



ФИНАМ

TRANSAQ XML Connector

(4.95 compatible)

ver. 1.67

Содержание

Изменения	4
1. Общее описание	8
1.1 Функция SendCommand	8
1.2 Функция SetCallback	8
1.3 Функция SetCallbackEx	9
1.4 Функция FreeMemory	9
1.5 Функция Initialize	9
1.6 Функция UnInitialize	9
1.7 Функция SetLogLevel	10
2. Форматы данных	11
2.1 Типы данных	11
2.2 Числовые значения	11
2.3 Дата и время	12
3. Список команд принимаемых коннектором	13
3.1 connect	13
3.2 disconnect	15
3.3 server_status	15
3.4 get_securities	15
3.5 subscribe	15
3.6 unsubscribe	16
3.7 get_history_data	16
3.8 neworder	17
3.9 newcondorder	19
3.10 newstoporder	20
3.10.1 Работа со стоп-заявками	21
3.10.2 Стоп-лосс	22
3.10.3 Тейк-профит	22
3.10.3.1 Простой TP	23
3.10.3.2 Следящий (trailing) TP	23
3.11 cancelorder	24
3.12 cancelstoporder	24
3.13 get_forts_position	24
3.14 get_client_limits	24
3.15 get_markets	24
3.16 get_servtime_difference	25
3.17 get_leverage_control	25
3.18 change_pass	25
3.19 subscribe_ticks	25
3.20 get_connector_version	26
3.21 get_securities_info	26
4. Список возможных сообщений, передаваемых колбек функции	27
4.1 Исторические данные	27
4.2 Состояние сервера	27
4.3 Клиентские счета	28

4.4 Доступные рынки	29
4.5 Информация о доступных периодах свечей	29
4.6 Список инструментов	30
4.7 Котировки по инструменту(ам)	32
4.8 Сделки по инструменту(ам)	34
4.9 Глубина рынка по инструменту(ам)	35
4.10 Заявка(и) клиента	36
4.11 Сделка(и) клиента	39
4.12 Позиции клиента	41
4.13 Лимиты клиента на срочном рынке	45
4.14 Режим кредитования	46
4.15 Возможность рыночных заявок	46
4.16 Максимальная покупка/продажа и плечо	47
4.17 Тиковые данные	47
4.18 Информация по инструменту	48
4.19 Возможные ошибки	49
5. Получение новостей в TXmlConnector	50
5.1 Спецификация команд	50
5.1.1 Получение идентификатора сервера	50
5.1.2 Получение заголовков более старых новостей	50
5.1.3 Получение тела новости	51
5.2 Список возможных сообщений, передаваемых callback функции	51
5.2.1 Идентификатор сервера	51
5.2.2 Заголовок новости	51
5.2.3 Тело новости	51
6. Приложения	52

Изменения

24.04.12 | Версия 1.67

- Исправлена ошибка открытия лога при смене дня;

05.04.12 | Версия 1.66

- Изменена система логирования;
- Добавлены интерфейсные функции Initialize, UnInitialize, SetLogLevel;
- Команда disconnect сделана синхронной;
- Модифицирована логика подключения к серверу.

09.02.12 | Версия 1.65

- В элементе <takeprofit> команды newstoporder поля spread и correction теперь необязательны;
- Добавлена проверка на наличие символа ';' в строке нового пароля в команде change_pass;

20.01.12 | Версия 1.64

- Модифицирован алгоритм выгрузки библиотеки;
- Модифицирован алгоритм команды disconnect: управление вызывающему коду не возвращается до момента завершения соединения с сервером;
- Добавлено необязательное поле <hidden> (скрытое количество) в команды neworder, newcondorder

30.12.11 | Версия 1.63

- Модифицирована структура ответа на запрос get_securities_info;
- В структуру <money_position> добавлен элемент <markets>;
- В структуру forts_money, forts_collaterals, spot_limit добавлены элементы <markets> и <shortname>;
- Добавлена функция SetCallbackEx;

19.12.11 | Версия 1.61

- Изменена внутренняя потоковая модель;
- Модифицированы алгоритмы обработки данных;
- Изменено логирование;

03.10.11 | Версия 1.59

- Устранена ошибка с лишним `server_status` в случае неудачной предыдущей попытки соединения;

22.09.11 | Версия 1.58

- Устранена ошибка с неизменностью значения `usecredit` в структуре `security`;
- В структуру `<quotation>` (структура `quotations`) добавлены следующие элементы:
 - `<volatility>`Волатильность`</ volatility >`
 - `<theoreticalprice>`Теоретическая цена`</ theoreticalprice >`
- Устранена ошибка, возникавшая при указании коррекции и защитного спреда в процентах (спред/коррекция должны быть числом);

24.08.11 | Версия 1.57

- Устранена ошибка при подключении через прокси;
- Устранена ошибка с появлением лишних данных в ответе (структура `<result>`) на выполнение запроса;
- При отправке команды `disconnect` до завершения стартапа (получения структуры `server_status`) выдается сообщение `<result success="false"><message>Connection is still in progress. Please wait for response.</message></result>`;

11.08.11 | Версия 1.56

- Исправлена ошибка с кавычками в `sec_info`;

25.07.11 | Версия 1.55

- Добавлен запрос на получение информации по инструменту `"get_securities_info"`;
- Исправлена ошибка с возвращением нулевого значения номера сервера;

18.07.11 | Версия 1.54

- В структуру `<quotation>` (структура `quotations`) добавлены следующие элементы:
 - `<biddeptht>`Совокупный спрос`</biddeptht>`
 - `<offerdepth>`Кол-во лотов на продажу по лучшей

цене</offerdepth>;

- Модифицированы алгоритмы завершения работы модуля;

23.06.11 | Версия 1.53

- Модифицированы алгоритмы обработки данных;

21.06.11 | Версия 1.52

- Исправлена ошибка в формировании brokerref для стоп-заявок;
- Повторная отправка команды connect теперь возможна только после завершения соединения с сервером (с помощью команды disconnect либо в случае ошибок, приводящих к разрыву соединения с сервером);

11.05.11 | Версия 1.51

- Исправлена ошибка с зависанием коннектора (переставали идти данные);
- Исправлена ошибка с неизвестным статусом в элементе <stoporder> структуры <orders>;

03.05.11 | Версия 1.50

- Переработана логика работы и взаимодействия внутренних потоков обработки данных для исключения ситуации когда при выставлении заявки структура <orders> приходила раньше чем результат (структура <result success="true" transactionid="id"/>) выполнения ф-ции SendCommand;

07.04.11 | Версия 1.49

- В заголовке новости в элементы <title> и <source> добавлен тэг CDATA;
- Исправлена ошибка с невозвращением структуры positions при выполнении некоторых команд;

25.03.11 | Версия 1.48

- В структуру security добавлен элемент «Стоимость пункта цены» (point_cost);
- Исправлена ошибка с возвращением пустого элемента <conditionvalue> при выставлении условных заявок;

21.03.11 | Версия 1.47

- Добавлена команда получения версии коннектора (`get_connector_version`);
- В команду `connect` добавлены элементы, определяющие таймауты на сессию работы с сервером и запрос соответственно: `session_timeout` и `request_timeout`;
- В ответную посылку `server_status` добавлены атрибуты `id` и `recover`;
- Исправлены ошибки в формате ответной посылки для команды `get_securities`;

1. Общее описание

TXmlConnector.dll – загружаемая библиотека функций, предназначенных для организации взаимодействия пользовательских приложений с сервером TRANSAQ от имени и с правами доступа зарегистрированного торгового клиента. (Далее по тексту - «Коннектор».)

