
vm	d26.2	Total variation margin at clearing time. Accrued calculated amount.
fee	d26.2	Total fee. Accrued calculated amount.
fee_ex	d26.2	Exchange fee. Accrued calculated amount.
fee_cc	d26.2	Clearing fee. Accrued calculated amount.
charge_exec	d16.2	Execution fee. Accrued calculated amount.
sbor_nosys	d16.2	Sum total of the exchange fee / fee of the settlement organization and the commission fee of the Clearing Center for OTC-trades (RUB). Accrued calculated amount.
xpos_exec	i8	Number of executed positions.
vm_settl_price	d16.2	Part of total variation margin related to the change in settlement price, as calculated after the clearing session for the client/firm and instrument.
vm_swap_rate	d16.2	Part of total variation margin related to funding (for perpetual futures), as calculated after the clearing session for the client/firm and instrument.
vm_index_div	d16.2	Part of total variation margin related to dividend adjustment (for perpetual futures on index or share), as calculated after the clearing session for the client/firm and instrument.
date_money_clearing	t	Clearing date
settl_datetime	t	Settlement date

4.17.1.10. Table opt_pos_sa: Positional state of SA on options as a result of clearing session

Table 103. Fields of table opt_pos_sa

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

Field	Type	Description
isin_id	i4	Instrument unique ID
sess_id	i4	Trading day ID
isin	c25	Symbol code of the instrument
settlement_account	c12	Settlement Account
xpos_beg	i8	Position on trading session start
xpos_end	i8	Position on trading session end
vm	d26.2	Total VM after the clearing session per client/firm and instrument. Accrued calculated amount.
fee	d26.2	Total fee of the client/firm and instrument. Accrued calculated amount.
fee_ex	d26.2	Exchange fee. Accrued calculated amount.
fee_cc	d26.2	Clearing fee. Accrued calculated amount.
premium	d26.2	Ruble premium under the option received/withdrawn at clearing session. Include the financial result of exercising the position. Accrued calculated amount.
premium_in_settl_currency	d26.2	Currency premium under the option received/withdrawn at clearing session. Include the financial result of exercising the position. Accrued calculated amount.
charge_exec	d16.2	Execution fee. Accrued calculated amount.
sbor_nosys	d16.2	Sum total of the exchange fee / fee of the settlement organization and the commission fee of the Clearing Center for OTC-trades (RUB). Accrued calculated amount.
xpos_exec	i8	Number of executed positions.
date_money_clearing	t	Clearing date
settl_datetime	t	Settlement date

4.17.1.11. Table option_series_settl: Settlement prices for option series

Table 104. Fields of table option_series_settl

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 day ID
date_clr	t	Clearing date
isin	c25	Symbol code of the instrument
isin_id	i4	Instrument ID number
underlying_price	d16.5	The current spot price of the instrument, or on expiration date, the price of the underlying asset according to which the options will be exercised (see the relevant contract specification for details)
lot_coefficient	i4	Coefficient indicating the volume of the underlying asset in the contract quote and strikes of option series
step_price	d16.5	The cost of the clearing session price step
step_price_curr	d16.5	The cost of the clearing session price step in currency

4.17.1.12. Table sys_events: table of events

Table 105. 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	Unique ID of the event
sess_id	i4	Trading day ID
event_type	i4	Type of the event

Field	Type	Description
message	c64	Description of the event
server_time	t	Event time

Notes:

- Possible types of events:

  event_type = 3
  message = "clearing_data_ready"
  Data are ready after the clearing session

4.18. Stream RTS_INDEX_REPL - online indices (Type=R)

4.18.1. Data scheme

Tables:

- rts_index - Indices

4.18.1.1. Table rts_index: Indices

The table contains data about Stock Exchange Indices values.

Table 106. Fields of table rts_index

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
name	c25	Index ID
moment	t	Time of the last update
usd_rate	d10.4	USD rate for indices which include both RUB and USD contract prices
cap	d18.4	Index capitalization
volume	d18.4	Volume of trades that compose index value
value_highprec	d18.6	Index value (high-precision value, 6 decimal places)
prev_close_value_highprec	d18.6	Close value (high-precision value, 6 decimal places)
open_value_highprec	d18.6	Open value (high-precision value, 6 decimal places)
max_value_highprec	d18.6	Max value (high-precision value, 6 decimal places)
min_value_highprec	d18.6	Min value (high-precision value, 6 decimal places)

4.19. Stream FORTS_VM_REPL - Variation margin and premium (Type=I)

4.19.1. Data scheme

Tables:

- fut_vm - Variation margin on futures by positions of clients
- opt_vm - Variation margin and premium on options in the context of client positions
- fut_vm_sa - Variation margin on futures in the context of SA positions
- opt_vm_sa - Variation margin and premium on options in the context of SA positions
- sys_events - Table of events

4.19.1.1. Table fut_vm: Variation margin on futures by positions of clients

Table 107. Fields of table fut_vm

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
isin_id	i4	Instrument unique ID

Field	Type	Description
sess_id	i4	Trading day ID
client_code	c7	Client code
vm	d16.5	The accumulated variation margin on trades calculated according to the current market price
vm_at_settl	d16.5	Indicative variation margin considering the fixed settlement price for the upcoming clearing. Before the settlement price is fixed, the current market price is used in the calculation
swap_rate	d16.5	Indicative funding on current client position (for perpetual futures)
swap_rate_curr	d16.5	Indicative funding on current client position (for perpetual futures). In points.
index_div	d18.4	Indicative dividend adjustment on current client position (for perpetual futures on an index or share)

Notes:

- Variation margin reflects (without considering the variability of the currency component, which is recorded in clearing) the realized financial result "if the position is closed now".
- Variation margin, which reflects the value "if clearing were to occur right now" (without considering discrepancies between fixing prices of underlying asset and market prices of perpetual futures in individual cases), requires following components to be summed:

```
VM = vm – swap_rate + index_div
```

- After the end of the trading day, its data is removed from the table during the assignment of the next trading day and generation of initial data for it. Thus, at any given time, the table transmits data for only one trading day.

4.19.1.2. Table opt_vm: Variation margin and premium on options in the context of client positions

Table 108. Fields of table opt_vm

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
isin_id	i4	Instrument unique ID
sess_id	i4	Trading day ID
client_code	c7	Client code
vm	d16.5	The accumulated variation margin on futures-style options trades calculated according to the current market price
vm_at_settl	d16.5	Indicative variation margin considering the fixed settlement price for the upcoming clearing. Before the settlement price is fixed, the current market price is used in the calculation
premium	d26.2	Accumulated premium under trades in options in RUB
premium_in_settl_currency	d26.2	Accumulated premium under trades in options in settlement currency

Notes:

- After the end of the trading day, its data is removed from the table during the assignment of the next trading day and generation of initial data for it. Thus, at any given time, the table transmits data for only one trading day.

4.19.1.3. Table fut_vm_sa: Variation margin on futures in the context of SA positions

Table 109. Fields of table fut_vm_sa

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
isin_id	i4	Instrument unique ID
sess_id	i4	Trading day ID
settlement_account	c12	Settlement Account
vm	d26.2	The accumulated variation margin on trades calculated according to the current market price

Field	Type	Description
vm_at_settl	d26.2	Indicative variation margin considering the fixed settlement price for the upcoming clearing. Before the settlement price is fixed, the current market price is used in the calculation
swap_rate	d16.5	Indicative funding for the current position of the SA (for perpetual futures)
swap_rate_curr	d16.5	Indicative funding on current client position (for perpetual futures). In points.
index_div	d18.4	Indicative dividend adjustment for the current position of the SA (for perpetual futures on an index or share)

Notes:

- Variation margin reflects (without considering the variability of the currency component, which is recorded in clearing) the realized financial result "if the position is closed now".
- Variation margin, which reflects the value "if clearing were to occur right now" (without considering discrepancies between fixing prices of underlying asset and market prices of perpetual futures in individual cases), requires following components to be summed:

```
VM = vm – swap_rate + index_div
```

- After the end of the trading day, its data is removed from the table during the assignment of the next trading day and generation of initial data for it. Thus, at any given time, the table transmits data for only one trading day.

4.19.1.4. Table opt_vm_sa: Variation margin for options

Table 110. Fields of table opt_vm_sa

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
isin_id	i4	Instrument unique ID
sess_id	i4	Trading day ID
settlement_account	c12	Settlement Account
vm	d26.2	The accumulated variation margin on futures-style options trades calculated according to the current market price
vm_at_settl	d26.2	Indicative variation margin considering the fixed settlement price for the upcoming clearing. Before the settlement price is fixed, the current market price is used in the calculation
premium	d26.2	Accumulated premium under trades in options in RUB
premium_in_settl_currency	d26.2	Accumulated premium under trades in options in settlement currency

Notes:

- After the end of the trading day, its data is removed from the table during the assignment of the next trading day and generation of initial data for it. Thus, at any given time, the table transmits data for only one trading day.

4.19.1.5. Table sys_events: Table of events

Table 111. 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	Unique ID of the event
sess_id	i4	Trading day ID
event_type	i4	Type of the event
message	c64	Description of the event
server_time	t	Event time

Notes:

- Possible types of events

  event_type = 1

message = "session_data_ready"
All data from the clearing system have been loaded into the trading system

event_type = 5
message = "clearing_started"
Start of the clearing session

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

event_type = 17
message = "clearing_trades_inserted"
Clearing trades have been generated

event_type = 18
message = "extension_of_limits_started_for_core"
Limits have been extended

event_type = 36
message = "start_of_calendar_date"
Beginning of the calendar day

event_type = 37
message = "settlement_session_started"
Beginning of the settlement session

event_type = 38
message = "settlement_session_finished"
End of the settlement session

event_type = 47
message = "settlpricecalc_finished"
Settlement price has been fixed

4.20. Stream FORTS_VOLAT_REPL - online volatility information (Type=I)

4.20.1. Data scheme

Tables:

- volat - Volatility
- sys_events - Table of events

4.20.1.1. Table volat: Volatility

Current volatility values and theoretical prices for options. Is sent in form of time snaps at 10 minute intervals.

Table 112. Fields of table volat

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
isin_id	i4	Instrument unique ID
sess_id	i4	Trading day ID
volat	d16.5	Option’s volatility
theor_price	d16.5	Option’s theoretical price
theor_price_limit	d16.5	Theoretical option price with limits
up_prem	d16.5	Upper limit for option price
down_prem	d16.5	Lower limit for option price
up_prem_weekend	d16.5	Upper price limit for option premium during the Additional Weekend Trading Session. On a regular trading day (not during Additional Weekend Trading Session), the field contains the value '-1'.
down_prem_weekend	d16.5	Lower price limit for option premium during the Additional Weekend Trading Session. On a regular trading day (not during Additional Weekend Trading Session), the field contains the value '-1'.
option_model	i1	Options pricing model

Notes:

- Field option_model can take the following values:

  0 Black-Scholes model.

  1 Bachelier model.

- After the end of the trading day, its data is removed from the table during the assignment of the next trading day and generation of initial data for it. Thus, at any given time, the table transmits data for only one trading day.

4.20.1.2. Table sys_events: Table of events

Table 113. 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	Unique ID of the event
sess_id	i4	Trading day ID
event_type	i4	Type of the event
message	c64	Description of the event
server_time	t	Event time

Notes:

- Possible types of events

  event_type = 1
  message = "session_data_ready"
  All data from the clearing system have been loaded into the trading system

  event_type = 5
  message = "clearing_started"
  Start of the clearing session

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

  event_type = 17
  message = "clearing_trades_inserted"
  Clearing trades have been generated

  event_type = 18
  message = "extension_of_limits_started_for_core"
  Limits have been extended

  event_type = 36
  message = "start_of_calendar_date"
  Beginning of the calendar day

  event_type = 37
  message = "settlement_session_started"
  Beginning of the settlement session

  event_type = 38
  message = "settlement_session_finished"
  End of the settlement session

  event_type = 47
  message = "settlpricecalc_finished"
  Settlement price has been fixed

4.21. Stream FORTS_RISKINFOBLACK_REPL - Risk parameters for Black-Scholes model (Type=I)

4.21.1. Data scheme

Tables:

- volat_coeff - Risk parameters for Black-Scholes model

4.21.1.1. Table volat_coeff: Risk parameters for Black-Scholes model

Table 114. Fields of table volat_coeff

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
option_series_id	i4	Series of Options ID
a	f	Volatility curve calculation parameter for Black-Scholes model.
b	f	Volatility curve calculation parameter for Black-Scholes model.
c	f	Volatility curve calculation parameter for Black-Scholes model.
d	f	Volatility curve calculation parameter for Black-Scholes model.
e	f	Volatility curve calculation parameter for Black-Scholes model.
s	f	Volatility curve calculation parameter for Black-Scholes model.

4.22. Stream FORTS_RISKINFOBACH_REPL - Risk parameters for Bachelier model (Type=I)

4.22.1. Data scheme

Tables:

- volat_coeff - Risk parameters for Bachelier model

4.22.1.1. Table volat_coeff: Risk parameters for Bachelier model

Table 115. Fields of table volat_coeff

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
option_series_id	i4	Series of Options ID
a	f	Volatility curve calculation parameter for Bachelier model.
b	f	Volatility curve calculation parameter for Bachelier model.
c	f	Volatility curve calculation parameter for Bachelier model.
d	f	Volatility curve calculation parameter for Bachelier model.
e	f	Volatility curve calculation parameter for Bachelier model.
s	f	Volatility curve calculation parameter for Bachelier model.
m	f	Volatility curve calculation parameter for Bachelier model.

4.23. Stream FORTS_INFO_REPL - additional reference information (Type=R)

4.23.1. Data scheme

Tables:

- currency_params - FX parameters
- base_contracts_params - Base contracts parameters
- futures_params - Futures parameters
- option_series_params - Series of options settings.
- options_params - Options parameters
- investor - Clients directory

- dealer - Companies directory
- multileg_dictionary - Multileg instruments dictionary
- common_params - Collateral calculation parameters
- brokers_base_contracts_params - Individual coefficient of IM in the context of the underlying contract and BF
- sys_events - Table of events

4.23.1.1. Table currency_params: FX parameters

Table 116. Fields of table currency_params

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
currency_id	i4	FX ID taken from directory 'rates' of stream 'FORTS_REFDATA_REPL'
radius	f	FX price fluctuation radius (specified during the last clearing session)

4.23.1.2. Table base_contracts_params: Base contracts parameters

Table 117. Fields of table base_contracts_params

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	Code of the underlying contract.
code_mcs	c25	Intercontract spread ID
volat_num	i1	Number of volatility curves
has_options	i1	Option on futures for given underlying asset. 0 - none, 1 - yes.
percent_rate	d16.5	Variation margin rate for interest rate futures
somc	f	Collateral rate for uncovered sells
msp_type	i1	Price pitch value type. 0 - fixed, 1 - taken from FX indicator value
currency_id	i4	FX ID taken from directory 'rates' of stream 'FORTS_REFDATA_REPL'
spot_price	f	The settlement price of the underlying asset, expressed in rubles, as determined by the results of the clearing session.
mr1	f	Market risk rate value
mr2	f	Market risk rate value (Concentration Limit 1)
mr3	f	Market risk rate value (Concentration Limit 2)
mr1_lrc	f	Market risk rate for low-risk client
mr2_lrc	f	Market risk rate (Concentration limit 1) for low-risk client
mr3_lrc	f	Market risk rate (Concentration limit 2) for low-risk client
mr1_mrc	f	Market risk rate for medium-risk client
mr2_mrc	f	Market risk rate (Concentration limit 1) for medium-risk client
mr3_mrc	f	Market risk rate (Concentration limit 2) for medium-risk client
mr1_hrc	f	Market risk rate for high-risk client
mr2_hrc	f	Market risk rate (Concentration limit 1) for high-risk client
mr3_hrc	f	Market risk rate (Concentration limit 2) for high-risk client
mr1_erc	f	Market risk rate for exceptional-risk client
mr2_erc	f	Market risk rate (Concentration limit 1) for exceptional-risk client
mr3_erc	f	Market risk rate (Concentration limit 2) for exceptional-risk client
lk1	i8	Amount of underlying asset, in units (Concentration Limit 1)
lk2	i8	Amount of underlying asset, in units (Concentration Limit 2)
risk_points_n	i4	Number of contract price fluctuation scenarios near risk calculation point.