Коннектор экспортирует следующие функции:

```
bool SetCallback(tcallback pCallback);  
bool SetCallbackEx(tcallbackEx pCallbackEx, void* userData);  
BYTE* SendCommand(BYTE* pData);  
bool FreeMemory(BYTE* pData);  
BYTE* Initialize(const BYTE* logPath, int logLevel);  
BYTE* UnInitialize();  
BYTE* SetLogLevel(int logLevel).
```

Тип tcallback объявлен следующим образом:

```
typedef bool (*tcallback)(BYTE* pData);  
typedef bool (*tcallbackEx)(BYTE* pData, void*);
```

Обмен данными осуществляется посредством текстовых сообщений, представленных в виде XML структур, описываемых ниже.

Сообщения формируются в кодировке UTF-8.

1.1 Функция SendCommand

Функция SendCommand служит для передачи команд Коннектору. SendCommand возвращает указатель на XML-сообщение, содержащее данные об успешности или не успешности выполнения команды. В случае успеха сообщение имеет следующий вид:

```
<result success="true"/>
```

В случае неуспешного выполнения команды сообщение будет иметь вид:

```
<result success="false">  
  <message>error message</message>  
</result>
```

1.2 Функция SetCallback

Функция SetCallback устанавливает callback функцию, которая будет принимать асинхронные ответы от Коннектора.

1.3 Функция SetCallbackEx

С помощью функции SetCallbackEx можно передать Коннектору указатель на данные пользователя (например, строку-идентификатор Коннектора, какой-либо пользовательский объект), который будет передаваться в callback функцию при каждом ее вызове.

1.4 Функция FreeMemory

Ответственность за высвобождение памяти, аллоцированной Коннектором для размещения данных, лежит на пользователе Коннектора. Для этих целей следует использовать функцию FreeMemory.

1.5 Функция Initialize

Функция Initialize служит для инициализации библиотеки.

Примечание: Данная функция должна быть выполнена перед началом работы с Коннектором, то есть до первого вызова функции SendCommand.

Данная функция в качестве аргументов принимает путь к папке (const BYTE* logPath), в которой будут созданы лог-файлы (XDF*.log, DSP*.txt, TS*.log), и уровень логирования (int logLevel).

Предусмотрено три уровня логирования, в соответствии с детализацией и размером лог-файла:

- Минимальный (logLevel = 1);
- Стандартный (logLevel = 2, рекомендуемый);
- Максимальный (logLevel = 3).

Функция Initialize может быть вызвана в процессе работы с Коннектором повторно для изменения директории и уровня логирования, но только в случае, когда библиотека остановлена, то есть была выполнена команда disconnect или соединение еще не было установлено.

1.6 Функция UnInitialize

Функция UnInitialize служит для очистки внутренних ресурсов, выделенных Коннектором.

Примечание: Данная функция должна быть выполнена перед выгрузкой библиотеки из памяти.

1.7 Функция SetLogLevel

Функция SetLogLevel позволяет изменять уровень логирования Коннектора без остановки библиотеки. Возможные значения параметра logLevel приведены в описании функции Initialize.

В случае успешного выполнения функций Initialize, UnInitialize, SetLogLevel возвращается 0, иначе – указатель на XML-сообщение об ошибке (память, на которую указывает данный указатель, подлежит очистке с помощью функции FreeMemory).

Примечание: Параметр logLevel в функциях Initialize и SetLogLevel определяет глубину расширенного логирования (запросов клиента и ответов коннектора).

2. Форматы данных

2.1 Типы данных

В настоящем руководстве для описания элементов структур используются следующие типы данных:

integer	– 32-х битное целое число;
integer64	– 64-х битное целое число;
double	– число двойной точности с плавающей точкой;
date	– дата и/или время (см. пункт 2.3).
string	– текстовое значение

Большинство параметров с типом string имеет длину, не превышающую 50 символов.

Исключения:

<sec_info><secname>	= 75 символов
<news_header><title>	= 128 символов
<news_body><text>	= Максимально возможное количество

Данные типа orderno и tradeno в биржевых системах ММББ и РТС имеют тип integer64.

2.2 Числовые значения

Разделителем между целой и дробной частями числовых значений является точка («.»).

Финансовые инструменты в Коннекторе идентифицируются кодами secid, которые назначаются сервером Transaq и неизменность которых гарантируется в течение всего времени текущего подключения к серверу. Справочник инструментов (массив структур securities, получаемый командой get_securities) позволяет установить соответствие кода secid постоянным идентификаторам инструментов, установленным Биржами.

2.3 Дата и время

Дата и время представляются в следующем формате:

"DD.ММ.YYYY hh:mm:ss", где

DD – день месяца начиная с 1 до 31

ММ – месяц в году, начиная с 1 до 12

YYYY – год

hh – час в сутках с 00 до 23

mm – минуты в часу с 00 до 59

ss – секунды в минуте 00 до 59

В некоторых случаях часть DD.ММ.YYYY может отсутствовать, это означает что время указано на текущую дату.

Например, при работе с коннектором 27 Января.2011 получение в какой-либо структуре элемента

```
<Time>12:15:31</Time>
```

эквивалентно получению элемента:

```
<Time>27.01.2010 12:15:31</Time>.
```

3. Список команд принимаемых коннектором

3.1 connect

Установить соединение с сервером:

```
<command id="connect">
  <login>user_id</login>
  <password>user_password</password>
  <host>server_ip</host>
  <port>server_port</port>
  <logsdirectory>logs_directory</logsdirectory>
  <loglevel>log_level</loglevel>
  <autopos>true/false</autopos>
  <notes_file>notes filename</notes_file>
  <proxy type="тип" addr="адрес" port="порт" login="логин"
  password="пароль"/>
  <rqdelay>Частота запросов к серверу</rqdelay>
  <session_timeout>Таймаут на сессию в
  секундах</session_timeout>
  <request_timeout>Таймаут на запрос в секундах
  </request_timeout>
</command>
```

Результатом является структура `server_status`.

Необязательный параметр `logsdirectory` позволяет переопределить путь для лог-файлов `DSP*.txt` и `TS*.log` (исключение составляет `XDF*.log`), указанный при инициализации библиотеки.

Адрес папки для логов должен обязательно заканчиваться на символ «\». Пример:

```
<logsdirectory>D:\Documents and Settings\user\Local Settings\Application
Data\TSLab\TSLab\Logs\</logsdirectory>
```

`Loglevel` может принимать значения 0, 1 и 2 в зависимости от требуемого уровня детализации логов. Значение 2 соответствует наивысшей степени детализации. `Loglevel` является не обязательным параметром, если он не указан, логирование не осуществляется.

Примечание: Параметр `loglevel` в команде `connect` определяет глубину логирования этапов работы и внутреннего состояния коннектора.

Параметр `autopos` указывает на необходимость автоматического запроса клиентских позиций на срочном рынке после каждой клиентской сделки.



Если `autopos` не указан, по умолчанию он принимается равным `true`. Использование `<autopos>false</autopos>` при активной торговле ускоряет взаимодействие с сервером.

Немедленным результатом на команду `connect` будет результат синтаксического разбора запроса. После этого в асинхронном режиме будет либо передано сообщение об ошибке подключения, либо будет предоставлена информация о доступных периодах исторической ценовой информации – свечей/баров (`candlekinds`) и рынках (`markets`). Когда все данные будут загружены, будет получена структура `<server_status connected="true"/>`.

Информация о финансовых инструментах (`securities`) может быть получена после появления соответствующей информации на сервере (после 9:50 мск).

Параметр `notes_file` задает имя файла, в котором будут храниться примечания к заявкам. Если не указать имя файла, по умолчанию будет использоваться файл `notes.xml`. Примечания хранятся в `xml`-формате.

Тэг `proxu` следует включать только тогда, когда есть необходимость соединиться через прокси-сервер. Аналогично атрибуты `login` и `password` используются только тогда, когда имеется необходимость авторизации на сервере. В противном случае не нужно указывать их вообще.

Типы прокси-сервера доступны следующие: `SOCKS4`, `SOCKS5`, `HTTP-CONNECT` (задаются именно в таком виде, с учетом регистра и дефиса).

`Rqdelay` задает частоту обращений Коннектора к серверу `Transaq` в миллисекундах. Минимальное значение — 300.

`Session_timeout` – интервал времени, в течении которого коннектор в случае ошибок связи будет выполнять попытки переподключения к серверу без повторной зачатки информации о доступных периодах исторической ценовой информации – свечах/барах (`candlekinds`), финансовых инструментах (`securities`) и рынках (`markets`). Если данный параметр не будет указан, используется значение по умолчанию равное 120 секундам.

`Request_timeout` - таймаут на выполнение запроса. Если данный параметр не будет указан, используется значение по умолчанию равное 20 секундам. Значение параметра `session_timeout` должно быть больше значения параметра `request_timeout`.

В случае успешного выполнения команды `connect` повторная посылка команды возможна только после отключения от сервера с помощью команды `disconnect`.

3.2 disconnect

Разорвать соединение.

```
<command id="disconnect"/>
```

Результатом является структура server_status.

В случае возникновения ошибок, приводящих к завершению подключения к серверу (приходит структура <server_status connected="false"/>), вызов команды disconnect перед командой connect не требуется.

3.3 server_status

Получить текущее состояние соединения с сервером.

```
<command id="server_status"/>
```

Результатом является структура "server_status" и "overnight status".

3.4 get_securities

Получить список доступных инструментов.

```
<command id="get_securities"/>
```

Результатом является структура "securities".

3.5 subscribe

Подписаться на получение котировок, сделок и глубины рынка (стакана) по одному или нескольким инструментам.

```
<command id="subscribe">
  <alltrades> - подписка на все сделки
    <secid>...</secid>
    <secid>...</secid>
  </alltrades>
  <quotations> - подписка на показатели торгов
    <secid>...</secid>
    <secid>...</secid>
  </quotations>
  <quotes> - подписка на изменения «стакана»
```

```
<secid>...</secid>  
<secid>...</secid>  
</quotes>  
</command>
```

Результатом работы является начало передач всех изменений по заданным бумагам путем передачи сообщений quotations, alltrades, quotes.

3.6 unsubscribe

Прекратить получение котировок, сделок и глубины рынка (стакана) по одному или нескольким инструментам.

```
<command id="unsubscribe">  
  <alltrades> - все сделки  
    <secid>...</secid>  
    <secid>...</secid>  
  </alltrades>  
  <quotations> - котировки  
    <secid>...</secid>  
    <secid>...</secid>  
  </quotations>  
  <quotes> - изменения стакана  
    <secid>...</secid>  
    <secid>...</secid>  
  </quotes>  
</command>
```

Результатом работы является прекращение передач всех изменений по заданным бумагам путем передачи сообщений quotations, alltrades, quotes.

3.7 get_history_data

Выдать последние N свечей заданного периода, по заданному инструменту.

```
<command id="gethistorydata" secid="..." period="идентификатор"  
count="количество свечей" reset="true/false"/>
```

Параметр reset="true" говорит, что нужно выдавать самые свежие данные, в противном случае будут выданы свечи в продолжение предыдущего запроса.

Результатом работы является структура вида candles.

Возможные значения для period присылаются при установке соединения с сервером в полях candlekinds (поле id).

3.8 neworder

Подать новую заявку на биржу

```

<command id="neworder">
  <secid>идентификатор бумаги</secid>
  <client>клиент</client>
  <price>цена</price>
  <hidden>скрытое количество в лотах</hidden>
  <quantity>количество в лотах</quantity>
  <buysell>B</buysell>
  ("B" - покупка, или "S" - продажа)
  <bymarket/>
  <brokerref>примечание</brokerref>
  (будет возвращено в составе структур order и trade)
  <unfilled>PutInQueue</unfilled>
  (другие возможные значения: CancelBalance, ImmOrCancel)
  <usecredit/>
  <nosplit/>
</command>

```

Usecredit, nosplit и bymarket должны быть заданы пустым тегом, либо отсутствовать вообще. При наличии тега bymarket, тег price игнорируется и может отсутствовать. Поле hidden является необязательным.

Так как в TC FORTS не предусмотрены заявки без цены, то рыночные заявки для фьючерсов эмулируются с помощью лимитированных следующим образом: заявки на покупку подаются по максимально возможной цене сессии, а заявки на продажу - по минимально возможной. Для таких заявок также автоматически устанавливается признак "Снять остаток".

Для опционов рыночные заявки не принимаются.

Значения unfilled:

PutInQueue: неисполненная часть заявки помещается в очередь заявок Биржи.

ImmOrCancel: сделки совершаются только в том случае, если заявка может быть удовлетворена полностью.

CancelBalance: неисполненная часть заявки снимается с торгов

Немедленным результатом запроса neworder будет структура <result>, сообщающая о том, была ли принята заявка сервером TRANSAQ (не Биржей!). В случае успеха результатом будет следующий тег:

```
<result success="true" transactionid="id"/>
```

Transactionid - это целочисленный номер, который однозначно идентифицирует транзакцию внутри текущего сеанса связи с сервером.