Field	Type	Description
window_size	f	Coefficient of determination smoothing window size for cross-contract spread margining
option_model	i1	Options pricing model
asset_class	i4	Underlying asset type.
cf_risk	f	Risk rate pertaining to changes in projected cash flows.
signs	i4	Flags field

Notes:

- Field 'option_model' can take the following values:

  0 Black-Scholes model.

  1 Bachelier model.

- Field 'asset_class' may contain the following values:

  1 Share

  2 Currency

  3 Bond

  4 Index

  5 Commodity

  6 Interest rate

- Field signs is a bit mask and defines the following values:

  0x1 Scheduled execution or execution in clearing session: 0 - clearing session; 1 - scheduled execution

  0x2 Foreign instrument: 0 - not foreign; 1 - foreign

  0x4 TAS futures: 0 - not TAS futures; 1 - TAS futures (*Not available in the current version*)

  0x8 For qualified investors

4.23.1.3. Table futures_params: Futures parameters

Table 118. Fields of table futures_params

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
isin	c25	Symbol code of the instrument
isin_id	i4	Instrument unique ID
base_contract_code	c25	Code of the underlying contract.
risk_range_center	d16.5	Risk calculation center
spread_aspect	i1	Flag of making up futures spread
step_price	f	Value of the minimum price increment
exp_date	t	Date of expiration
settlement_price	d16.5	Settlement price after the last clearing session
min_step	f	Minimal price increment
lot	i4	Number of underlying asset in instrument, in units. The field is deprecated and will be removed in version 10.3.
xlot_volume	f	Number of underlying asset in instrument, in units. Fractional values are possible.
attribute	i4	Bit flags defining futures type
interest_rate_risk_up	f	Interest risk variable rate on rate up scenario
interest_rate_risk_down	f	Interest risk variable rate on rate down scenario

Field	Type	Description
time_to_expiration	f	Time before instrument expiration, in fraction of year
normalized_spot	f	Theoretical price value of underlying asset on spot market, in points, reduced to dimension of the primary one
mr_addon_up	f	Up-addition for NormalizedSpot to control initial margin on futures level, specified in NormalizedSpot fractions.
mr_addon_down	f	Down-addition for NormalizedSpot to control initial margin on futures level, specified in NormalizedSpot fractions.
enforce_ims_half_netting	i1	Flag - consider the risks of intermonth spread according to the "half-netto" rule: "1" - yes; "0" - no.
tas_base_fut_isin_id	i4	The UA-futures identifier (isin_id) for the TAS-futures. (*Not available in the current version*)

Notes:

- Field spread_aspect can take the following values:

  0 It is not included in spread

  2 It is included into calendar spread

- Field 'attribute' can take the following values:

  0 Ordinary futures

  0x100000 TAS futures (*Not available in the current version*)

  0x800000 Collateral

4.23.1.4. Table option_series_params: Parameters for series of options

Table 119. Fields of table option_series_params

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
small_name	c25	Symbol code
isin_base	c25	Real futures ID
a_black	f	Volatility curve calculation parameter for Black-Scholes model.
b_black	f	Volatility curve calculation parameter for Black-Scholes model.
c_black	f	Volatility curve calculation parameter for Black-Scholes model.
d_black	f	Volatility curve calculation parameter for Black-Scholes model.
e_black	f	Volatility curve calculation parameter for Black-Scholes model.
expiration_date	t	Expiration period closing date
use_null_volat	i1	Zero volatility
s_black	f	Volatility curve calculation parameter for Black-Scholes model.
strike_step	f	Strike pitch
exp_clearings_sa	i4	Number of clearing sessions for SA, before commencing to block the collateral, preliminary calculated for the whole SA according to the given expiration model. Specified by NCC.
exp_clearings_bf	i4	Number of clearing sessions for BF, before commencing to block the collateral, preliminary calculated for the whole BF according to the given expiration model. Specified by NCC.
exp_clearings_cc	i4	Number of clearing sessions, before 'exp_weight' applies for BF clients. Specified by NCC.
volatility_risk	f	Volatility risk range rate
volatility_risk_mismatch	f	Volatility risk variable rate for different maturity dates of series of options
time_to_expiration	f	Time before expiration, in fraction of year
option_series_id	i4	Series of Options ID
underlying_id	i4	Futures ID

Field	Type	Description
a_bach	f	Volatility curve calculation parameter for Bachelier model.
b_bach	f	Volatility curve calculation parameter for Bachelier model.
c_bach	f	Volatility curve calculation parameter for Bachelier model.
d_bach	f	Volatility curve calculation parameter for Bachelier model.
e_bach	f	Volatility curve calculation parameter for Bachelier model.
s_bach	f	Volatility curve calculation parameter for Bachelier model.
m_bach	f	Volatility curve calculation parameter for Bachelier model.
margin_style	i4	Option margin method. 0 - Futures-style option; 1 - Equity-style option.
settlement_type	i4	Option type. 0 - Cash-settled; 1 - Deliverable.
exercise_style	i4	Exercise style of option. 0 - American; 1 - European.
sub_risk	i1	Risk treating by sub-points: 1 - Risk treatment by sub-points is enabled, 0 - Disabled.
spread_aspect	i1	Attribute of option series position: 0 - Outside the spread, 2 - Within spread.
enforce_half_netting	i1	Attribute of forced activation of the half-netting rule when cross-margining option series with a collateral instrument, which have spread_aspect=2.
min_step	f	The minimum price movement.
step_price	f	The tick value.
lot_coefficient	i4	Coefficient indicating the volume of the underlying asset in the contract quote and strikes of option series.
r	f	Risk-free interest rate.
interest_rate_risk_up	f	Interest rate mismatch rate in the scenario of rate movement ‘r’ upward.
interest_rate_risk_down	f	Interest rate mismatch rate in the downward rate movement scenario ‘r’.
fixed_spot_discount	f	Sum of discounted values of declared cash flows. CF calculated on the next day T+1
projected_spot_discount	f	Sum of discounted values of forecasted cash flows.
r2	f	Risk-free interest rate FX2 of the currency pair FX2/FX1 (for premium options on the currency); dividend yield rate ‘q’ (for premium options on the index).
interest_rate2_risk_up	f	Interest rate mismatch rate in the scenario of rate movement ‘r2’ upward.
interest_rate2_risk_down	f	Interest rate mismatch rate in the downward rate movement scenario ‘r2’.
lot_volume	i4	Lot, i.e. number of units of the underlying asset in the instrument
fixed_spot_discount_clr	f	Sum of discounted values of declared cash flows. CF calculated at the beginning of the trading day T

4.23.1.5. Table options_params: Options parameters

Table 120. Fields of table options_params

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
isin	c25	Symbol code of the instrument
isin_id	i4	Instrument unique ID
isin_base	c25	Futures ID
strike	d16.5	Option's strike
opt_type	i1	Option's type: 1 - PUT, 2 - CALL
settl_price	d16.5	Settlement price

4.23.1.6. Table investor: Clients directory

Table 121. Fields of table investor

Field	Type	Description
replID	i8	Service field of the replication subsystem

Field	Type	Description
replRev	i8	Service field of the replication subsystem
replAct	i8	Service field of the replication subsystem
client_code	c7	Client code
calendar_spread_margin_type	i1	Margin type for client calendar spread, current value: • 3 - half nett • 4 - nett
num_clr_2delivery	i4	Number of clearing sessions before expiration to start expiration scenarios calculation.
exp_weight	d3.2	Expiration scenario weight, in total collateral.
coeff_im	d16.5	Total collateral ratio value.
no_fut_discount	i1	Discount on futures: • 1 - Discount prohibited • 0 - Discount allowed
short_option_minimum_charge_ratio	d5.3	Individual coefficient of SOMC scenario weight.
ics_margin_type	i1	Margin type for cross-contract spreads: • 3 - half nett • 4 - nett
is_blank	i4	The account-blank flag for online registration
client_risk_level	i4	Client risk level category.
xstatus	i8	Client's flags, extended.
not_use_nov	i4	Status of using the special mode for accounting of premium option (without Initial margin and NOV, with a prohibition on negative positions in Premium option)

Notes:

- Field client_risk_level can take the following values:
  - 0 - Not specified (default value)
  - 1 - Low-risk client
  - 2 - Medium-risk client
  - 3 - High-risk client
  - 4 - Exceptional-risk client
- Xstatus field is a bit mask:
  - 0x1 - Trust Management
  - 0x2 - Separated
  - 0x4 - Brokerage Firm (Trust Management type)
  - 0x80 - Private entity
  - 0x100 - Legal entity
  - 0x200 - Non-resident
  - 0x2000 - Individual investment account
  - 0x4000 - Flag for allowing cross-trades for negotiated orders. 1 - cross-trades allowed; 0 - cross-trades prohibited
  - 0x8000 - Stateless person
  - 0x20000 - Own
  - 0x40000 - Client
  - 0x80000 - Special BF

- 0x10000000 - Additional own account
- 0x10000000000 - Qualified investor
- 0x40000000000 - Cancel a passive order in a cross-trade

- The not_use_nov field can take the following values:
  - -1 - "Same as for BF" (inherited from dealer.not_use_nov_client_default)
  - 0 - Standard method for accounting of options
  - 1 - Special mode (without Initial margin and NOV, with a prohibition on negative positions in Premium option)

4.23.1.7. Table dealer: Companies directory

Table 122. Fields of table dealer

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
margin_type	i1	Margin type, according to BF's sections, current value: • 3 - half nett • 4 - nett
calendar_spread_margin_type	i1	Margin type for calendar spreads, for BF portfolio, current value: • 3 - half nett • 4 - nett
check_limit_on_withdrawal	i1	Verify collateral sufficiency, for BF, upon funds depositing/withdrawal, current value: • 1 - Verify • 0 - Do not verify
limit_tied_money	i1	BF trading limit accordance with the BF's total funds amount (all sections): • 1 - maintain accordance • 0 - virtual (independent) limit. The value may change according to the profit/loss values only, resulting from the clearing session. Current value.
num_clr_2delivery	i4	Number of clearing sessions before expiration to start BF expiration scenarios calculation. Current value.
exp_weight	d3.2	Expiration scenario weight for BF, in total collateral. Current value.
coeff_im	d16.5	Total collateral ratio value, for BF. Current value.
no_fut_discount	i1	Discount on futures for BF portfolio, current value: • 1 - Discount prohibited • 0 - Discount allowed
num_clr_2delivery_client_default	i4	Number of clearing sessions before expiration to start clients expiration scenarios calculation (default value). Current value.
exp_weight_client_default	d3.2	Expiration scenario weight for clients, in total collateral (default value). Current value.
no_fut_discount_client_default	i1	Discount on futures for BF's clients, default value: • 1 - DIscount prohibited • 0 - DIscount allowed Current value.
short_option_minimum_charge_ratio	d5.3	Individual coefficient of SOMC scenario weight.
ics_margin_type	i1	Margin type for cross-contract spreads:

Field	Type	Description
		• 3 - half nett • 4 - nett
order_allowed_in_morning_session	i1	Access to trading during the morning trading session.
order_allowed_in_weekend_session	i1	Access to trading during an additional weekend trading session.
not_use_nov_client_default	i1	Default status of using the special mode for accounting of premium option for clients of the Brokerage Firm.

Notes:

- Field order_allowed_in_morning_session can take the following values:
  - 0 Access to trading during the morning trading session is limited. Trading operations are prohibited, except for orders cancellation operations.
  - 1 Access to trading during the morning trading session is allowed.
- Field order_allowed_in_weekend_session can take the following values:
  - 0 Access to trading during an additional weekend trading session is limited. Trading operations are prohibited, except for orders cancellation operations.
  - 1 Access to trading during an additional weekend trading session is allowed.
- The not_use_nov_client_default field can take the following values:
  - 0 Standard method for accounting of options.
  - 1 Special mode (without Initial margin and NOV, with a prohibition on negative positions in premium option).

4.23.1.8. Table multileg_dictionary: Multileg instruments dictionary

Table 123. Fields of table multileg_dictionary

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
isin_id	i4	Multileg instrument ID
isin_id_leg	i4	ID of the instrument which is a component of specified multileg instrument
leg_order_no	i1	Leg order in a multileg instrument. The default value is 0.

4.23.1.9. Table common_params: Collateral calculation parameters

Table 124. Fields of table common_params

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
common_rev	i4	Revision number - surrogate key
edge_coeff	f	Marginal risk factor ratio

4.23.1.10. Table brokers_base_contracts_params: Individual coefficient of IM in the context of the underlying contract and BF

Table 125. Fields of table brokers_base_contracts_params

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
broker_code	c4	Brokerage Firm code
base_contract_code	c25	Underlying contract code.

Field	Type	Description
coeff_im	d16.5	Coefficient of IM under the underlying contract for all clients of this BF. Current value.

4.23.1.11. Table sys_events: Table of events

Table 126. 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_type	i4	Type of the event
event_id	i8	Unique ID of the event
sess_id	i4	Trading day ID
message	c64	Description of the event
server_time	t	Event time

Notes:

- Possible types of events

  event_type = 1
  message = "session_data_ready"
  All data from the clearing system have been loaded into the trading system

  event_type = 5
  message = "clearing_started"
  Start of the clearing session

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

  event_type = 17
  message = "clearing_trades_inserted"
  Clearing trades have been generated

  event_type = 18
  message = "extension_of_limits_started_for_core"
  Limits have been extended

  event_type = 36
  message = "start_of_calendar_date"
  Beginning of the calendar day

  event_type = 37
  message = "settlement_session_started"
  Beginning of the settlement session

  event_type = 38
  message = "settlement_session_finished"
  End of the settlement session

  event_type = 47
  message = "settlpricecalc_finished"
  Settlement price has been fixed

4.24. Stream FORTS_TNPENALTY_REPL - information about Transaction fees (Type=I)

4.24.1. Data scheme

Tables:

- fee_tn - Detailed information on the number of incorrect transaction
- fee_all - Information on the number of points accrued
- heartbeat - Server times table

4.24.1.1. Table fee_tn: Detailed information on the number of incorrect transaction

Table 127. Fields of table fee_tn

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
time	i8	Time value in 'YYYYMMddhhmmssSSS' format
p2login	c64	Login
sess_id	i4	Trading day ID
tn_type	i4	Type of invalid transaction (AddOrder - 1, DelOrder - 2, MoveOrder - 3, DelUserOrders - 4)
err_code	i4	Error code
count	i4	Number of invalid transactions
num_orders	i4	Number of canceled orders

4.24.1.2. Table fee_all: Information on the number of points accrued

Table 128. Fields of table fee_all

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
time	i8	Time value in 'YYYYMMddhhmmssSSS' format
p2login	c64	Login
sess_id	i4	Trading day ID
points	i4	Number of points assessed for a second 'time' from
fee	d16.2	The total fee for invalid transactions by 'time' from the beginning of the trading day

4.24.1.3. Table heartbeat: Server times table

Records in this table are added periodically by the trading system's core. It can be used for synchronization purposes. The table is insert-only, no modifications or deletions occur during trading session.