После того, как в результате транзакции на Бирже появится соответствующий объект (заявка), основным идентификатором этого объекта становится регистрационный номер Биржи.

Transactionid является также основным идентификатором условной заявки, являющейся результатом транзакции, до тех пор, пока заявка не будет выставлена на Биржу и приобретёт регистрационный номер Биржи.

После того, как заявка будет принята Биржей, в асинхронном режиме будет получена структура <order>, имеющая в составе данных ссылку на номер транзакции.

Иногда бывает так, что ордер приходит от Биржи только через какое-то время после того как Биржа примет транзакцию по выставлению заявки. В этом случае Коннектор сначала пришлет активный ордер с нулевым <orderno>, а после получения ордера от Биржи пришлет ордер с биржевым номером (и возможно с новым статусом, например "исполнен"). Из этого следует, что внешнее приложение должно интерпретировать активный ордер с нулевым <orderno> как "заявка выставляется на Биржу" (по аналогии с тем, что при выполнении cancelorder значение <status> cancelled и нулевое значение элемента <withdrawtime> означает, что заявка находится в процессе снятия).

Примечание: В условных заявках <orderno> будет нулевым до тех пор пока не исполнится условие (и заявка не будет отправлена на Биржу)!

Примечание: Brokerref позже будут возвращены в структурах order и trade. Сохраняются примечания локально на диске в файле notes.xml. При утрате этого файла, или при запуске dll из разных мест, примечания не сохраняются.

Примечание для FORTS: На рынке FORTS не доступны параметры usecredit и nosplit. Также для параметра unfilled не доступно значение ImmOrCancel. Для опционов также не доступны рыночные заявки.

3.9 newcondorder

Выставить новую условную заявку на Сервер TRANSAQ

```

<command id="newcondorder">
  <secid>идентификатор бумаги</secid>
  <client>клиент</client>
  <price>цена</price>
  <hidden>скрытое количество в лотах</hidden>
  <quantity>количество в лотах</quantity>
  <buysell>"B" - покупка, "S" - продажа</buysell>
  <bymarket/>
  <brokerref>примечание</brokerref>
  <cond_type>тип условия (см. ниже)</cond_type>
  <cond_value>значение</cond_value>
  <validafter></validafter>
  <validbefore></validbefore>
  <nosplit/>
  <usecredit/>
</command>

```

Validafter и validbefore задаются в форме даты, описанном выше. Для validafter можно передать значение "0", если заявка начинает действовать немедленно. Для validbefore значение "0" означает, что заявка будет действительна до конца сессии. Так же validbefore может принимать текстовое значение "till_canceled", которое говорит о том, что заявка будет актуальна пока ее не снимут.

Допустимые типы условия:

Bid	= лучшая цена покупки
BidOrLast	= лучшая цена покупки или сделка по заданной цене и выше
Ask	= лучшая цена продажи
AskOrLast	= лучшая цена продажи или сделка по заданной цене и ниже
Time	= время выставления заявки на Биржу
CovDown	= обеспеченность ниже заданной
CovUp	= обеспеченность выше заданной
LastUp	= сделка на рынке по заданной цене или выше
LastDown	= сделка на рынке по заданной цене или ниже

Принцип работы запроса newcondorder тот же, что и запроса neworder. После запроса возвращается либо сообщение об ошибке, либо структура <result> с номером транзакции. Заявка при этом не попадает сразу на биржу, а дожидается наступления условия. В отдаваемой асинхронно структуре order заявка будет иметь статус watching. Как только условие будет выполнено, заявка будет передана на биржу, в результате чего callback будет вызван еще раз со структурой order, которая отразит биржевой номер заявки и ее характеристики. Элемент <orderno> будет нулевым до тех пор пока не

исполнится условие (и заявка не будет отправлена на Биржу).

Поле hidden является необязательным.

3.10 newstoporder

Выставить стоп-заявку:

```

<command id="newstoporder">
  <secid>идентификатор бумаги</secid>
  <client>идентификатор клиента</client>
  <buysell>B/S</buysell>
  <linkedorderno>номер связанной заявки</linkedorderno>
  (при отсутствии тэга без привязки)
  <validfor>заявка действительно до</validfor>
  (не обязательно)
  <expdate>дата экспирации (только для ФОРТС)</expdate>
  (не обязательно)
  <stoploss>
    <activationprice>Цена активации</activationprice>
    <orderprice>Цена заявки</orderprice>
    <bymarket/> - Выставить заявку по рынку
    (в этом случае orderprice игнорируется)
    <quantity>Объем</quantity>
    <usecredit/> - использование кредита
    <guardtime>Защитное время</guardtime>
    (не обязательно)
    <brokerref>Примечание брокера</brokerref>
    (не обязательно)
  </stoploss>
  <takeprofit>
    <activationprice>Цена активации</activationprice>
    <quantity>Количество</quantity>
    <usecredit/> - использование кредита
    <guardtime>Защитное время</guardtime>
    (не обязательно)
    <brokerref>Примечание брокера</brokerref>
    (не обязательно)
    <correction>Коррекция</correction>
    <spread>Защитный спрэд</spread>
    <bymarket/>
  </takeprofit>
</command>

```

Защитный спрэд, объем quantity, цену заявки для stop loss и коррекцию можно задавать как в абсолютной величине, так и в процентах (от цены либо от позиции клиента по смыслу). Для задания процентов, достаточно поставить после числа символ '%', например:

```
<quantity>10%</quantity>
```

Существенно, для стоп-заявок brokerref не сохраняется локально в notes_file, и таким образом для стопов допустимы только короткие brokerref.

3.10.1 Работа со стоп-заявками

Стоп-заявка в системе TRANSAQ (или стоп) — это распоряжение, предписывающее серверу TRANSAQ выполнить в автоматизированном режиме транзакционные действия, реализующие логику стоп-лосс или тейк-профит в зависимости от ценовых условий на рынке.

Основное предназначение стоп-заявок — автоматизировать управление существующей или потенциальной позицией клиента, которое заключается в своевременном выполнении одного из следующих действий:

- либо при неблагоприятном изменении цены своевременно закрыть позицию, ограничив убытки заранее определённой величиной (исполнить стоп-лосс);
- либо зафиксировать прибыль после достижения ценой достаточно прибыльного уровня при возникновении признаков окончания благоприятного тренда (исполнить тейк-профит).

Стоп-заявки могут также с успехом применяться для автоматизированного открытия новой позиции при определённых условиях, например, при возникновении признаков смены тренда, или наоборот, для открытия позиции «по тренду».

Стоп-заявка состоит из двух частей:

- Стоп-лосс (далее SL)
- Тейк-профит (далее TP)

При выполнении условия для одной части стоп-заявки, вторая ее часть снимается.

Внимание! Вторая часть стоп-заявки снимается даже в том случае, если попытка выставить заявку на Биржу при исполнении оказалась неуспешной. Допускаются стоп-заявки, содержащие только одну часть (SL или TP).

Примечание 1. Для закрытия коротких позиций следует выставлять стопы на покупку, для закрытия длинных позиций - стопы на продажу.

Обе части стоп-заявки допускают использование признака «по рынку».

3.10.2 Стоп-лосс

SL предназначен для закрытия позиций с целью ограничения убытков от удержания позиции при неблагоприятном движении цены на рынке. Следовательно,

- для стопа на продажу часть SL активируется когда цена на рынке станет меньше либо равна цене активации стоп-лосса.
- для стопа на покупку часть SL активируется когда цена на рынке станет больше либо равна цене активации стоп-лосса.

При выставлении SL необходимо задать Цену активации и Цену заявки, которая будет отправлена на Биржу при достижении рынком цены активации.

Использование необязательного параметра Защитное время позволяет предотвратить исполнение стопа при «проколах» на рынке, т.е. в таких ситуациях, когда цены на рынке лишь кратковременно достигают уровня цены активации, и вскоре возвращаются обратно. В частности, такие явления наблюдаются при интервенциях или ошибках крупных участников, когда на рынке внезапно регистрируется значительное отклонение цен произошедших сделок, но реальной возможности совершить сделки по таким ценам на рынке не возникает.

Защитное время задаётся в целых секундах и отсчитывается от времени получения сервером первой сделки, удовлетворяющей цене активации. Если в течение защитного времени на рынке будут зафиксированы сделки, не подтверждающие наступление уровня цены активации, (т.е. сделки по цене выше цены активации для SL на продажу, либо по цене ниже цены активации для SL на покупку) то стоп-лосс возвращается в состояние «Ожидает активации».

Если защитное время не задано, то даже одна сделка на рынке по цене, удовлетворяющей цене активации, приводит к исполнению стоп-лосса.

3.10.3 Тейк-профит

TP предназначены для закрытия позиций с фиксацией прибыли. Следовательно,

- для стопа на продажу часть TP активируется когда цена на рынке станет больше либо равна цене активации.
- для стопа на покупку часть TP активируется когда цена на рынке станет меньше либо равна цене активации.

3.10.3.1 Простой TP

Для ввода простого TP достаточно задать Цену активации и Количество. В этом случае на Биржу будет отправлена заявка с ценой, равной цене первой же сделки на рынке, которая удовлетворяет цене активации.

Можно увеличить вероятность совершения сделки при исполнении стопа, задав Защитный спрэд. Для определения цены заявки, исполняющей TP на покупку, защитный спрэд прибавляется к цене рынка. Для определения цены заявки, исполняющей TP на продажу, защитный спрэд вычитается из цены рынка.

Для простого TP также можно установить защитное время (см. выше описание SL).

Примечание 2. При использовании защитного времени, в качестве цены рынка (от которой определяется цена заявки исполняющей TP) принимается цена последней по времени сделки, полученной сервером в течение защитного времени.

3.10.3.2 Следящий (trailing) TP

Позволяет выставить на Биржу заявку, закрывающую позицию в момент появления признаков окончания благоприятного тренда на рынке.

Для того, чтобы включить механизм отслеживания тренда, необходимо при выставлении TP задать значение поля Коррекция.

Это значение используется сервером TRANSAQ для определения момента окончания благоприятного тренда следующим образом:

- для TP на продажу считается, что растущий тренд заканчивается в тот момент, когда после того, как рынок вырос до уровня цены активации или выше, он снизится на величину коррекции от максимальной цены.
- для TP на покупку считается, что нисходящий тренд заканчивается в тот момент, когда после того, как рынок снизился до уровня цены активации или ниже, он вырастет на величину коррекции от минимальной цены.

Величина коррекции может быть задана как в виде абсолютного изменения цены, так и в виде процента от цены. Для следящего TP также могут быть заданы Защитный спрэд и/или Защитное время.

Если для следящего TP задано защитное время, то оно действует как при определении факта достижения цены активации, так и при определении факта окончания тренда (достижения заданного уровня коррекции).

3.11 cancelorder

Отменить заявку

```
<command id="cancelorder">  
  <transactionid>номер из структуры orders</transactionid>  
</command>
```

Результатом работы является отмена (снятие) заявки. Если произойдут какие-либо ошибки (заявка уже исполнена, уже снята, некорректный номер и т. д.), об этом будет сообщено в возвращаемом теге <result>.

3.12 cancelstoporder

Отменить стоп-заявку

```
<command id="cancelstoporder">  
  <transactionid>номер из структуры orders</transactionid>  
</command>
```

3.13 get_forts_position

Запрос позиций клиента по FORTS:

```
<command id="get_forts_positions" client="клиент"/>
```

Запрашивает у сервера позиции клиента на FORTS. Если не указать клиента, запрос будет выполнен по всем доступным клиентам.

3.14 get_client_limits

Запрос лимитов клиента.

```
<command id="get_client_limits" client="клиент" />
```

3.15 get_markets

Запросить список доступных рынков:

```
<command id="get_markets"/>
```

3.16 get_servtime_difference

```
<command id="get_servtime_difference"/>
```

Получить разницу между временем на компьютере пользователя и серверным временем. Формат ответа:

```
<result success="true" diff="кол-во секунд"/>
```

3.17 get_leverage_control

```
<command id="get_leverage_control" client="код клиента">  
  <secid>идентификатор бумаги</secid>  
  <secid>идентификатор бумаги</secid>  
  .....  
  <secid>идентификатор бумаги</secid>  
</command>
```

Запрос на получении информации по максимальным значениям покупки и продажи по перечисленным клиентам для заданного клиента. Так же будет возвращено плановое и фактическое плечо для заданного клиента.

3.18 change_pass

```
<command id="change_pass"  
  oldpass="текущий пароль"  
  newpass="новый пароль"  
>
```

Запрос на смену пароля. Выполняется синхронно.

Указываются действующий и новый желаемый пароль в качестве параметров. Результатом является сообщение об успешной смене пароля, либо сообщение об ошибке с указанием ошибки. На демо-сервере смена пароля запрещена, выдается сообщение: «Вам не разрешено изменять пароль».

3.19 subscribe_ticks

```
<command id="subscribe_ticks" filter="true/false">  
  <security secid="..." tradeno="..." />  
  <security secid="..." tradeno="..." />  
  ...  
</command>
```

Подписка на тиковые данные. С атрибутом filter="true" будут отдаваться только сделки нормального периода торгов. По умолчанию отдаются все сделки.

В tradeno указывается номер сделки, после которой надо начать отдавать тики по данному инструменту. Если 0, то с момента выполнения подписки. Чтобы получить все известные серверу тики, то можно задать заведомо малое значение tradeno, например, единицу.

Тики выдаются в хронологическом порядке вперемешку по инструментам. Каждый последующим запрос отменяет существующую подписку. Для того, чтобы отменить подписку вообще, необходимо сделать запрос с пустым списком инструментов.