Table 129. 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

4.25. Stream MOEX_RATES_REPL - online currency rates (Type=I)

4.25.1. Data scheme

Tables:

- curr_online - Currency rates values

4.25.1.1. Table curr_online: Currency rates values

Table 130. Fields of table curr_online

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
rate_id	i4	Currency rate ID

Field	Type	Description
value	d16.5	Current currency rate
moment	t	Currency rate calculation time

Notes:

- Field 'rate_id' can take the following values:

  1 USD/RUB

  50 USD/JPY

  51 USD/CHF

  52 JPY/RUB

  53 CHF/RUB

  56 GBP/RUB

  58 EUR/RUB

  62 USD/TRY

  63 CAD/RUB

  64 TRY/RUB

  66 CNY/RUB

  67 EUR/USD

  70 GBP/USD

  72 HKD/RUB

4.26. Stream FORTS_FORECASTIM_REPL - Risk forecast after limits extension (Type=I)

4.26.1. Data scheme

Tables:

- part_sa_forecast - Free funds for SA volume forecast
- sys_events - Table of events

4.26.1.1. Table part_sa_forecast: Free funds for SA volume forecast

Table 131. Fields of table part_sa_forecast

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
settlement_account	c12	Settlement account
money_free	d26.2	Funds available
MarketDataRev	i8	Revision number (field 'replRev' value) of the most recent data change (for streams transmitting orders and trades data) included into risk parameters forecast. Orders and trades with the 'replRev' revision value less than field 'MarketDataRev' value will be included into the forecast. Orders and trades with the 'replRev' revision value greater than field 'MarketDataRev' value will NOT be included into the forecast. For more information about field 'replRev' see section 3.3.1. Service replication fields.

4.26.1.2. Table sys_events: Table of events

Table 132. Fields of table sys_events

Field	Type	Description
replID	i8	Service field of the replication subsystem

Field	Type	Description
replRev	i8	Service field of the replication subsystem
replAct	i8	Service field of the replication subsystem
event_id	i8	Unique ID of the event
sess_id	i4	Trading day ID
event_type	i4	Type of the event
message	c64	Description of the event
server_time	t	Event time

Notes:

- Possible types of events

  event_type = 1
  message = "session_data_ready"
  All data from the clearing system have been loaded into the trading system

  event_type = 5
  message = "clearing_started"
  Start of the clearing session

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

  event_type = 17
  message = "clearing_trades_inserted"
  Clearing trades have been generated

  event_type = 18
  message = "extension_of_limits_started_for_core"
  Limits have been extended

  event_type = 36
  message = "start_of_calendar_date"
  Beginning of the calendar day

  event_type = 37
  message = "settlement_session_started"
  Beginning of the settlement session

  event_type = 38
  message = "settlement_session_finished"
  End of the settlement session

  event_type = 47
  message = "settlpricecalc_finished"
  Settlement price has been fixed

4.27. Stream FORTS_USER_REPL - Users (Type=R)

4.27.1. Data scheme

Tables:

- user - System users
- sma_master - SMA login binding to MASTER login
- sma_pre_trade_check - SMA login pre-trade verification settings.
- sys_events - Table of events

4.27.1.1. Table user: System users

Table 133. Fields of table user

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

Field	Type	Description
login	c20	Trading participant's login
start_date	t	Login start time
end_date	t	Login end time
client_code	c7	7-symbol client code
operation_mask	i4	Bitmask: • 2 - Limit open positions for BF • 8 - Limit BF on funds transfer. The setting is available to Clearing Firm operator or Trading Administrator only. • 16 - Money back. • 32 - Limit client positions. • 128 - Client restrictions. • 1024 - Orders-related restrictions for SMA logins.
lang	i2	Message language code
sma_flags	i4	Bitmask (see Notes below): • 1st bit - Cancel on Disconnect • 2nd bit - Cancel on DropCopy Disconnect • 3rd bit - SMA login.
sma_status	i4	Bitmask (see Notes below): • 1st bit - enable/disable trading transactions for the login. • 2nd bit - cancel/do not cancel orders when trading transactions are disabled for the login.
aspref	i4	Client ID. For orders added by SMA login - MASTER login ID.
user_level	i1	User login level: • 1 - CF • 2 - BF • 3 - Client
password_expiration_date	t	Password expiration date.

Notes:

- Field 'sma_flags' is bitmask:
  - 1st bit: 0 - Cancel on Disconnect is disabled for the login, 1 - Cancel on Disconnect is enabled for the login
  - 2nd bit: 0 - Cancel on Drop-Copy Disconnect is disabled for the login, 1 - Cancel on Drop-Copy Disconnect is enabled for the login
  - 3rd bit: 0 - SMA mode is disabled for the login, 1 - SMA mode is enabled for the login.
- Field 'sma_status' is bitmask::
  - 1st bit: 0 - trading transactions are enabled for the login, 1 - trading transactions are disabled for the login
  - 2nd bit: 0 - do not cancel orders when trading transactions are disabled for the login, 1 - cancel orders when trading transactions are disabled for the login.

4.27.1.2. Table sma_master: SMA login binding to MASTER login

The table contains information on how SMA login is binding to MASTER login.

Table 134. Fields of table sma_master

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

Field	Type	Description
sma_asp	c20	SMA login
sma_aspref	i4	SMA login ID
master_asp	c20	MASTER login
master_aspref	i4	MASTER login ID

4.27.1.3. Table sma_pre_trade_check: SMA login pre-trade verification settings

The table contains information on SMA login pre-trade verification settings.

Table 135. Fields of table sma_pre_trade_check

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
check_id	i8	Unique record ID.
sma_asp	c20	SMA login
sma_aspref	i4	SMA login ID
check_number	i1	Verification number (1 - 7).
base_contract_code	c25	Underlying asset code.
instrument_type	i1	Instrument type: • 0 - Futures • 1 - Option • 3 - Calendar spread
client_code_check	c7	Client code under verification.
value	d26.2	Verification number.

4.27.1.4. Table sys_events: Table of events

Table 136. 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
sess_id	i4	Trading day ID
event_id	i8	Unique ID of the event
event_type	i4	Type of the event
message	c64	Description of the event
server_time	t	Event time

Notes:

- Possible types of events

  event_type = 1
  message = "session_data_ready"
  All data from the clearing system have been loaded into the trading system

  event_type = 5
  message = "clearing_started"
  Start of the clearing session

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

  event_type = 17
  message = "clearing_trades_inserted"
  Clearing trades have been generated

event_type = 18
message = "extension_of_limits_started_for_core"
Limits have been extended

event_type = 36
message = "start_of_calendar_date"
Beginning of the calendar day

event_type = 37
message = "settlement_session_started"
Beginning of the settlement session

event_type = 38
message = "settlement_session_finished"
End of the settlement session

event_type = 47
message = "settlpricecalc_finished"
Settlement price has been fixed

4.28. Stream FORTS_REJECTEDORDERS_REPL - Register of orders rejected during the clearing (Type=R)

4.28.1. Data scheme

Tables:

- rejected_orders - Register of orders rejected during the clearing

4.28.1.1. Table rejected_orders: Register of orders rejected during the clearing

Table 137. Fields of table rejected_orders

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
order_id	i8	Order ID number
sess_id	i4	Trading day ID
moment	t	Order status changing time
isin_id	i4	Instrument unique ID
client_code	c7	Client code
dir	i1	Direction
xamount	i8	Volume, in units of the instrument
price	d16.5	Price
date_exp	t	Order's expiration date
id_ord1	i8	ID number of the first order
moment_reject	t	Time when the order was rejected
ret_code	i4	Return code of the re-entering procedure
ret_message	c255	Text of the message containing the reason for rejection of the order when it is re-entered
comment	c20	Trader's comment
login_from	c20	Login of the user who has entered the order
ext_id	i4	External ID number. It is added to orders, trades
opt_type	i4	Instrument type

4.29. Stream FORTS_RMT_REPL - Collateral without orders and current operational risk (Type=I)

4.29.1. Data scheme

Tables:

- rmt_im - Collateral without orders and current operational risk in the context of clients
- sys_events - Table of events

4.29.1.1. Table rmt_im: Collateral without orders and current operational risk in the context of clients

The table includes values of exchange collateral without orders and current operational risk, as well as Indicative variation margin, Net Option Value and premium, calculated by a specialized service and used by trading members for self-assessment of Risk Coverage Standard 1 and Risk Coverage Standard 2.

The service performs calculations at least once every 10 minutes based on the snapshot of current data, including:

- Risk parameters of instruments;
- Positions;
- Current market data (price, currency rate).

Each data snapshot is identified by its key (rev_heartbeat) and its timestamp (im_time). The table transmits only the latest actual calculation. Data snapshots used for calculations are archived for record-keeping and audit purposes.

Table 138. Fields of table rmt_im

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
rev_heartbeat	i8	Internal calculation identifier
sess_id	i4	Trading day ID
im_time	t	Calculation timestamp
client_code	c7	Client code
client_risk_level	i4	Client risk level category
rmt_im	d16.2	Exchange initial margin without orders, taking into account the current realized risk (considering the current price)
rmt_posrisk	d16.2	Current operational risk (an additional value, requested by members, not used in Risk Coverage Standard calculation, that serves a predictive function and answers the question of what collateral would be if clearing were to occur right now)
rmt_posrisk_at_settl	d16.2	Current operational risk considering the fixed settlement price for the upcoming clearing. Before the settlement price is fixed, the current market price is used in the calculation
rmt_vm	d26.2	Indicative variation margin considering the current indicative currency rate, Russian rubles (calculated similarly to the current indicative variation margin, including the variation margin on closed positions)
rmt_vm_at_settl	d26.2	Indicative variation margin considering the fixed settlement price for the upcoming clearing. Before the settlement price is fixed, the current market price is used in the calculation
rmt_nov	d26.2	Net Option Value, Russian rubles (current Net Option Value for a portfolio with 7-digit client code, considering the current rate and current option prices)
rmt_premium	d26.2	Accrued premium for premium options from trades, payable in the next clearing session, converted to RUB. Does not include the financial result of position execution

Notes:

- Field client_risk_level can take the following values:
  - 0 - Not specified (default value)
  - 1 - Low-risk client
  - 2 - Medium-risk client
  - 3 - High-risk client
  - 4 - Exceptional-risk client
- When crossing the boundary of the trading session and the clearing session, the data from the previous trading day in the table is cleared. If a client (7-digit client code) holds no positions after the clearing session, no new entry will appear in the table until positions are available.

4.29.1.2. Table sys_events: table of events

Table 139. 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	Unique ID of the event
sess_id	i4	Trading day ID
event_type	i4	Type of the event
message	c64	Description of the event
server_time	t	Event time

Notes:

- Possible types of events

  event_type = 1
  message = "session_data_ready"
  All data from the clearing system have been loaded into the trading system

  event_type = 5
  message = "clearing_started"
  Start of the clearing session

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

  event_type = 17
  message = "clearing_trades_inserted"
  Clearing trades have been generated

  event_type = 18
  message = "extension_of_limits_started_for_core"
  Limits have been extended

  event_type = 36
  message = "start_of_calendar_date"
  Beginning of the calendar day

  event_type = 37
  message = "settlement_session_started"
  Beginning of the settlement session

  event_type = 38
  message = "settlement_session_finished"
  End of the settlement session

  event_type = 47
  message = "settlpricecalc_finished"
  Settlement price has been fixed

4.30. Stream FORTS_SESSIONSTATE_REPL - Current trading day status (Type=I)

4.30.1. Data scheme

Tables:

- session_state - Current trading day status
- sys_events - Table of events

4.30.1.1. Table session_state: Current trading day status

Table 140. Fields of table session_state

Field	Type	Description
replID	i8	Service field of the replication subsystem

Field	Type	Description
replRev	i8	Service field of the replication subsystem
replAct	i8	Service field of the replication subsystem
sess_id	i4	Trading day ID
public_state	i4	Trading day state

Notes:

- Field public_state can take the following values:

  0 Trading is scheduled

  1 Trading is running

  2 Trading is suspended

  4 Trading is completed because the time is up

4.30.1.2. Table sys_events: table of events

Table 141. 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	Unique ID of the event
sess_id	i4	Trading day ID
event_type	i4	Type of the event
message	c64	Description of the event
server_time	t	Event time

Notes:

- Possible types of events

  event_type = 1
  message = "session_data_ready"
  All data from the clearing system have been loaded into the trading system

  event_type = 5
  message = "clearing_started"
  Start of the clearing session

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

  event_type = 17
  message = "clearing_trades_inserted"
  Clearing trades have been generated

  event_type = 18
  message = "extension_of_limits_started_for_core"
  Limits have been extended

  event_type = 36
  message = "start_of_calendar_date"
  Beginning of the calendar day

  event_type = 37
  message = "settlement_session_started"
  Beginning of the settlement session

  event_type = 38
  message = "settlement_session_finished"
  End of the settlement session

  event_type = 47
  message = "settlpricecalc_finished"

Settlement price has been fixed

4.31. Stream FORTS_INSTRUMENTSTATE_REPL - Statuses of instruments for the current trading day (Type=I)

4.31.1. Data scheme

Tables:

- instrument_state - Statuses of instruments for the current trading day
- sys_events - Table of events

4.31.1.1. Table instrument_state: Statuses of instruments for the current trading day

Table 142. Fields of table instrument_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
isin_id	i4	Instrument ID
public_state	i4	Instrument trading status

Notes:

- Field public_state can take the following values:

  0 Trading day for this instrument is scheduled. One can cancel orders for this instrument

  1 Trading day for this instrument is running. One can both add and cancel orders for this instrument.

  2 Trading in all instruments has been suspended. One can cancel orders for each instrument.

  4 Trading day for this instrument has been completed because the time is up. Orders can be neither added nor cancelled

  5 Trading in this instrument has been suspended. One can cancel orders for this instrument

  6 Opening auction: add orders.

  7 Opening auction: matching orders

  8 Trading day for this instrument is running. You can put orders to close position for this instrument.

  9 The opening auction for this instrument is running. You can put orders to close position for this instrument.

4.31.1.2. Table sys_events: table of events

Table 143. 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	Unique ID of the event
sess_id	i4	Trading day ID
event_type	i4	Type of the event
message	c64	Description of the event
server_time	t	Event time

Notes:

- Possible types of events

  event_type = 1
  message = "session_data_ready"
  All data from the clearing system have been loaded into the trading system

  event_type = 5

message = "clearing_started"
Start of the clearing session

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

event_type = 17
message = "clearing_trades_inserted"
Clearing trades have been generated

event_type = 18
message = "extension_of_limits_started_for_core"
Limits have been extended

event_type = 36
message = "start_of_calendar_date"
Beginning of the calendar day

event_type = 37
message = "settlement_session_started"
Beginning of the settlement session

event_type = 38
message = "settlement_session_finished"
End of the settlement session

event_type = 47
message = "settlpricecalc_finished"
Settlement price has been fixed

4.32. Stream FORTS_SECURITYGROUPSTATE_REPL - Group status of instruments for the current trading day (Type=I)

4.32.1. Data scheme

Tables:

- security_group_state - Group status of instruments for the current trading day
- sys_events - Table of events

4.32.1.1. Table security_group_state: Group status of instruments for the current trading day

The table is provided by the instrument status broadcast service, which is used for the accelerated broadcasts of information on trading statuses of instruments and sessions. By using the security_group_state table and instrument reference guides, clients can independently calculate the current statuses of instruments (for more details, see "Section 2.5.7, “Instrument status broadcast service”").