3.20 get_connector_version

```
<command id = "get_connector_version"/>
```

Запрос на получение версии модуля XmlConnector. В ответ на данную команду внешнему приложению возвращается структура

```
<connector_version>Номер_версии_коннектора</connector_version>
```

3.21 get_securities_info

```
<command id = "get_securities_info">  
  <secid>идентификатор бумаги</secid>  
</command>
```

Запрос на получение информации по инструменту.

В ответ на данную команду внешнему приложению возвращается структура <sec_info>.

4. Список возможных сообщений, передаваемых CallBack функции

4.1 Исторические данные

```
<candles secid=".." period="идентификатор периода" status="..">
  <andle date="дата" open="..." high="..." low="..." close="..."
    volume="..." oi="open_interest"/>
</candles/>
```

secid	<i>:integer</i>	high	<i>:double</i>
period	<i>:integer</i>	low	<i>:double</i>
status	<i>:integer</i>	close	<i>:double</i>
date	<i>:date</i>	volume	<i>:integer</i>
open	<i>:double</i>	oi	<i>:integer</i>

Параметр "status" показывает, осталась ли еще история. Возможные значения:

- 0 - данных больше нет (дочерпали до дна)
- 1 - заказанное количество выдано (если надо еще - делать еще запрос)
- 2 - продолжение следует (будет еще порция)
- 3 - требуемые данные недоступны (есть смысл попробовать запросить позже).

Open interest передается только для фьючерсов и опционов.

4.2 Состояние сервера

```
<server_status id="ID сервера" connected="true/false/error"
  recover="true/атрибут отсутствует"/>
```

id	<i>:integer</i>
connected	<i>:string</i>
recover	<i>:string</i>

Данное сообщение передается после изменения состояния соединения с сервером, либо по запросу.

Значением атрибута id является идентификационный номер сервера, с которым в данный момент работает коннектор.

Атрибут recover – необязательный параметр. Присутствует только в случае, если коннектор пытается восстановить потерянное соединение с сервером

(находится в состоянии рекавера).

В случае нахождения коннектора в процессе рекавера клиенту будет передана структура вида:

```
<server_status id="ID сервера" connected="true" recover="true"/>.
```

В случае успешного рекавера будет передана структура вида:

```
<server_status id="ID сервера" connected="true"/>.
```

Если рекавер закончился неудачей, соединение с сервером будет разорвано, а структура server_status будет иметь вид:

```
<server_status id="ID сервера" connected="false"/>.
```

В случае состояния connected="error", внутри тега будет содержать сообщение об ошибке. Атрибуты id и recover будут отсутствовать. Например:

```
<server_status connected="error">Сервер недоступен</server_status>
```

4.3 Клиентские счета

```
<client id="CLIENT_ID" remove="true/false">
  <type>тип клиента</type>
  <currency>валюта фондового портфеля клиента</currency>
  <ml_intraday>дневной кредит</ml_intraday>
  <ml_overnight>овернайт кредит</ml_overnight>
  <ml_restrict>у.м. ограничительный</ml_restrict>
  <ml_call>у.м. требования</ml_call>
  <ml_close>у.м. закрытия</ml_close>
</client>
```

id	<i>:string</i>	ml_overnight	<i>:integer</i>
remove	<i>:string</i>	ml_restrict	<i>:double</i>
type	<i>:string</i>	ml_call	<i>:double</i>
currency	<i>:string</i>	ml_close	<i>:double</i>
ml_intraday	<i>:integer</i>		

Данные сообщения для каждого из клиентских счетов передаются сразу после коннекта. Кроме того, это сообщение передается в случае добавления/удаления администратором TRANSAQ доступа к клиентскому счету во время текущей сессии.

Возможные типы клиента: spot (кассовый), leverage (плечевой), margin_level (маржинальный).

Поля ml_overnight и ml_intraday присылаются только для плечевых клиентов. Выражаются целым числом и показывают величину плеча.

Поля ml_restrict, ml_call и ml_close присылаются только для маржинальных клиентов.

Параметр remove говорит о том, добавился ли клиент или удален. В случае удаления клиента никаких свойств клиента передано не будет.

Валюта фондового портфеля currency может принимать следующие значения: NA (если клиент не имеет фондового портфеля), RUB, EUR, USD.

4.4 Доступные рынки

```
<markets>  
  <market id="внутренний код рынка">название рынка</market>  
</markets>
```

id	<i>:integer</i>
название рынка	<i>:string</i>

4.5 Информация о доступных периодах свечей

```
<andlekinds>  
  <kind>  
    <id>идентификатор периода</id>  
    <period>количество секунд в периоде</period>  
    <name>наименование периода</name>  
  </kind>  
</andlekinds>
```

id	<i>:integer</i>
period	<i>:string</i>
name	<i>:string</i>

4.6 Список инструментов

```

<securities>
  <security secid="внутренний код" active="true/false">
    <seccode></seccode>
    <market>Идентификатор рынка</market>
    <shortname>Наименование бумаги</shortname>
    <decimals>Количество десятичных знаков в
цене</decimals>
    <minstep>Шаг цены</minstep>
    <lotsize>Размер лота</lotsize>
    <point_cost>Стоимость пункта цены</point_cost>
    <opmask usecredit="yes/no" bymarket="yes/no"
nosplit="yes/no" immorcancel="yes/no"
cancelbalance="yes/no"/>
    <sectype>Тип бумаги</sectype>
  </security>
</securities>

```

secid	<i>:integer</i>
active	<i>:string</i>
seccode	<i>:string</i>
market	<i>:integer</i> или <i>:string</i> (в случае ошибки)
shortname	<i>:string</i>
decimals	<i>:string</i>
minstep	<i>:double</i>
lotsize	<i>:integer</i>
point_cost	<i>:double</i>
usecredit	<i>:string</i>
bymarket	<i>:string</i>
nosplit	<i>:string</i>
immorcancel	<i>:string</i>
cancelbalance	<i>:string</i>
sectype	<i>:string</i>

Массив инструментов выдается автоматически после успешного подключения к серверу.

Может быть также получен впоследствии по запросу `get_securities`. Следует иметь в виду, что список инструментов может приходиться не единым блоком, а несколькими, а также в ходе сессии по мере подключения рынков, и динамического получения доступа к отдельным инструментам.

Secid действителен в течение сессии, постоянным уникальным ключом инструмента между сессиями является `Seccode+Market`.

Атрибут `active` указывает, является ли бумага торгуемой в обычном режиме.



Для неторгуемых бумаг не будут возвращаться поля `minstep`, `lotsize` и `ormask`.

В качестве `sectype` могут встречаться следующие поля:

Торгуемые инструменты:

- SHARE - акции
- BOND - облигации корпоративные
- FUT - фьючерсы FORTS
- OPT - опционы
- GKO - гос. бумаги
- FOB - фьючерсы ММББ

Неторгуемые (все кроме `IDX` приходят только с зарубежных площадок):

- IDX - индексы
- QUOTES - котировки (прочие)
- CURRENCY - валютные пары
- ADR - АДР
- NYSE - данные с NYSE
- METAL - металлы
- OIL - нефтянка

Так же:

- ERROR - в случае внутренней ошибки (не должно появляться)

`ormask` возвращается только для торгуемых инструментов (для которых `active="true"`), его параметрами являются параметры заявок (см. `neworder`).

Чтобы получить стоимость шага цены необходимо значение элемента `<point_cost>` умножить на значение элемента `<minstep>`. Полученное значение выражено в копейках. Для перевода значения в рубли необходимо его разделить на 100.

4.7 Котировки по инструменту(ам)

```

&ltquotations>
  &ltquotation secid ="внутренний код">
    &ltaccruedintvalue>НКД на дату торгов в расчете на одну
    бумагу, руб.</accruedintvalue>
    &ltopen>Цена первой сделки</open>
    &ltwaprice>Средневзвешенная цена</waprice>
    &ltbiddepth> Кол-во лотов на покупку по лучшей цене
    </biddepth>
    &ltbiddeptht>Совокупный спрос</biddeptht>
    &ltnumbids>Заявок на покупку</numbids>
    &ltofferdepth>Кол-во лотов на продажу по лучшей
    цене</offerdepth>
    &ltofferdeptht>Совокупное предложение</offerdeptht>
    &ltbid>Лучшая котировка на покупку</bid>
    &ltoffer>Лучшая котировка на продажу</offer>
    &ltnumoffers>Заявок на продажу</numoffers>
    &ltnumtrades>Сделок</numtrades>
    &ltvoltoday>Объем совершенных сделок в лотах</voltoday>
    &ltopenpositions>Общее количество открытых
    позиций(FORTS)</openpositions>
    &ltdeltapositions>Изм.открытых
    позиций(FORTS)</deltapositions>
    &ltlast>Цена последней сделки</last>
    &ltquantity>Объем последней сделки, в лотах.</quantity>
    &lttime>Время заключения последней сделки</time>
    &ltchange>Изменение цены последней сделки по отношению
    к цене последней сделки предыдущего торгового
    дня</change>
    &ltpriceminusprevwaprice>Цена последней сделки к оценке
    предыдущего дня</priceminusprevwaprice>
    &ltvaltoday>Объем совершенных сделок, руб.</valtoday>
    &ltyield>Доходность, по цене последней сделки</yield>
    &ltyieldatwaprice>Доходность по средневзвешенной
    цене</yieldatwaprice>
    &ltmarketpricetoday>Рыночная цена по результатам торгов
    сегодняшнего дня</marketpricetoday>
    &lthighbid>Наибольшая цена спроса в течение торговой
    сессии</highbid>
    &ltlowoffer>Наименьшая цена предложения в течение
    торговой сессии</lowoffer>
    &lthigh>Максимальная цена сделки</high>
    &ltlow>Минимальная цена сделки</low>
    &ltcloseprice>Цена закрытия</closeprice>
    &ltcloseyield>Доходность по цене закрытия</closeyield>
    &ltstatus>Статус «торговые операции
  </quotation>
</quotations>

```

```

        разрешены/запрещены»</status>
        <tradingstatus>Состояние торговой сессии по
        инструменту</tradingstatus>
        <buydeposit>ГО покупок/покр</buydeposit>
        <selldeposit>ГО продаж/непокр</selldeposit>
        <volatility>Волатильность </volatility >
        <theoreticalprice>Теоретическая цена</theoreticalprice>
    </quotation>
    <quotation secid ="внутренний код">
        ....
    </quotation>
</quotations>

```

secid	<i>:integer</i>
accruedintvalue	<i>:double</i>
open	<i>:double</i>
waprice	<i>:double</i>
biddepth	<i>:integer</i>
biddeptht	<i>:integer</i>
numbids	<i>:integer</i>
offerdepth	<i>:integer</i>
offerdeptht	<i>:integer</i>
bid	<i>:double</i>
offer	<i>:double</i>
numoffers	<i>:integer</i>
numtrades	<i>:integer</i>
voltoday	<i>:integer</i>
openpositions	<i>:integer</i>
deltapositions	<i>:integer</i>
last	<i>:double</i>
quantity	<i>:integer</i>
time	<i>:date</i>
change	<i>:double</i>
priceminusprevwaprice	<i>:double</i>
valtoday	<i>:double</i>
yield	<i>:double</i>
yieldatwaprice	<i>:double</i>
marketpricetoday	<i>:double</i>
highbid	<i>:double</i>
lowoffer	<i>:double</i>
high	<i>:double</i>
low	<i>:double</i>
closeprice	<i>:double</i>
closeyield	<i>:double</i>
status	<i>:string</i>
tradingstatus	<i>:string</i>
buydeposit	<i>:double</i>
selldeposit	<i>:double</i>
volatility	<i>:double</i>

theoreticalprice **:double**

Передается после подписки путем команды subscribe. В сообщении в первый раз передаются все релевантные инструменту поля, для которых имеются актуальные данные, впоследствии передаются только измененные поля.

4.8 Сделки по инструменту(ам)

```

<alltrades>
  <trade secid ="внутренний код">
    <tradeno>биржевой номер сделки</tradeno>
    <time>время сделки</time>
    <board> наименование борда </board>
    <price>цена сделки</price>
    <quantity>объем сделки</quantity>
    <buysell>покупка (B) / продажа (S)</buysell>
    <openinterest>...</openinterest>
    <period>Период торгов (O - открытие, N - торги, C -
      закрытие)</period>
  </trade>
  <trade secid ="внутренний код">
    ....
  </trade>
</alltrades>

```

secid	:integer
tradeno	:integer64
time	:date
board	:string
price	:double
quantity	:integer
buysell	:string
openinterest	:integer
period	:string

Передается после подписки путем команды subscribe. В сообщении могут быть переданы не все поля, а только те, у которых есть значения.

Информация по следкам РПС, РЕПО и сделкам по неполным лотам не передается.

Параметр open interest передается только для фьючерсов и опционов.

4.9 Глубина рынка по инструменту(ам)

```
<quotes>
  <quote secid ="внутренний код">
    <price>цена</price>
    <yield>доходность (актуально только для
    облигаций)</yield>
    <buy>количество бумаг к покупке</buy>
    <sell>количество бумаг к продаже</sell>
  </quote>
  <quote secid ="внутренний код">
    ....
  </quote>
</quotes>
```

secid	<i>:integer</i>
price	<i>:double</i>
yield	<i>:integer</i>
buy	<i>:integer</i>
sell	<i>:integer</i>

Передается после подписки путем команды subscribe. В сообщении передаются не все поля, а только те, у которых изменились значения. Если некоторая строка исчезла, то в полях buy и sell будет передано значение «-1».