Table 144. Fields of table security_group_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
security_group_id	i8	Instrument group ID
type	i4	Halt type (group status of instruments)
trade_mode_mask	i4	Trading mode (Bitmask with one set bit)
group_mask	i8	Instrument group (Bitmask with one set bit)
section_id	i4	Section identifier
base_contract_id	i4	Underlying futures contract identifier
state	i4	Group mode
moment	t	Time of status change

Notes:

- Field type can take the following values:

  1 Trading day level

2 Section level

3 Section + Trading mode level

4 Underlying futures contract level

5 Instruments within the trading mode

6 Instruments within the trading mode + Underlying futures contract level

7 Instrument group level

8 Instrument group + Underlying futures contract level

- Field trade_mode_mask can take the following values:

0 (All bits cleared)	All trading modes
0x1	Futures, anonymous
0x4	Options, anonymous
0x2000	Calendar spreads, anonymous
0x10000	Collateral, negotiated
0x20000	One-day futures with auto-prolongation, anonymous

- Field group_mask can take the following values:

0 (All bits cleared)	All groups
0x1	Free
0x2	HaltGroupMask overflow
0x10	Scheduled execution
0x10	Intraday clearing session settlement (delivery)
0x20	Halts related to discrete auctions

- Field section_id can take the following values:

0 All sections

1 Stock section

2 Commodity Section

3 Money Section

- Field state can take the following values:

4 All prohibited

2 All prohibited except for order cancellation

16 Position increase prohibited

8 Order collection during the opening auction

0 All allowed

Modes are listed in order of decreasing priority (strictness of restriction)

4.32.1.2. Table sys_events: table of events

Table 145. 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	Unique ID of the event

Field	Type	Description
sess_id	i4	Trading day ID
event_type	i4	Type of the event
message	c64	Description of the event
server_time	t	Event time

Notes:

- Possible types of events

  event_type = 1
  message = "session_data_ready"
  All data from the clearing system have been loaded into the trading system

  event_type = 5
  message = "clearing_started"
  Start of the clearing session

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

  event_type = 17
  message = "clearing_trades_inserted"
  Clearing trades have been generated

  event_type = 18
  message = "extension_of_limits_started_for_core"
  Limits have been extended

  event_type = 36
  message = "start_of_calendar_date"
  Beginning of the calendar day

  event_type = 37
  message = "settlement_session_started"
  Beginning of the settlement session

  event_type = 38
  message = "settlement_session_finished"
  End of the settlement session

  event_type = 47
  message = "settlpricecalc_finished"
  Settlement price has been fixed

5. Commands description

5.1. Method AddOrder - Adding orders

Message type: 474

Reply message type: 179

Used to add orders for futures, options and multi-leg instruments.

Table 146. Input parameters

Name	Type	Default value	Description
broker_code	c4	""	Brokerage Firm code
isin_id	i4		Instrument unique ID
client_code	c3		Client code
dir	i4		Order direction
type	i4		Order type
amount	i4		Amount
price	c17		Price
comment	c20	""	Order comment
broker_to	c20	""	RTS code of the company to whom the negotiated order is addressed
ext_id	i4	0	External ID
is_check_limit	i4	0	Flag of checking price corridors for option orders
date_exp	c8	""	Order's expiration date
dont_check_money	i4	0	Whether to calculate client risks for given order
match_ref	c10	""	Identical text values entered by both trading parties to match negotiated orders
ncc_request	i1	0	Request to NCC for performing trades with Trading Member
compliance_id	c1	" "	Order adding method

Table 147. Execution result

Name	Type	Default value	Description
code	i4		Return code
message	c255		Message text
order_id	i8		Order's ID

Return codes:

0	operation completed successfully
Any other value	error

Notes:

- The ***'type'*** field may contain the following values:

  1 Day order

  2 Immediate-or-Cancel order

  3 Fill-or-Kill order

  4 Book-or-Cancel order

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

  1 buy order

  2 sell order

- The ***'price'*** field contains the order price as string: 'nnnnnnnnnn.mmmmm'.

- The ***'comment'*** field must not contain control characters (0x00-0x1F and 0x7F) and must be converted from cp1251 to utf-8.
- The ***'is_check_limit'*** field may contain the following values:

  0 Do not verify option price limits

  1 Verify option price limits

- The ***'date_exp'*** field contains order expiration date as 'YYYYMMDD'. Empty string indicates a common order. If there is certain date set in the string, the order are automatically relisted in the next session with a new number and a new time, until the date expires (multiday order). Orders with the expired date are removed automatically after the end of the trading session. When relisted, the orders are verified for instrument availability, client details and funds availability. Date may vary in the range from >= today to <= 1 year ahead. You cannot indicate the order expiration date in negotiated orders.
- The ***'dont_check_money'*** order parameter may contain the following values:

  0 Verify collaterals for client section

  1 Do not verify collaterals for client section

  The parameter is eligible for using by a login with the appropriate right. All other logins using this parameter will have their orders rejected.

- The ***'match_ref'*** field contains identical text values entered by both trading parties to match the negotiated orders. Negotiated orders can only be exercised when the value of the ***'match_ref'*** field of one order exactly matches the value of another. When storing a field value, whitespace from the beginning and end of the string is trimmed. A value consisting only of spaces is interpreted by the system as NULL.
- Flag ***'ncc_request'*** indicates a request to National Clearing Centre for performing trades with Trading Member. This request is used to settle a Trading Member's obligations against a Clearing Member, also in case of the Trading Member's insufficient amount of collateral. The request can be sent only by a Clearing Member which is not simultaneously a Trading Member, and only from a login belonging to a Clearing Firm/Brokerage Firm.
- Field 'compliance_id' can take the following values:

" " (space or empty string)	Not filled in/Not specified
M	Manual input
S	As a result of the conditional request (stop-loss order)
R	As a result of the robot algorithm work
A	As a result of the auto-following algorithm
D	Covering a position as a result of an unexecuted Margin Call

  Please note that the ***compliance_id*** parameter does not affect the trade execution, are not checked for valid values, and is displayed in the gateway without changes (FORTS_REFDATA_REPL:orders_log/multileg_orders_log, FORTS_USERORDERBOOK_REPL:orders) and is transmitted to the Regulator's reports. It is impossible to correct it after the fact.

5.2. Method DelOrder - Deletion of orders

Message type: 461

Reply message type: 177

Used to delete orders for futures, options and multi-leg instruments.

Table 148. Input parameters

Name	Type	Default value	Description
broker_code	c4	""	Brokerage Firm code
order_id	i8		Order ID to delete
ncc_request	i1	0	Request to NCC for performing trades with Trading Member
client_code	c3		Client code
isin_id	i4		Instrument unique ID

Table 149. Execution result

Name	Type	Default value	Description
code	i4		Return code
message	c255		Message text

Name	Type	Default value	Description
amount	i4		Order's amount on deletion moment

Return codes:

0 operation completed successfully

Any other value error

Notes:

- The return code = 14 (order is not found for removing) indicates that there is no such order in queue. Possible reasons: wrong order number, or the order has not been placed today. It does not make sense to continue sending removal requests for the same order number (may be useful for automatic systems).
- Flag ***'ncc_request'*** indicates cancellation of a Request to NCC for performing trades with Trading Member. Please note that requests may be cancelled only by a Clearing Member which is not simultaneously a Trading Member, and only from a login belonging to a Clearing Firm/Brokerage Firm.

5.3. Method DelUserOrders - Mass cancel orders

Message type: 466

Reply message type: 186

Mass cancellation of all orders under a criteria. You can cancel orders for futures, options and multi-leg instruments.

Table 150. Input parameters

Name	Type	Default value	Description
broker_code	c4	""	Brokerage Firm code
buy_sell	i4		Whether to cancel orders on their directions
non_system	i4		Whether to cancel orders on their negotiated sign
code	c3		Client code
base_contract_code	c25		Underlying asset code
ext_id	i4	0	External ID
isin_id	i4		Instrument unique ID
instrument_mask	i1		Instrument group mask

Table 151. Execution result

Name	Type	Default value	Description
code	i4		Return code
message	c255		Message text
num_orders	i4		Number of canceled orders

Return codes:

0 operation completed successfully

Any other value error

Notes:

- The ***'buy_sell'*** parameter may contain the following values:

  1 Buy orders

  2 Sell orders

  3 All orders

- The ***'non_system'*** parameter may contain the following values:

  0 Common orders

  1 Negotiated orders

  2 All orders

- The ***'instrument_mask'*** parameter are bit mask:

  0x1 Futures

  0x2 Options

  0x4 Multi-leg instruments

- If the ***'code'*** parameter is not set or is '%%%', then all orders for all clients' accounts are removed.
- If the ***'base_contract_code'*** parameter is not set or is '%', then all orders for all contracts are removed.
- If the ***'ext_id'*** parameter value is not 0, then all orders with the corresponding ***'ext_id'*** are removed. The values of the ***'buy_sell'***, ***'non_system'***, ***'base_contract_code'***, and ***'isin_id'*** parameters are ignored, but their values must be within the allowed range.
- This command is used to cancel orders only. The request to NCC for performing trades with Trading Member will not be cancelled by this command.
- The command is not available for a Clearing Member which is not simultaneously a Trading Member. In order to mass cancel orders, a Clearing Member may use command DelOrdersByBFLimit.
- If the transaction is completed successfully (return code 0), but no orders are canceled (***num_orders=0***), a penalty is charged for erroneous transactions.

5.4. Method MoveOrder - Modify orders

Message type: 476

Reply message type: 176

Used to modify orders for futures and options.

Table 152. Input parameters

Name	Type	Default value	Description
broker_code	c4	""	Brokerage Firm code
regime	i4		Mode
order_id1	i8		ID of the 1st order to remove
amount1	i4		New amount for the 1st order
price1	c17		New price for the 1st order
ext_id1	i4		New external ID for the 1st order
order_id2	i8		ID of the 2nd order to remove
amount2	i4		New amount for the 2nd order
price2	c17		New price for the 2nd order
ext_id2	i4		New external ID for the 2nd order
is_check_limit	i4	0	Flag of checking price corridors for option orders
ncc_request	i1	0	Request to NCC for performing trades with Trading Member.
client_code	c3		Client code
isin_id	i4		Instrument unique ID
compliance_id	c1	" "	Order adding method

Table 153. Execution result

Name	Type	Default value	Description
code	i4		Return code
message	c255		Message text
order_id1	i8		New ID of the 1st modified order
order_id2	i8		New ID of the 2nd modified order

Return codes:

0 operation completed successfully

Any other value error

Notes (in this note, the term 'amount' means the number of units of the instrument):

- The ***'regime'*** parameter defines the command work mode. It may contain the following values:

  0 Do not change amount of orders. The current amounts of orders remains unchanged, the newly sent amounts are ignored.

  1 Change amounts of orders. If there is any order found, it will be replaced with the new order with new price and amount.

  2 Remove old orders. If any order volume does not coincide with the newly sent one, both orders are removed. Otherwise, the orders will be modified.

  3 Set orders amounts to that of received, excluding the matched part (not less than 0). If the amount received is less than the amount of the matched part, both orders will be removed.

- The ***'is_check_limit'*** may contain the following values:

  0 Do not verify option price limits

  1 Verify option price limits

- All new orders will be auctioned.
- Orders can be shifted only within the same trading instrument and only within the same client register.
- Orders are not shifted by multy-legs.
- Negotiated orders are not shifted.
- When shifting, the direction of order is not changed.
- Once an order has been removed (or shifted, or fully matched), it is not relisted, and the error message appears.
- If one order of a pair cannot be shifted, then another order is not shifted, too, and the error message appears.
- If two orders with opposite directions are shifted in the way their prices coincide, then the parameters are considered as incorrect, shifting is not performed, and the error message appears.
- If, when shifting a pair of orders, one order meets a cross-trade (matching an order sent from either the same ITN or the same client register), than it is rejected, and another order of the pair is shifted.
- Upon moving orders, the ***'date_exp'*** parameters are transferred into new orders.
- After the command has been processed, the ***'order_id1'*** field and ***'order_id2'*** field are filled with new orders numbers. If no order has been placed, the corresponding field is set to 0.
- Flag '***ncc_request***' indicates a change made to 'Request to NCC for performing trades with Trading Member'. Please note that changes to requests may be made only by a Clearing Member which is not simultaneously a Trading Member, and only from a login belonging to a Clearing Firm/Brokerage Firm.
- If the moving of a Book-or-Cancel order leads to its execution, then the moving rolled back, and order itself remains in the system.
- When a pair of Book-or-Cancel orders are moved, one of them may be executed, it is rolled back, and the other order is moved.
- Field 'compliance_id' can take the following values:

" " (space or empty string)	Not filled in/Not specified
M	Manual input
S	As a result of the conditional request (stop-loss order)
R	As a result of the robot algorithm work
A	As a result of the auto-following algorithm
D	Covering a position as a result of an unexecuted Margin Call

  Please note that the ***compliance_id*** parameter does not affect the trade execution, are not checked for valid values, and is displayed in the gateway without changesto trade execution, does not checked for valid values and parameter is displayed in the gateway without changes (FORTS_REFDATA_REPL:orders_log/multileg_orders_log, FORTS_USERORDERBOOK_REPL:orders) and is transmitted to the Regulator's reports. It is impossible to correct it after the fact.

5.5. Method IcebergAddOrder - Adding iceberg orders

Message type: 475

Reply message type: 180

Used to add iceberg orders.

Table 154. Input parameters

Name	Type	Default value	Description
broker_code	c4	""	Brokerage Firm code
isin_id	i4		Instrument unique ID
client_code	c3		Client code
dir	i4		Order direction
type	i4		Order type
disclose_const_amount	i4		Number of instrument units in the visible part of the iceberg order
iceberg_amount	i4		Total amount of instruments in iceberg order
variance_amount	i4	0	Amplitude of deviation (in contracts) of the random allowance to the visible part of the iceberg order
price	c17		Price
comment	c20	""	Order comment
ext_id	i4	0	External ID
is_check_limit	i4	0	Flag of checking price corridors for option orders
date_exp	c8	""	Order's expiration date
dont_check_money	i4	0	Whether to calculate client risks for given order
ncc_request	i1	0	Request to NCC for performing trades with Trading Member
compliance_id	c1	" "	Order adding method

Table 155. Execution result

Name	Type	Default value	Description
code	i4		Return code
message	c255		Message text
iceberg_order_id	i8		Iceberg order ID

Return codes:

0 operation completed successfully

Any other value error

Notes:

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

  1 buy order

  2 sell order

- The ***'type'*** field may contain the following values:

  1 Day order

  4 Book-or-Cancel order

- The ***'price'*** field contains the order price as string: 'nnnnnnnnnn.mmmmm'.
- The ***'comment'*** field must not contain control characters (0x00-0x1F and 0x7F) and must be converted from cp1251 to utf-8.
- The ***'is_check_limit'*** field may contain the following values:

  0 Do not verify option price limits

  1 Verify option price limits

- The ***'date_exp'*** field contains order expiration date as 'YYYYMMDD'. Empty string indicates a common order. If there is certain date set in the string, the order are automatically relisted in the next session with a new number and a new time, until the date expires (multiday order). Orders with the expired date are removed automatically after the end of the trading session. When relisted, the orders are verified for instrument availability, client details and funds availability. Date may vary in the range from >= today to <= 1 year ahead.
- The ***'dont_check_money'*** order parameter may contain the following values:

  ○ 0 – verify collaterals for client section

  - 1 – do not verify collaterals for client section

  The parameter is eligible for using by a login with the appropriate right. All other logins using this parameter will have their orders rejected.

- Flag ***'ncc_request'*** indicates a request to National Clearing Centre for performing trades with Trading Member. This request is used to settle a Trading Member's obligations against a Clearing Member, also in case of the Trading Member's insufficient amount of collateral. The request can be sent only by a Clearing Member which is not simultaneously a Trading Member, and only from a login belonging to a Clearing Firm/Brokerage Firm.

- Field 'compliance_id' can take the following values:

" " (space or empty string)	Not filled in/Not specified
M	Manual input
S	As a result of the conditional request (stop-loss order)
R	As a result of the robot algorithm work
A	As a result of the auto-following algorithm
D	Covering a position as a result of an unexecuted Margin Call

  Please note that the ***compliance_id*** parameter does not affect the trade execution, are not checked for valid values, and is displayed in the gateway without changes (FORTS_REFDATA_REPL:orders_log/multileg_orders_log, FORTS_USERORDERBOOK_REPL:orders) and is transmitted to the Regulator's reports. It is impossible to correct it after the fact.

5.6. Method IcebergDelOrder - Deletion of iceberg orders

Message type: 464

Reply message type: 182
Used to delete iceberg orders. The command can work both on 'public_order_id' and on 'private_order_id'. That the command will work on 'public_order_id' only if the visible part with such a number is still in the system (has not been matched), otherwise an error will be returned about the absence of an order with such a number. Therefore, we recommend working with iceberg orders on 'private_order_id'.

Table 156. Input parameters

Name	Type	Default value	Description
broker_code	c4	""	Brokerage Firm code
order_id	i8		Order ID to delete
isin_id	i4		Instrument unique ID
ncc_request	i1	0	Request to NCC for performing trades with Trading Member

Table 157. Execution result

Name	Type	Default value	Description
code	i4		Return code
message	c255		Message text
amount	i4		Order's amount on deletion moment

Return codes:

0	operation completed successfully
Any other value	error

Notes:

- The return code = 14 (order is not found for removing) indicates that there is no such order in queue. Possible reasons: wrong order number, or the order has not been placed today. It does not make sense to continue sending removal requests for the same order number (may be useful for automatic systems).

- Flag ***'ncc_request'*** indicates cancellation of a Request to NCC for performing trades with Trading Member. Please note that requests may be cancelled only by a Clearing Member which is not simultaneously a Trading Member, and only from a login belonging to a Clearing Firm/Brokerage Firm.

5.7. Method IcebergMoveOrder - Modify iceberg orders

Message type: 477

Reply message type: 181

Used to modify iceberg orders. The command can work both on 'public_order_id' and on 'private_order_id'. That the command will work on 'public_order_id' only if the visible part with such a number is still in the system (has not been matched), otherwise an error will be returned about the absence of an order with such a number. Therefore, we recommend working with iceberg orders on 'private_order_id'.

Table 158. Input parameters

Name	Type	Default value	Description
broker_code	c4	""	Brokerage Firm code
order_id	i8		ID of the order to be modified
isin_id	i4		Instrument unique ID
price	c17		New price of the order
ext_id	i4		New external ID of the order
ncc_request	i1	0	Request to NCC for performing trades with Trading Member.
is_check_limit	i4	0	Flag of checking price corridors for option orders
compliance_id	c1	" "	Order adding method

Table 159. Execution result

Name	Type	Default value	Description
code	i4		Return code
message	c255		Message text
order_id	i8		New ID of the modified order

Return codes:

0	operation completed successfully
Any other value	error

Notes:

- The ***'is_check_limit'*** may contain the following values:

  0 Do not verify option price limits

  1 Verify option price limits

- Flag '***ncc_request***' indicates a change made to 'Request to NCC for performing trades with Trading Member'. Please note that changes to requests may be made only by a Clearing Member which is not simultaneously a Trading Member, and only from a login belonging to a Clearing Firm/Brokerage Firm.

- Field 'compliance_id' can take the following values:

" " (space or empty string)	Not filled in/Not specified
M	Manual input
S	As a result of the conditional request (stop-loss order)
R	As a result of the robot algorithm work
A	As a result of the auto-following algorithm
D	Covering a position as a result of an unexecuted Margin Call

  Please note that the ***compliance_id*** parameter does not affect the trade execution, are not checked for valid values, and is displayed in the gateway without changes (FORTS_REFDATA_REPL:orders_log/multileg_orders_log, FORTS_USERORDERBOOK_REPL:orders) and is transmitted to the Regulator's reports. It is impossible to correct it after the fact.

5.8. Method ChangeClientMoney - Change client limits

Message type: 458

Reply message type: 187

The command allows to change funds limits for a client's account.

Table 160. Input parameters

Name	Type	Default value	Description
broker_code	c4	""	Brokerage Firm code

Name	Type	Default value	Description
mode	u1		Mode
code	c3		Client code
coeff_im	c17	""	Clients collateral ratio
is_auto_update_limit	i4	-1	Flag of automatic adjustment of the limit by the amount of income after clearing
check_limit	i4	1	Funds sufficiency verification flag
limit_money	c17	""	Funds limit

Table 161. Execution result

Name	Type	Default value	Description
code	i4		Return code
message	c255		Message text

Return codes:

0 operation completed successfully

Any other value error

Notes:

- Command work mode (the ***'mode'*** field):

  11 Remove the limit, disable checking for sufficient funds

  12 Set the funds limit in the amount of ***'limit_money'***

  13 Change the funds limit by the amount of ***'limit_money'***

- The ***'is_auto_update_limit'*** flag, being set to '1', allows to automatize the limit changing process in accordance with the previous day results. The value "-1" for the ***'is_auto_update_limit'*** parameter means that the value is not set by the user.

- To change parameter ***'is_auto_update_limit'***, the mode '13' must be used. The ***'limit_money'*** parameter value must be set to '0'.

- The following values are set in the ***check_limit*** parameter:

  0 Do not verify funds sufficiency. Change limit unconditionally.

  1 Verify funds sufficiency. Do not change limit if there are insufficient funds

- The ***check_limit*** check is performed only when decreasing the limit. When increasing the limit, no check is performed, even if it is enabled. This includes increasing the limit from a negative value to another negative value. For example, -5000 -> -3000 (delta +2000, but the resulting number is still negative). Permitted operations with limit check enabled (***check_limit***=1):

  ○ Increasing the limit from negative to negative: -10000 -> -5000 - Check not performed, Limit changed / Success

  ○ Increasing the limit from negative to positive: -10000 -> 1000 - Check not performed, Limit changed / Success

  ○ Decreasing the limit from positive to positive: 5000 -> 3000 - Check passed, Limit changed / Success

  Prohibited operations:

  ○ Decreasing the limit from positive to negative: 5000 -> -3000 - Check failed, Limit not changed / Error

  ○ Decreasing the limit from negative to negative: -5000 -> -10000 - Check failed, Limit not changed / Error

- For the field type 'c17', it is possible to specify empty string in order to prevent changing the parameter value, which had been sent into the trading system before.

5.9. Method ChangeBFMoney - Change brokerage firm limits

Message type: 426

Reply message type: 107

The command allows to change amounts of money in your brokerage firms' accounts. Once the account size increases, the required amount of money is transferred from the clearing firm's account. When you decrease the account size, the required amount of money is deposited back to the clearing firm's account.

Table 162. Input parameters

Name	Type	Default value	Description
broker_code	c4	""	Brokerage Firm code
mode	i4		Mode
code	c2		Brokerage firm code
limit_money	c17		Funds limit

Table 163. Execution result

Name	Type	Default value	Description
code	i4		Return code
message	c255		Message text

Return codes:

0 operation completed successfully

Any other value error

Notes:

- Comand work mode (the ***'mode'*** field):

  12 Set limit equal to***limit_money***

  13 Change limit***limit_money***

- To get access to the procedure, a clearing firm's login must obtain the appropriate right from the Trading Administrator.

5.10. Method OptChangeExpiration - Request for the exercise of options

Message type: 471

Reply message type: 190

Submission of request for early exercise of options, for cancellation of automatic exercise of options. This command can also be used to submit a request to exercise "out-of-the-money" options on the day of expiration.

Table 164. Input parameters

Name	Type	Default value	Description
broker_code	c4	""	Brokerage Firm code
mode	i4		Mode
order_id	i8		ID of the request to exercise the options
code	c3		Client code
isin	c25		Instrument code
amount	i8	0	Volume of exercise

Table 165. Execution result

Name	Type	Default value	Description
code	i4		Return code
message	c255		Message text
order_id	i8		ID of the request to exercise the options

Return codes:

0 operation completed successfully

Any other value error

Notes:

- Command work mode (the ***'mode'*** field):

  11 Delete

12 Add/Update

- The key fields for exercise requests are: ***'isin'*** and ***'code'***. For one combination of 'isin + code', more than one exercise request cannot be submitted.
- When adding a new exercise request, set '0' in the 'order_id' field. It should be remembered that if an request with the same 'isin + code' already exists in the system, then instead of adding the new request for exercise, the old request will be modified.
- Operations 'Delete' or 'Update' of request can be made:
  - By 'order_id' ('code' and 'isin' are not set).
  - By 'code + isin' (order_id=0).
  - By 'order_id' and 'code + isin'. In this case, after finding the request by 'order_id', it checks if the 'code' and 'isin' match the sent values.
- You can turn off the automatic exercise feature by adding a negative amount of option contracts into the 'OptChangeExpiration' command (field 'amount'). The amount of option contracts specified will not exercise automatically.
- A broker may impose a blanket ban on accepting client requests regarding options. In this case, no client will be able to independently exercise an option early or opt out of automatic option expiration.

5.11. Method FuturesExecutionRequest - Exercise requests of daily futures contracts with auto-prolongation

Message type: 470

Reply message type: 189

Submission of the exercise requests of daily futures contracts with auto-prolongation. You can add, change and delete requests.

Table 166. Input parameters

Name	Type	Default value	Description
broker_code	c4	""	Brokerage Firm code
mode	i4		Mode
order_id	i8	0	ID of the request to exercise the futures
code	c3		Last three characters of client code
isin	c25		Future code
amount	i8		Number of contracts to be exercised

Table 167. Execution result

Name	Type	Default value	Description
code	i4		Return code
message	c255		Message text
order_id	i8		ID of the request to exercise the futures

Return codes:

0	operation completed successfully
Any other value	error

Notes:

- Command work mode (the ***'mode'*** field):

  11 Delete

  12 Add/Update

- The key fields for exercise requests are: ***'isin'*** and ***'code'***. For one combination of 'isin + code', more than one exercise request cannot be submitted.
- When adding a new exercise request, set '0' in the 'order_id' field. It should be remembered that if an request with the same 'isin + code' already exists in the system, then instead of adding the new exercise request, the old request will be modified.
- Operations 'Delete' or 'Update' of request can be made:

  - By 'order_id' ('code' and 'isin' are not set).
  - By 'code + isin' (order_id=0).
  - By 'order_id' and 'code + isin'. In this case, after finding the request by 'order_id', it checks if the 'code' and 'isin' match the sent values.
- An exercise requests cannot be deleted (updated) if it was submitted by a higher-level login.

5.12. Method FutChangeClientProhibit - Modify client's restrictions for futures

Message type: 469

Reply message type: 115

Table 168. Input parameters

Name	Type	Default value	Description
broker_code	c4	""	Brokerage Firm code
mode	i4		Mode
code	c3		Code of the client's account or '%%%' – for all
base_contract_code	c25		Code of the underlying asset or '%' - for all
isin	c25		Instrument code or '%' - for all
state	i4	0	Restriction
state_mask	i4	3	Mask for parameter 'state'
client_priority	i4	0	Custom priority

Table 169. Execution result

Name	Type	Default value	Description
code	i4		Return code
message	c255		Message text

Return codes:

0 operation completed successfully

Any other value error

Notes:

- The ***'mode'*** field specifies the command work mode:

  11 delete

  12 update/insert
- The ***'state'*** field may contain the following values:

  0 No prohibitions. Used for pinpoint permission in case of a broader prohibition;

  1 Prohibited to open positions;

  2 Prohibited to add any orders;

  3 Prohibited to open sell positions;

  0x08 BF prohibition to add requests for exercising;

  0x10 Chief Trader prohibition to add requests for exercising; but to himself - it is possible;

  0x20 Prohibition of requests without auto-confirmation (RFS);

  0x40 Prohibition to request liquidity stream (RFS);

  0x80 Prohibition to perform trades with insufficient number of quotes (RFS);

  0x100 Prohibition to request liquidity stream with limited lifetime of quotes (RFS).
- The 'state_mask' bitmask defines those bits of the 'state' field whose values need to be changed as a result of the command execution. For example, in the prohibition existing in the system, it is necessary to change the prohibition on orders only (0-3), without affecting

the remaining bits, in this case this parameter should be set to '3', and the operation will be performed only on the first two bits only of the 'state' field only.

- When setting a certain instrument in the ***'isin'*** field, the code of the corresponding underlying asset must be set in the ***'base_contract_code'*** field.
- The 'client_priority' field can take the following values:

  10 low

  11 medium

  12 high

  If the 'client_priority' field contains the value '10', '11' or '12', then a prohibition is set with the specified user priority. If the custom priority is not specified (the value is '0'), the priority will be set automatically, in accordance with the prohibition parameters.

5.13. Method OptChangeClientProhibit - Modify client's restrictions for options

Message type: 468

Reply message type: 117

Table 170. Input parameters

Name	Type	Default value	Description
broker_code	c4	""	Brokerage Firm code
mode	i4		Mode
code	c3		Code of the client's account or '%%%' – for all
base_contract_code	c25		Code of the underlying asset or '%' - for all
isin	c25		Instrument code or '%' - for all
state	i4	0	Restriction
state_mask	i4	8	Mask for parameter 'state'
client_priority	i4	0	Custom priority

Table 171. Execution result

Name	Type	Default value	Description
code	i4		Return code
message	c255		Message text

Return codes:

0 operation completed successfully

Any other value error

Notes:

- Command work mode (the ***'mode'*** field):

  11 delete

  12 update/insert

- The ***'state'*** field may contain the following values:

  0 No prohibitions. Used for pinpoint permission in case of a broader prohibition;

  1 Prohibited to open positions;

  2 Prohibited to add any orders;

  3 Prohibited to open sell positions;

  0x08 BF prohibition to add requests for exercising;

  0x10 Chief Trader prohibition to add requests for exercising; but to himself - it is possible;

  0x20 Prohibition of requests without auto-confirmation (RFS);

0x40 Prohibition to request liquidity stream (RFS);

0x80 Prohibition to perform trades with insufficient number of quotes (RFS);

0x100 Prohibition to request liquidity stream with limited lifetime of quotes (RFS).

- The 'state_mask' bitmask defines those bits of the 'state' field whose values need to be changed as a result of the command execution. For example, in the prohibition existing in the system, it is necessary to change the prohibition on orders only (0-3), without affecting the remaining bits, in this case this parameter should be set to '3', and the operation will be performed only on the first two bits only of the 'state' field only.
- Limits for futures and options are applied independently.
- The 'client_priority' field can take the following values:

  10 low

  11 medium

  12 high

  If the 'client_priority' field contains the value '10', '11' or '12', then a prohibition is set with the specified user priority. If the custom priority is not specified (the value is '0'), the priority will be set automatically, in accordance with the prohibition parameters.

5.14. Method ExchangeBFMoney - Transfer of funds between two BFs of the same SA

Message type: 427

Reply message type: 130

The command allows to transfer funds between two BF belonging to the same SA.

Table 172. Input parameters

Name	Type	Default value	Description
broker_code	c4	""	Brokerage Firm code
mode	i4		Mode
code_from	c2		Source account code
code_to	c2		Destination account code
amount_money	c17		Amount of collateral to transfer in roubles

Table 173. Execution result

Name	Type	Default value	Description
code	i4		Return code
message	c255		Message text

Return codes:

0 operation completed successfully

Any other value error

Notes:

- Command work mode (the ***'mode'*** field):

  1 Transfer only at trading

  3 Transfer at trading and clearing

5.15. Method OptRecalcCS - Recalculate central strike request

Message type: 422

Reply message type: 132

The command allows to recalculate the central strike in accordance with the market-maker's obligations (for which the "Offset by demand" recalculation option is selected). Developed for market-makers.

Table 174. Input parameters

Name	Type	Default value	Description
broker_code	c4	""	Brokerage Firm code
option_series_id	i4		Series of Options ID

Table 175. Execution result

Name	Type	Default value	Description
code	i4		Return code
message	c255		Message text

Return codes:

0 operation completed successfully

Any other value error

5.16. Method TransferClientPosition - Transfer client positions

Message type: 430

Reply message type: 173

The command allows to transfer positions between your brokerage firms' accounts.

Table 176. Input parameters

Name	Type	Default value	Description
broker_code	c4	""	Brokerage Firm code
code_from	c7		Donor code
code_to	c7		Recipient code
isin	c25		Instrument ID
amount	i8		Amount of position to transfer

Table 177. Execution result

Name	Type	Default value	Description
code	i4		Return code
message	c255		Message text

Return codes:

0 operation completed successfully

Any other value error

Notes:

To get access to the procedure, a clearing firm's login must obtain the appropriate right from the Trading Administrator.

The transfer of a client's position to a non-qualified investor is prohibited for instruments requiring the qualified investor attribute.

5.17. Method OptChangeRiskParametersNextSession - Risk parameters settings for options

Message type: 69

Reply message type: 140

The command allows to change risk parameters of options.

Table 178. Input parameters

Name	Type	Default value	Description
broker_code	c4	""	Brokerage Firm code
client_code	c3		Client code

Name	Type	Default value	Description
num_clr_2delivery	i4	0	Number of clearing sessions (including intraday clearing sessions) to turn on automatic exercise scenario of risk calculation for the non-quarterly series of options with the closest expiration date for this account
use_broker_num_clr_2delivery	i1	0	Use broker num_clr_2delivery
exp_weight	c4	"0"	Expiration Risk Weight
use_broker_exp_weight	i1	0	Use broker exp_weight
short_option_minimum_charge_ratio	c6	"-1"	Individual coefficient of SOMC scenario weight

Table 179. Execution result

Name	Type	Default value	Description
code	i4		Return code
message	c255		Message text

Return codes:

0 operation completed successfully

Any other value error

Notes:

To get access to the procedure, a clearing firm's login must obtain the appropriate right from the Trading Administrator.

5.18. Method ChangeBFParametersNextSession - Change BF's parameters by a clearing member

Message type: 442

Reply message type: 162

The command allows a Clearing member to change BF's parameters. Please note that the Clearing member must belong to a Clearing Firm to use the command. All changes made will be applied during the clearing session.

Table 180. Input parameters

Name	Type	Default value	Description
broker_code	c4	""	Brokerage Firm code
code_bf	c2		BF code
margin_type	i4	-1	Margin type, according to BF's sections: • 3 - half nett • 4 - nett
calendar_spread_margin_type	i1	-1	Margin type for calendar spreads, for BF portfolio: • 3 - half nett • 4 - nett
num_clr_2delivery	i4	-1	Number of clearing sessions before expiration to start BF expiration scenarios calculation.
exp_weight	c17	""	Expiration scenario weight for BF, in total collateral.
go_ratio	c17	""	Total collateral ratio value, for BF.
check_limit_on_withdrawal	i4	-1	Verify collateral sufficiency, for BF, upon funds depositing/withdrawal: • 1 - Verify • 0 - Do not verify
limit_tied_to_money	i4	-1	BF trading limit accordance with the BF's total funds amount (all sections): • 1 - maintain accordance • 0 - virtual (independent) limit.

Name	Type	Default value	Description
check_limit_for_orders	i4	-1	Verification of collateral sufficiency upon adding orders, for BF: • 1 - Verify • 0 - Do not verify
no_fut_discount	i4	-1	Discount on futures for BF portfolio: • 1 - Discount prohibited • 0 - Discount allowed
ics_margin_type	i1	-1	Margin type for cross-contract spreads: • 3 - half nett • 4 - nett

Table 181. Execution result

Name	Type	Default value	Description
code	i4		Return code
message	c255		Message text

Return codes:

0 operation completed successfully

Any other value error

Notes:

- For the field type 'c17', it is possible to specify empty string in order to prevent changing the parameter value, which had been sent into the trading system before.
- For the fields type 'i4' and 'i1', it is possible to specify '-1' in order to prevent changing the parameter value, which had been sent into the trading system before.

5.19. Method ChangeClientParameters - Change parameters of client account

Message type: 443

Reply message type: 178

The command allows to change parameters for client accounts by a Clearing member. Please note that the login must belong to a Brokerage Firm/Clearing Firm to use the command.

Table 182. Input parameters

Name	Type	Default value	Description
broker_code	c4	""	Brokerage Firm code
code	c3		Client code
coeff_go	c17	""	Client IM ratio
no_fut_discount	i4	-1	Flag of prohibition to provide discounts for futures

Table 183. Execution result

Name	Type	Default value	Description
code	i4		Return code
message	c255		Message text

Return codes:

0 operation completed successfully

Any other value error

Notes:

- The ***'coeff_go'*** field can take a value from 0 to 10,00000.

- For the field type 'c17', it is possible to specify empty string in order to prevent changing the parameter value, which had been sent into the trading system before.
- For the field type 'i4', it is possible to specify '-1' in order to prevent changing the parameter value, which had been sent into the trading system before.

5.20. Method ChangeClientParametersNextSession - Change parameters of client account in clearing session

Message type: 441

Reply message type: 163

The command allows to change parameters for client accounts by a Clearing member. Please note that the Clearing member must belong to a Brokerage Firm/Clearing Firm to use the command.

Table 184. Input parameters

Name	Type	Default value	Description
broker_code	c4	""	Brokerage Firm code
code	c3		Client code
calendar_spread_margin_type	i1	-1	Margin type for calendar spreads, for client: • 3 - half nett • 4 - nett
ics_margin_type	i1	-1	Margin type for cross-contract spreads: • 3 - half nett • 4 - nett

Table 185. Execution result

Name	Type	Default value	Description
code	i4		Return code
message	c255		Message text

Return codes:

0 operation completed successfully

Any other value error

Notes:

- For the field type 'i1', it is possible to specify '-1' in order to prevent changing the parameter value, which had been sent into the trading system before.

5.21. Method ChangeBFClientDefaultParametersNextSession - Change default parameters of client sections

Message type: 402

Reply message type: 602

The command allows to change default parameters for client sections of a single BF. Please note that the login must belong to a Brokerage Firm/Clearing Firm to use the command. All changes made will be applied during the clearing session.

Table 186. Input parameters

Name	Type	Default value	Description
broker_code	c4	""	Brokerage Firm code
code_bf	c2		BF code
num_clr_2delivery_client_default	i4	-1	Number of clearing sessions before expiration to start clients expiration scenarios calculation.
exp_weight_client_default	c17	""	Expiration scenario weight for client, in total collateral.
no_fut_discount_client_default	i4	-1	Discount on futures for client section portfolios:

Name	Type	Default value	Description
			• 1 - Discount prohibited • 0 - Discount allowed

Table 187. Execution result

Name	Type	Default value	Description
code	i4		Return code
message	c255		Message text

Return codes:

0 operation completed successfully

Any other value error

Notes:

- For the field type 'c17', it is possible to specify empty string in order to prevent changing the parameter value, which had been sent into the trading system before.
- For the field type 'i4', it is possible to specify '-1' in order to prevent changing the parameter value, which had been sent into the trading system before.

5.22. Method ChangeBFClientBaseContractParametersNextSession - Changing the parameters of BF clients for the basic contract

Message type: 1057

Reply message type: 1143

The command is used to change the IM ratio under the underlying contract for all clients of this BF. Please note that the login must belong to a Brokerage Firm/Clearing Firm to use the command. All changes made will be applied during the clearing session.

Table 188. Input parameters

Name	Type	Default value	Description
broker_code	c4	""	Brokerage Firm code
bf_code	c2	""	BF code set by the client
base_contract_code	c25	""	Underlying contract code
coeff_im	c17	""	Coefficient of IM under the underlying contract for all clients of this BF

Table 189. Execution result

Name	Type	Default value	Description
code	i4		Return code
message	c255		Message text

Return codes:

0 operation completed successfully

Any other value error

Notes:

- The ***'coeff_im'*** field can take a value from 0 to 10,00000.
- When setting the Coefficient of IM for a Brokerage firm (BF) + Underlying asset (UA), if the UA is part of the cross-contract spread, the coefficient value is applied to all underlying assets within that cross-contract spread. Specifically, for the given brokerage firm, the highest Coefficient of IM value among all UAs within the cross-contract spread is selected. In this case, the system will return a warning: "Warning: The UA is part of the cross-contract spread. The highest Coefficient of IM value among the UAs within the cross-contract spread for this BF will be applied."

5.23. Method ChangeBFLimit - Change BF trading limits

Message type: 428

Reply message type: 161

The command allows to change BF trading limits

Table 190. Input parameters

Name	Type	Default value	Description
broker_code	c4	""	Brokerage Firm code
mode	i4		Mode
code	c2		Brokerage firm code
limit_money	c17		Funds limit
check_limit	i4		Verify BF funds sufficiency

Table 191. Execution result

Name	Type	Default value	Description
code	i4		Return code
message	c255		Message text

Return codes:

0 operation completed successfully

Any other value error

Notes:

- Command work mode (the ***'mode'*** field):

  12 Set limits value to that of ***limit_money***

  13 Change limits value to that of ***limit_money***

- The following values are set in the ***check_limit*** parameter:

  0 Do not verify

  1 Verify

- The ***check_limit*** check is performed only when decreasing the limit. When increasing the limit, no check is performed, even if it is enabled. This includes increasing the limit from a negative value to another negative value. For example, -5000 -> -3000 (delta +2000, but the resulting number is still negative). Permitted operations with limit check enabled (***check_limit***=1):

  - Increasing the limit from negative to negative: -10000 -> -5000 - Check not performed, Limit changed / Success
  - Increasing the limit from negative to positive: -10000 -> 1000 - Check not performed, Limit changed / Success
  - Decreasing the limit from positive to positive: 5000 -> 3000 - Check passed, Limit changed / Success

  Prohibited operations:

  - Decreasing the limit from positive to negative: 5000 -> -3000 - Check failed, Limit not changed / Error
  - Decreasing the limit from negative to negative: -5000 -> -10000 - Check failed, Limit not changed / Error

5.24. Method CODHeartbeat - Heartbeat message for Cancel on Disconnect Service

Message type: 10000

The heartbeat message informs the client connection monitoring service that this client login is active.

Table 192. Input parameters

Name	Type	Default value	Description
seq_number	i4	0	Sequence number of heartbeat (currently not used)

A client of COD (Cancel on Disconnect) service is should send heartbeat messages to the trading system not less than once per 10 second. If the user stays inactive (sends no messages to the trading system) within 20 seconds, all their orders will be automatically cancelled.

Note:

Only the COD service clients are obliged to send heartbeat messages.

The monitoring service does not send any replies on heartbeat messages. Please set flag value to '0' (no reply expected) when calling the heartbeat message sending function (cg_pub_post(pub, msgptr, 0).

Calling the 'cg_pub_post' function with the 'CG_PUB_NEEDREPLY' flag when sending a heartbeat will result in a 'FORTS_MSG100' error with return code: '10001 - Undefined message type'.

5.25. Method SetSmaPreTradeCheck - Enable pre-trade verification mode for SMA login orders

Message type: 406

Reply message type: 166

The command enables pre-trade verification mode for SMA login orders.

Table 193. Input parameters

Name	Type	Default value	Description
broker_code	c4	""	Brokerage Firm code
sma_asp	c20	""	SMA login ID
check_number	i1		Verification number (1-7)
base_contract_code	c25	""	Underlying asset code
instrument_type	i1	0	Instrument type: • 0 -Futures • 1 - Option • 2 - Calendar spread
client_code_check	c3	""	Client code under verification (filled in for verifications 6 and 7 only)
value	c29		Verification number

Table 194. Execution result

Name	Type	Default value	Description
code	i4		Return code
message	c255		Message text

Return codes:

0 operation completed successfully

Any other value error

Notes:

- The command is exclusively available to the logins obtained the appropriate right from the Trading Administrator.
- The ***'client_code_check'*** field should only be filled in for verifications 6 and 7. For other verifications it should be left blank.
- Below is the table containing verification number description for field ***'check_number'***:

Table 195. Verification numbers

Number	What to verify	Fields involved
1	Price fluctuation against the current price.	Field 'value' contains price value in order fluctuated against the current price value, in percent. Field 'sma_asp' contains an SMA login to be verified. Fields 'instrument_type' and/or 'base_contract_code' will be specified if there is a certain instrument/all instruments to be verified for a selected underlying asset.
2	Maximum volume of order, in contracts.	Field 'value' contains maximum number of contracts allowed in a single order. Field 'sma_asp' contains an SMA login to be verified. Fields 'instrument_type' and/or 'base_contract_code' will be specified if there is a certain instrument/all instruments to be verified for a selected underlying asset.
3	Allow/disallow negotiated mode	Field 'value' contains: ○ 0 - allow negotiated mode

Number	What to verify	Fields involved
		○ 1 - disallow negotiated mode.
4	Maximum volume of order, in RUR.	Field 'value' contains maximum amount in RUR allowed in a single order. Field 'sma_asp' contains an SMA login to be verified. Fields 'instrument_type' and/or 'base_contract_code' will be specified if there is a certain instrument/all instruments to be verified for a selected underlying asset.
5	Maximum number of orders (gross) allowed per trading day.	Field 'value' contains maximum number of orders (gross) allowed per trading day. Field 'sma_asp' contains an SMA login to be verified. Fields 'instrument_type' and/or 'base_contract_code' will be specified if there is a certain instrument/all instruments to be verified for a selected underlying asset.
6	Maximum number of contracts in a single buy order.	Field 'value' contains maximum number of contracts in a single buy order for trading member specified in field 'client_code_check'. Fields 'instrument_type' and/or 'base_contract_code' will be specified if there is a certain instrument/all instruments to be verified for a selected underlying asset.
7	Maximum number of contracts in a single sell order.	Field 'value' contains maximum number of contracts in a single sell order for trading member specified in field 'client_code_check'. Fields 'instrument_type' and/or 'base_contract_code' will be specified if there is a certain instrument/all instruments to be verified for a selected underlying asset.

5.26. Method DelSmaPreTradeCheck - Disable pre-trade verification mode for SMA login orders

Message type: 407

Return message type: 167

The command disables pre-trade verification mode for SMA login orders.

Table 196. Input parameters

Name	Type	Default value	Description
broker_code	c4	""	Brokerage Firm code
check_id	i8		

Table 197. Execution result

Name	Type	Default value	Description
code	i4		Return code
message	c255		Message text

Return codes:

0 operation completed successfully

other value error

Note:

The command is exclusively available to the logins obtained the appropriate right from the Trading Administrator.

5.27. Method UserKillSwitch - Disable transactions for trading member login

Message type: 408

Return message type: 168

The command disables trading transactions for a trading member login.

Table 198. Input parameters

Name	Type	Default value	Description
login	c20		Trading member's login to enable/disable trading transactions for.

Name	Type	Default value	Description
disable	i1		Transaction allowance settings for the login: • 0 - trading transactions are enabled for the login • 1 - trading transactions are disabled for the login.
cancel_orders	i1	0	Order cancellation settings if trading transactions are disabled for the login: • 0 - cancel orders • 1 - do not cancel orders.

Table 199. Output results

Name	Type	Default values	Description
code	i4		Return code
message	c255		Message text
num_orders	i4		Number of canceled orders

Return codes:

0 operation completed successfully

other value error

Notes:

The command is exclusively available to the logins obtained the appropriate right from the Trading Administrator.

Flag 'cancel_orders = 1 ' is only available if 'disable =1'.

5.28. Method SetBrokerFeeParamNextSession - Setting parameters for calculating the brokerage fee

Message type: 453

Reply message type: 183

The command allows to add, change and delete parameters that are used in calculating the brokerage fee for clients trades. Parameters can be set for an individual client and for the entire brokerage firm. The parameters set for the BF are used in the calculation for all of its clients. The command is available for the login CF and BF levels only, to which the Trading Administrator has set the necessary rights. The set parameters will be applied in the next trading session.

Table 200. Input parameters

Name	Type	Default value	Description
broker_code	c4		Brokerage Firm code
mode	i4		Mode
client_code	c3	""	Client code
lower_fee	c27		Minimum possible amount of brokerage fee per contract
upper_fee	c27		Maximum possible amount of brokerage fee per contract
multiplier	c27		Multiplier to the amount of exchange and clearing fees
additive	c27		Constant addition per contract

Table 201. Execution result

Name	Type	Default value	Description
code	i4		Return code
message	c255		Message text

Return codes:

0 operation completed successfully

Any other value error

Notes:

- Command work mode (the 'mode' field):

  1 Add / Edit

  2 Delete

- You have to specify the client code in the command ('client_code' field) if you set parameters for him only. If you set parameters for the entire BF, the 'client_code' field should be empty.
- Acceptable values for 'lower_fee' parameter from 0.00 to +100.00.
- Acceptable values for 'upper_fee' parameter from from 0.00 to +10 000.00.
- Acceptable values for 'multiplier' parameter from 0.00 to +100.00.
- Acceptable values for 'additive' parameter from 0.00 to +1 000.00.
- When adding (changing) client parameters ('mode=1' in the command), a new record with 'sess_id=-1' is added to the 'broker_fee_params' table. New parameters will be applied in the next trading session.
- When deleting parameters ('mode=2' in the command):
  - If the client has the parameters added today only (entry in the 'broker_fee_params' table with 'sess_id=-1'), then they are deleted from the table.
  - If the client has the current parameters only, then these parameters are marked for deletion in the 'broker_fee_params' table. For this, a new record with current parameters is added to the table and 'sess_id=-2' is set for it. But parameters will be deleted when the trading session changes.
  - If the client has both current and parameters added today, then the newly added parameters (entries with 'sess_id=-1') are deleted from the 'broker_fee_params' table, and the current parameters are marked for deletion (new record with 'sess_id=-2') and will be deleted when the trading session changes.

5.29. Method DelOrdersByBFLimit - Request to NCC for collateral sufficiency verification of Brokerage Firm

Message type: 419

Return message type: 172

The command is used by a Clearing Member to cancel orders in order to prevent insufficiency of collateral for a position. If the Brokerage Firm's Free limit amount is negative (FreeMoney < 0), all active orders of the given Brokerage Firm's clients will be cancelled.

Table 202. Input parameters

Name	Type	Default value	Description
broker_code	c4		Brokerage Firm code

Table 203. Execution result

Name	Type	Default value	Description
code	i4		Return code
message	c255		Message text
num_orders	i4		Number of cancelled orders

Return codes:

0 operation completed successfully

other value error

Note:

The command is available only to logins belonging to Clearing Firm/Brokerage Firm.

The command is eligible only for a Clearing Member which is not simultaneously a Trading Member.

5.30. Method ChangeClientNotUseNOVMode - Changing the special margining mode for premium options for a client

Message type: 479

Return message type: 193

The command enables or disables the special margining mode for premium options at the client level (without initial margin and NOV, with a prohibition on negative positions in premium options). Values set by the command take effect during the clearing session.

Table 204. Input parameters

Name	Type	Default value	Description
broker_code	c4		Brokerage Firm code
client_code	c3		Client code
not_use_nov	i1		Special mode for premium option margining for a client: • 0 - Disable • 1 - Enable
inherit_not_use_nov_from_broker	i1		Inherit special mode settings set at the Brokerage Firm level: • 0 - No • 1 - Yes

Table 205. Execution result

Name	Type	Default Value	Description
code	i4		Return code
message	c255		Message text

Return codes:

0 Success

other value Error

Note:

The special mode for premium option position accounting can be enabled for a client only if they have no negative positions in premium options and the number of contracts in sell orders does not exceed the position volume.

5.31. Method ChangeBFClientDefaultNotUseNOVMode - Changing the special margining mode for premium options for Brokerage Firm clients

Message type: 480

Return message type: 194

The command enables or disables the special margining mode for premium options (without Initial margin and NOV, with a prohibition on negative positions in premium options) for Brokerage Firm clients. Values set by the command take effect during the clearing session.

Table 206. Input parameters

Name	Type	Default Value	Description
broker_code	c4		Brokerage Firm code
code_bf	c2		Brokerage Firm code, for whose clients the special margining mode for premium options needs to be set
not_use_nov_client_default	i1		Special mode for premium option margining for Brokerage Firm clients: • 0 - Disable • 1 - Enable

Table 207. Execution result

Name	Type	Default Value	Description
code	i4		Return code
message	c255		Message text

Return codes:

0 Success

other value Error

Note:

The special mode for premium option position accounting can be enabled for Brokerage Firm clients only if all of them have no negative positions in premium options and the number of contracts in sell orders does not exceed the position volume.

5.32. Method ChangePassword - Change user password for the Trading System

Message type: 421

Return message type: 169

The command allows to change user password for the Trading System. The command requires a dedicated protocol 'p2mqpwd' provided with the CGate API.

Table 208. Input parameters

Name	Type	Default value	Description
old_pwd	c65		Current password
new_pwd	c65		New password

Table 209. Execution result

Name	Type	Default value	Description
code	i4		Return code
message	c255		Message text

Return codes:

0 Success

other value error

Note:

Once any error occurs during the password change procedure, the user password will not be changed.

PLAZA II data types

PLAZA II	C++	ODBC	Details
u1	UINT8	SMALLINT	Integer, size: 1 byte
u2	UINT16	INTEGER	Integer, size: 2 bytes
u4	UINT32	NUMERIC,10	Integer, size: 4 bytes
u8	UINT64	NUMERIC,20	Integer, size: 8 bytes
i1	INT8	SMALLINT	Integer with sign, size: 1 byte
i2	INT16	SMALLINT	Integer with sign, size: 2 bytes
i4	INT32	INTEGER	Integer with sign, size: 4 bytes
i8	INT64	BIGINT	Integer, size: 8 bytes
a	CHAR	VARCHAR,1	Symbol string, size: 1 byte.
cN	CHAR[N+1]	VARCHAR,N	Symbol string, ended with zero.
dN.M sN.M	P2BCDII	NUMERIC,N,M	Fixed-point decimal number coded in binary system, where: • N is the whole quantity of digits • M is quantity of digits in the fractional part
t	P2TIME	TIMESTAMP	Date and time.
f	DOUBLE	REAL	Double-precision number with floating point, size: 8 bytes.
bN		VARBINARY,N	Data unit.
zN		VARBINARY,N	Data unit., where the buffer lenght is set by the first 4 bytes.

Note:

Win1251 is used to encode symbol strings.

List of return codes

Return code	Description
-1	Error performing operation.
0	Operation successful.
1	User not found.
2	Brokerage Firm code not found.
3	Session inactive. No trading session assigned or trading for this instrument is currently inactive.
4	Session halted.
5	Error performing operation.
6	Insufficient rights to perform operation.
7	Cannot perform operation: incorrect Clearing Firm account.
8	Insufficient rights to perform order deletion.
9	Operations with orders are blocked for the firm by the Clearing Centre.
10	Insufficient funds to reserve.
12	Options premium exceeds the limit allowed.
13	Total amount of positions exceeds the market limit.
14	Order not found. Order number not found due to incorrect entry, execution, deletion, or move operation.
25	Unable to add order: prohibited by the Trading Administrator.
26	Unable to open position: prohibited by Trading Administrator. Trading Administrator has blocked the operation. Contact technical support for details.
27	Unable to open short position: prohibited by Trading Administrator.
28	Unable to perform operation: insufficient rights.
31	Matching order for the same account/ITN is not allowed. Cross-order submission attempted. Allowed only for negotiated orders, provided that both parties have the corresponding permission enabled at the client level. See "Cross-trades" in p2gate docs.
32	Trade price exceeds the limit allowed. Futures price range: limit_up/limit_down in 'fut_sess_contents' table, REFDATA stream. Options: up_prem/down_prem (up_prem_weekend/down_prem_weekend in addit. weekend trading session) in 'volat' table, VOLAT stream.
33	Operations with orders are blocked for this firm by the Clearing Administrator.
34	Cannot perform operation: wrong client code.
35	Invalid input parameters. Invalid order parameters: zero quantity/price, invalid client code length, or incorrect ISIN. If the cause of the error is unclear, contact technical support.
36	Cannot perform operation: wrong underlying.
37	Multi-leg orders cannot be moved.
38	Negotiated orders cannot be moved.
39	Price is not a multiple of the tick size.
40	Unable to add Negotiated order: counterparty not found.
41	User's trading rights have expired or are not valid yet.
42	Operations are prohibited by Chief Trader of Clearing Firm.
43	The 'Lock Mode' flag is not set by the Trading Administrator.
44	Clearing Firm's Chief Trader flag not found for this firm.
45	Unable to add Negotiated orders: no RTS code found for this firm.
46	Only Negotiated orders are allowed for this security.
47	There was no trading in this security during the session specified.
48	This security is being delivered. Only Negotiated orders from all Brokerage Firms within the same Clearing Firm are allowed.
49	Unable to add Negotiated order: a firm code must be specified.
50	Order not found. Order number not found due to incorrect entry, execution, deletion, or move operation.
53	Error setting input parameter - amount.
54	Unable to perform operation: exceeded operations quota for this client.
55	File not found.

Return code	Description
56	Unable to perform operations using this login/code pair: insufficient rights. Please contact the Trading Administrator.
57	Unable to connect to the Exchange server: insufficient rights. Please contact the Trading Administrator.
58	Unable to add orders without verifying client funds sufficiency: insufficient rights.
60	Auction halted for all risk-netting instruments.
61	Trading halted in all risk-netting instruments.
62	Trading halted on all floors of MOEX Derivatives Market.
63	Auction halted in all risk-netting instruments with this underlying.
64	Trading halted in all risk-netting instruments with this underlying.
65	Trading halted on all boards in all securities with this underlying.
66	Trading halted in this risk-netting instrument.
67	Unable to open positions in this risk-netting instrument: prohibited by the Trading Administrator.
68	Unable to add orders for all risk-netting instruments: prohibited by the Brokerage Firm.
69	Unable to add orders for all risk-netting instruments: prohibited by the Chief Trader.
70	Trading operation is not supported.
71	Position size exceeds the allowable limit.
72	Order is being moved.
73	Aggregated buy order quantity exceeds the allowable limit.
74	Aggregated sell order quantity exceeds the allowable limit.
75	Non-trading operation was unsuccessful due to timeout.
76	No record to delete.
77	No identification data for the specified trading account. Attempt to perform an operation using an inactive 7-digit client code.
78	Clearing Firm code not found.
79	Operations are prohibited by the Clearing Administrator.
80	Non trading operation is not supported.
81	Cannot perform operation: input validation error of data relevance.
82	The Book-or-Cancel order cannot be added/replaced because it leads to immediate execution.
85	Trading halted on all boards in all Futures, Calendar spreads and Daily futures contracts with auto-prolongation with this underlying.
90	Trading halted on all boards in all Options with this underlying.
95	Trading halted in Anonymous mode in all securities with this underlying.
100	Trading halted in Anonymous mode in all Futures, Calendar spreads and Daily futures contracts with auto-prolongation with this underlying.
105	Trading halted in Anonymous mode in all Options with this underlying.
110	Trading halted in Negotiated mode in all securities with this underlying.
115	Trading halted in Negotiated mode in all Futures, Calendar spreads and Daily futures contracts with auto-prolongation with this underlying.
120	Trading halted in Negotiated mode in all Options with this underlying.
125	Trading halted on the Securities floor of MOEX Derivatives Market.
130	Trading halted on the Commodity floor of MOEX Derivatives Market.
135	Trading halted on the Money floor of MOEX Derivatives Market.
140	Increasing position is prohibited in all instruments with this underlying.
310	Unable to add order: prohibited by Clearing Administrator.
311	Unable to open position: prohibited by Clearing Administrator.
312	Unable to open short position: prohibited by Clearing Administrator.
314	Unable to add orders in the client account: prohibited by the Trader. Broker has blocked the operation. Contact your broker for details.
315	Unable to open position in the client account: prohibited by the Trader. Broker has blocked the operation. Contact your broker for details.

Return code	Description
316	Unable to open short position in the client account: prohibited by the Trader. Broker has blocked the operation. Contact your broker for details.
317	Amount of buy/sell orders exceeds the limit allowed.
318	Unable to add order for the client account: client does not have a deposit account for settlement of Money Market securities. Prohibited by Clearing Administrator.
320	Amount of active orders exceeds the limit allowed for the client account for this security.
331	Insufficient funds in the Settlement Account.
332	Insufficient client funds. Increase client level limits or contact your broker.
333	Insufficient Brokerage Firm funds.
335	Unable to buy: amount of securities exceeds the limit set for the client.
336	Unable to buy: amount of securities exceeds the limit set for the Brokerage Firm.
337	Unable to sell: amount of securities exceeds the limit set for the client.
338	Unable to sell: amount of securities exceeds the limit set for the Brokerage Firm.
339	Collateral recalculation in progress.
680	Insufficient client funds.
681	Insufficient Clearing Firm funds.
682	Insufficient funds to increase position.
3000	Modification and cancellation of the quote is prohibited due to Speed bump.
3001	Operation is prohibited.
3002	The sell price must be more than the buy price for the quote.
3003	Negotiated orders by Settlement Accounts "Ин" are prohibited
4000	Invalid input parameters.
4001	Unable to perform operation: insufficient rights.
4002	Unable to change trading limit for the client: no active trading sessions.
4004	Unable to change trading limit for the client: client code not found.
4005	Unable to change the trading limit for the client: insufficient funds. Insufficient funds at broker or clearing firm level. Check the amount of free funds.
4006	Invalid input parameters: this "Operating mode" is not supported.
4007	Invalid input parameters: the "Funds limit" parameter is not a number.
4008	Invalid input parameters: the "Clients collateral ratio" parameter is not a number.
4009	Invalid input parameters: invalid value for "Clients collateral ratio" parameter.
4010	Invalid input parameters: invalid value for "Minus check flag" parameter.
4011	Invalid input parameters: invalid value for "Flag of automatic adjustment of the limit by the amount of income after clearing" parameter.
4012	Unable to set trading limit for the client: error performing operation.
4013	Unable to set trading limit for the client: error performing operation.
4014	Unable to change parameters: no active trading sessions.
4015	Unable to change parameters: client code not found.
4016	Unable to change parameters: underlying's code not found.
4017	Invalid input parameters: invalid value for "Funds limit" parameter.
4018	Collateral calculation parameters are being changed by the Trading Administrator.
4021	Unable to set requested amount of pledged funds for Clearing Firm: insufficient amount of free funds.
4022	Unable to set requested amount of funds for Clearing Firm: insufficient amount of free funds.
4023	Unable to change trading limit for the Brokerage Firm: no active trading sessions.
4024	Unable to change trading limit for the Brokerage Firm: the Brokerage Firm is not registered for trading.
4025	Unable to set requested amount of pledged funds for the Brokerage Firm: insufficient amount of free funds in the Clearing Firm.
4026	Unable to set requested amount of funds for the Brokerage Firm: insufficient amount of free funds in the balance of the Separate Account.

Return code	Description
4027	Unable to set requested amount of pledged funds for the Clearing Firm: insufficient amount of pledged funds in the balance of the Separate Account.
4028	Unable to set requested amount of funds for the Brokerage Firm: insufficient amount of free funds in the Clearing Firm.
4030	Unable to change parameters for the Brokerage Firm: no active sessions.
4031	Unable to change parameters for the Brokerage Firm: Brokerage Firm code not found.
4032	Unable to change parameters for the Brokerage Firm: underlying's code not found.
4033	Unable to change parameters for the Brokerage Firm: insufficient rights to trade this underlying.
4034	Transfer of pledged funds from the Separate account is prohibited.
4035	Transfer of collateral is prohibited.
4040	Unable to change Brokerage Firm limit on risk-netting: no active sessions.
4041	Unable to change Brokerage Firm limit on risk-netting: Brokerage Firm is not registered for trading.
4042	Unable to change Brokerage Firm limit on risk-netting: Brokerage Firm code not found.
4043	Unable to change Brokerage Firm limit on risk-netting: error performing operation.
4044	Unable to change Brokerage Firm limit on risk-netting: error performing operation.
4045	Unable to delete Brokerage Firm limit on risk-netting: error performing operation.
4046	Unable to remove Chief Trader's restriction on trading in risk-netting instruments: insufficient rights.
4050	Unable to process the exercise request: restricted by the Chief Trader.
4051	Unable to process the exercise request: restricted by the Brokerage Firm.
4052	Unable to process the exercise request: wrong client code and/or security.
4055	Unable to process the exercise request: request number not found.
4060	Unable to process the exercise request: insufficient rights.
4061	Unable to process the exercise request: deadline for submitting requests has passed.
4062	Unable to process the exercise request: client code not found.
4063	Unable to process the exercise request: request not found.
4064	Unable to process the exercise request: insufficient rights.
4065	Unable to process the exercise request: option contract not found.
4066	Unable to process the exercise request: request to disable automatic exercise may only be submitted on the option's expiration date.
4067	Unable to process the exercise request: requests for exercise/ refusal to exercise on Cash-settled European options are prohibited.
4069	Unable to process the exercise request: error performing operation.
4070	Unable to process the exercise request: insufficient amount of positions in the client account.
4090	No active sessions.
4091	Client code not found.
4092	Underlying's code not found.
4093	Futures contract not found.
4094	Futures contract does not match the selected underlying.
4095	Partial selection of futures contracts not accepted: underlying flag set 'For all'.
4096	Unable to remove restriction: no restriction set.
4097	Unable to remove: the Chief Trader's restriction cannot be removed by Brokerage Firm trader.
4098	Security not found in the current trading session. Instrument ID invalid; the instrument has expired or is not traded in the current session.
4099	Both securities must have the same underlying.
4100	Exercise date of the near leg of a multi-leg order must not be later than that of the far leg.
4101	Unable to make a multi-leg order: lots are different.
4102	No position to move.
4103	The FOK order has not been fully matched.
4104	Anonymous repo order must contain a repo type.
4105	Order containing a repo type is restricted in this multi-leg order.

Return code	Description
4106	Multi-leg orders can be added only on the Money Market.
4107	This procedure is not eligible for adding orders for multi-leg securities.
4108	Unable to trade risk-netting instruments in T0: insufficient rights.
4109	Rate/swap price is not a multiple of the tick size.
4110	The near leg price differs from the settlement price.
4111	The rate/swap price exceeds the limit allowed.
4112	Unable to set restrictions for multi-leg futures.
4115	Unable to transfer funds between Brokerage Firm accounts: no active sessions.
4116	Unable to transfer funds between Brokerage Firm accounts: the donor Brokerage Firm is not registered for trading.
4117	Unable to transfer funds between Brokerage Firms: the receiving Brokerage Firm is not registered for trading.
4118	Broker Firm does not have sufficient amount of free funds.
4119	Brokerage Firm does not have sufficient amount of collateral.
4122	Clearing Firm does not have sufficient amount of free funds.
4123	Brokerage Firm does not have sufficient amount of collateral.
4124	Brokerage Firm code not found.
4125	Unable to transfer funds between accounts of different Clearing Firms.
4126	Unable to transfer: error while transferring.
4127	Insufficient free funds in the Settlement Account.
4128	Brokerage firm does not have sufficient amount of free funds.
4129	Insufficient amount of free funds in the balance of the Separate Account.
4130	Clearing Firm does not have sufficient amount of free funds.
4131	Brokerage Firm code not found.
4132	Unable to withdraw: error in withdrawal logic.
4133	No requests to cancel.
4134	Brokerage Firm does not have sufficient amount of funds.
4135	Clearing firm does not have sufficient amount of funds.
4136	Prohibited to transfer pledged funds.
4137	Brokerage Firm does not have sufficient amount of pledged funds.
4138	Insufficient funds to withdraw from the Settlement Account.
4139	Insufficient free collateral in the Settlement Account.
4140	Unable to transfer: position not found.
4141	Unable to transfer: insufficient number of open positions.
4142	Cannot transfer positions from the client account to an account with a different ITN.
4143	Unable to transfer position: the Brokerage Firms specified belong to different Clearing Firms.
4144	Cannot transfer positions to 'XXYY000' Brokerage Firm account.
4145	Unable to transfer positions for the selected Brokerage Firm: restricted by the Trading Administrator.
4146	Transferring positions in the selected securities is prohibited.
4147	Option contract not found.
4148	Settlement Account does not have sufficient amount of pledged funds.
4149	Settlement Account does not have sufficient amount of funds.
4150	Unable to balance risk using specified futures instrument.
4151	Specified FX Market Firm code not found.
4152	Specified FX Market Settlement Account not found.
4153	Specified FX Market financial instrument not found.
4154	Unable to add request for FX Market: the required parameters are not registered in the system.
4155	Required Administrator login for adding a risk balancing request is not registered in the system.
4160	Unable to perform operation. To transfer funds between settlement accounts, you are required to apply to NCC.
4161	Withdrawal is prohibited. Settlement account is included in the Unified Collateral Pool.

Return code	Description
4162	Unable to perform operation. The Brokerage Firms must be of the same Settlement account.
4163	Unable to perform operation. To transfer funds for Brokerage Firm with virtual limit, you are required to apply to NCC.
4164	Unable to perform operation. It is prohibited to change settings for client accounts.
4165	Unable to perform operation. Only Clearing Firm logins are able to perform the operation.
4166	Incorrect combination of flag values.
4167	Settlement Account not found.
4169	Cannot perform operation: the operation is available for Clearing Firm/Brokerage Firm login only.
4170	Cannot perform operation: incorrect Brokerage Firm account.
4171	Cannot perform operation: incorrect client account.
4172	Cannot perform operation: insufficient rights for the Clearing Member.
4173	Cannot perform operation: insufficient rights for the Trading Member.
4174	GTD multileg order is canceled by trading system.
4175	The Clearing Member has the option to take into funds only on the Settlement Account.
4180	Input parameter error: comment field contains invalid characters.
4181	Attention: This base contact is included in the intercontract spread. The highest value of the guarantee collateral coefficient will be applied for the base contracts included in the intercontract spread for this broker firm.
4200	Cannot confirm request. Trading participant's MASTER login is not connected.
4201	Cannot confirm request. Price value in request exceeded the current price value.
4202	Cannot confirm request. Maximum number of contracts exceeded in request.
4203	Cannot confirm request. Negotiated mode is not allowed.
4204	Cannot confirm request. Maximum volume in Russian Ruble exceeded in request.
4205	Cannot confirm request. Amount in Russian Ruble exceeded total available amount in requests per trading day.
4206	Cannot confirm request. Number of buy orders exceeded maximum available number in position.
4207	Cannot confirm request. Number of sell orders exceeded maximum available number in position.
4208	Cannot confirm request. Total quantity of simultaneous restrictions on position size for different clearing register exceeded for given SMA login.
4220	Trading operations for user are prohibited.
4221	Unable to perform operation: Clearing Member and Trading Member represent the same entity.
4222	Unable to perform operation with orders: insufficient rights for Clearing Member.
4223	Unable to send request to NCC: insufficient rights for Trading Member.
4224	Unable to perform operation: insufficient rights for active MASTER logins.
4225	Clearing member is under liquidation netting process, all operations are prohibited.
4226	All trading operations are prohibited for this BF during the morning session, except for orders cancellation operations.
4227	The transaction requires qualified investor status.
4230	Orders will not be cancelled: collateral requirements are met for the Brokerage Firm.
4231	Settlement session is already active.
4232	Price limits are not set.
4233	Position must be non-negative for all premium options on this client account; the number of contracts in sell orders must not exceed the position volume.
4234	Position must be non-negative for all premium options on all client accounts of this brokerage firm; the number of contracts in sell orders must not exceed the position volume.
4258	Negotiated iceberg orders are prohibited.
4259	Change of only a single iceberg order is possible.
4260	The iceberg visible part size is less than the minimum acceptable value.
4261	The iceberg visible part size is more than the iceberg order volume.
4262	The random addition size is more than the maximum acceptable value.
4264	The random addition size is less than zero.
4266	Trading system administrator lock mode is set for Settlement Account.
4268	Iceberg order can be changed only at the price.

Return code	Description
4269	Expiration order date cannot be indicated in the negotiated order.
4280	Invalid input parameters: "Client Code" parameter was not specified.
4281	Invalid input parameters: invalid value for "Prohibit Type" parameter.
4282	Invalid input parameters: for the parameter "Operating mode" = 12, the "Prohibit mask" = 0 cannot be set.
4283	Invalid input parameters: invalid value for "Priority" parameter.
4284	Unable to set a prohibition. The prohibition with same parameters already exists.
4285	The Brockerage Firm has been prohibited from submitting orders during the additional weekend session.
4300	Move orders are prohibited in the opening auction.
4301	Invalid instrument in the opening auction.
4302	FOK and IOC orders are prohibited in the opening auction.
4303	Negotiated orders are prohibited in the opening auction.
4304	A cross-order was found in the opening auction.
4305	BOC orders are prohibited in the opening auction.
5052	Unable to process the exercise request for daily futures contract with automatic prolongation: wrong client code and/or security.
5055	Unable to process the exercise request for daily futures contract with automatic prolongation: request number not found.
5061	Unable to process the exercise request for daily futures contract with automatic prolongation: deadline for submitting requests has passed.
5062	Unable to process the exercise request for daily futures contract with automatic prolongation: client code not found.
5063	Unable to process the exercise request for daily futures contract with automatic prolongation: request not found.
5064	Unable to process the exercise request for daily futures contract with automatic prolongation: insufficient rights.
5065	Unable to process the exercise request for daily futures contract with automatic prolongation: futures contract not found.
5069	Unable to process the exercise request for daily futures contract with automatic prolongation: error performing operation.
5071	Unable to process the exercise request for daily futures contract with automatic prolongation: invalid value of the number of futures for exercising.
5072	Unable to process the exercise request for daily futures contract with automatic prolongation: it is not time to apply requests yet.
5073	Unable to process the exercise request for daily futures contract with automatic prolongation: invalid futures contract.
5074	Unable to process the exercise request for daily futures contract with automatic prolongation: there are no rights to cancel/change a request submitted by a higher-level login.
10000	System level error while processing message.
10001	Undefined message type.
10004	Invalid message type.
10005	MQ address is too large
10006	Error parsing message. Message cannot be parsed according to the structure defined for this message type. Check CGATE version and stream schema compatibility.
20000	General code for parameter validation errors.
20001	Client code is not specified.
20002	Client code is invalid.
20003	Brokerage Firm code is not specified.
20004	Brokerage Firm with this code not found.
20005	Clearing Firm code is not specified.
20006	Clearing Firm code is invalid.
20007	Settlement account (SA) is not specified.
20008	Invalid value for parameter "Operation mode" (Mode).
20009	Invalid value for parameter "Operation mode" (Regime).
20010	Invalid value for parameter "Number of instrument units" (Amount).
20011	Limit value is out of acceptable range.

Return code	Description
20012	Limit value is not a number.
20013	Value for "Expiration scenarios weight" is out of acceptable range.
20014	Invalid value for parameter "Expiration scenarios weight"
20015	Value for "Individual scenario weight coefficient" is out of acceptable range.
20016	Initial margin coefficient value is out of acceptable range.
20017	Field ‘match_ref’ is not specified.
20018	Invalid value for parameter "Order direction" (dir). Possible values: 1 — buy, 2 — sell.
20019	Invalid value for parameter "Order type" (type).
20101	Base contract code is not specified.
20102	No parameters specified for modification.
20103	Parameter "Initial margin coefficient for underlying contract" is not a number.
20104	Parameter "Initial margin coefficient for underlying contract" must be set from 0.00000 to 10.00000.
20105	Parameter "Number of clearings before expiration" must be set from 0 to 999.
20106	Parameter "Expiration scenarios weight for client in total initial margin" (exp_weight_client_default) is not a number.
20107	Parameter "Expiration scenarios weight for client in total initial margin" (exp_weight_client_default) must be set from 0.00 to 1.00.
20108	Parameter "Prohibition on futures discount usage" must be set to 0 or 1 only.
20109	Parameter "Calendar spread margining type" must be set to 3 or 4 only.
20110	Parameter "Expiration scenarios weight" is not a number.
20111	Parameter "Expiration scenarios weight" must be set from 0.00 to 1.00.
20112	Parameter "Total initial margin coefficient for Brokerage Firm" is not a number.
20113	Parameter "Total initial margin coefficient for Brokerage Firm" must be set from 0 to 10.
20114	Parameter "Checking initial margin sufficiency for Brokerage Firm during fund inflow/outflow" must be set to 0 or 1 only.
20115	Parameter "Brokerage Firm Trading limit correspondence to sum of funds on Brokerage Firm sections" must be set to 0 or 1 only.
20116	Parameter "Checking initial margin sufficiency for Brokerage Firm during order submission" must be set to 0 or 1 only.
20117	Parameter "Cross-Contract spread margining type" must be set to 3 or 4 only.
20129	Invalid value for parameter "Mode" (mode).
20130	Expiration order not processed. Error in parameters.
30000	Common code for authorization errors.
30001	Insufficient permissions for the login to perform the operation.
30002	Insufficient permissions for the login to perform the operation.
40000	Common code for business logic errors.
40001	Order number duplication on double change: order_id1 cannot equal order_id2.
40002	Translation error: Donor matches recipient.
40003	Instrument (isin_id) in order_id1 does not match the instrument in order_id2. Double order change is not allowed for different isin_id.
40004	On double order change, order numbers for different clients are specified. Orders placed by a single client must be specified.
40005	On double change of opposite-direction orders, the buy price cannot be greater than or equal to the sell price — a Cross-trade is possible.
40006	On order modification, the number of instrument units (amount) cannot be negative.
40007	Parameter match_ref is filled, but parameter broker_to is not. For a negotiated order, the counterparty code must be added; for an anonymous order, both parameters should be left empty.