4.10 Заявка(и) клиента

```
<orders>
  <order transactionid ="идентификатор транзакции сервера
  Transaq">
    <orderno>биржевой номер заявки</orderno>
    <secid>идентификатор бумаги</secid>
    <board>идентификатор борда</board>
    <client>идентификатор клиента</client>
    <status>статус заявки (см. ниже в таблице 3)</status>
    <buysell>покупка (B) / продажа (S)</buysell>
    <time>время регистрации заявки биржей</time>
    <accepttime>время регистрации заявки сервером Transaq
    (только для условных заявок)</accepttime>
    <brokerref>примечание</brokerref>
    <value>объем заявки в копейках</value>
    <accruedint>НКД</accruedint>
    <settlecode>Код поставки (значение биржи, определяющее
    правила расчетов - смотрите подробнее в документах
    биржи)</settlecode>
    <balance>Неудовлетворенный остаток объема заявки в лотах
    (контрактах)</balance>
    <price>Цена</price>
    <quantity>Количество</quantity>
    <hidden>Скрытое количество в лотах</hidden>
    <yield>Доходность</yield>
    <withdrawtime>Время снятия заявки, 0 для
    активных</withdrawtime>
    <condition>Условие, см. newcondorder</condition>
    <conditionvalue>Цена для условной заявки, либо
    обеспеченность в процентах</conditionvalue>
    <validafter>с какого момента времени действительна (см.
    newcondorder)</validafter>
    <validbefore>до какого момента действительно (см.
    newcondorder)</validbefore>
    <maxcomission>максимальная комиссия по сделкам
    заявки</maxcomission>
    <result>сообщение биржи в случае отказа выставить
    заявку</result>
  </order>
  <stoporder transactionid="идентификатор стопа">
    <activeorderno></activeorderno>
    <secid></secid>
    <board></board>
    <client></client>
    <buysell></buysell>
    <canceller></canceller>
    <alltradeno></alltradeno>
```

```

<validbefore></validbefore>
<author></author>
<accepttime></accepttime>
<linkedorderno></linkedorderno>
<expdate></expdate>
<status></status>
<stoploss usecredit="yes/no">
  <activationprice></activationprice>
  <guardtime></guardtime>
  <brokerref></brokerref>
  <quantity></quantity>
  <bymarket/>
  <orderprice></orderprice>
</stoploss>
<takeprofit>
  <activationprice></activationprice>
  <guardtime></guardtime>
  <brokerref></brokerref>
  <quantity></quantity>
  <extremum></extremum>
  <level></level>
  <correction></correction>
  <guardspread></guardspread>
</takeprofit>
</stoporder>
</orders>

```

<order>

transactionid	<i>:integer</i>
orderno	<i>:integer64</i>
secid	<i>:integer</i>
board	<i>:string</i>
client	<i>:string</i>
status	<i>:string</i>
buysell	<i>:string</i>
time	<i>:date</i>
accepttime	<i>:date</i>
brokerref	<i>:string</i>
value	<i>:double</i>
accruedint	<i>:double</i>
settlecode	<i>:string</i>
balance	<i>:integer</i>
price	<i>:double</i>
quantity	<i>:integer</i>
hidden	<i>:integer</i>
yield	<i>:double</i>
withdrawtime	<i>:date</i>
condition	<i>:string</i>
conditionvalue	<i>:double</i>

validafter	<i>:date</i>
validbefore	<i>:date</i>
maxcomission	<i>:integer</i>
result	<i>:string</i>
<stoporder>	
transactionid	<i>:integer</i>
activeorderno	<i>:integer64</i>
secid	<i>:integer</i>
board	<i>:string</i>
client	<i>:string</i>
buysell	<i>:string</i>
canceller	<i>:string</i>
alltradeno	<i>:integer64</i>
validbefore	<i>:date</i>
author	<i>:string</i>
accepttime	<i>:date</i>
linkedorderno	<i>:integer64</i>
expdate	<i>:date</i>
status	<i>:string</i>
<stoploss>	
usecredit	<i>:string</i>
activationprice	<i>:double</i>
guardtime	<i>:date</i>
brokerref	<i>:string</i>
quantity	<i>:integer</i> или <i>:double</i> (в случае %)
orderprice	<i>:double</i>
<takeprofit>	
activationprice	<i>:double</i>
guardtime	<i>:date</i>
brokerref	<i>:string</i>
quantity	<i>:integer</i> или <i>:double</i> (в случае %)
extremum	<i>:double</i>
level	<i>:double</i>
correction	<i>:double</i>
guardspread	<i>:double</i>

Передается автоматически после установки соединения, а так же после изменения какого-либо из полей. В последнем случае передаются только изменившиеся поля. В таких сообщениях для условных заявок идентификатором является transactionid, для всех остальных – orderno.

Transactionid - это идентификатор транзакции сервера Transaq, неизменность которого гарантируется внутри текущего сеанса связи с сервером. Он служит для идентификации условных заявок, а также заявок, отклоненных Биржей. После выставления заявки на биржу, ей присваивается параметр orderno - это постоянный идентификатор заявки, который не меняется вплоть до окончания биржевой торговой сессии.

Для стоп-заявок поля соответствуют аналогам при выставлении заявок.

Статусы стоп-заявок и обычных заявок различаются - смотрите их список в конце документации. Любое из полей может отсутствовать, если оно неактуально или не было задано при выставлении заявки.

Значение <status> cancelled и нулевое значение элемента <withdrawtime> означает, что заявка находится в процессе снятия.

Ненулевое значение <withdrawtime> при значении <status> cancelled означает, что заявка снята.

Alltradeno – биржевой регистрационный номер сделки, явившейся основанием для перехода стопа в текущее состояние.

Canceller – идентификатор трейдера, который отменил стоп.

Activeorderno – номер заявки Биржевой регистрационный номер заявки, выставленной на рынок в результате исполнения стопа.

Maxcomission – это максимальная комиссия, которая может быть взята по сделкам по этой заявке. Фактически показывает, сколько денег заблокировано под комиссию для данной заявки. Maxcomission будет ненулевым пока заявка активна и равен нулю для заявок ФОРТС и для ВСЕХ условных заявок.

4.11 Сделка(и) клиента

```
<trades>
  <trade>
    <secid>Идентификатор бумаги</secid>
    <tradeno>Номер сделки на бирже</tradeno>
    <orderno>Номер заявки на бирже</orderno>
    <board>Идентификатор борда</board>
    <client>Идентификатор клиента</client>
    <buysell>B - покупка, S - продажа</buysell>
    <time>время сделки</time>
    <brokerref>примечание</brokerref>
    <value>объем сделки</value>
    <comission>комиссия</comission>
    <price>цена</price>
    <quantity>количество лотов</quantity>
    <yield>доходность</yield>
    <accruedint>НКД</accruedint>
    <tradetype>тип сделки: 'T' – обычная 'N' – РПС 'R' – РЕПО
    'P' – размещение</tradetype>
    <settlecode>код поставки</settlecode>
    <currentpos>Текущая позиция по инструменту</currentpos>
  </trade>
</trades>
```

secid	<i>:integer</i>	commission	<i>:double</i>
tradeno	<i>:integer64</i>	price	<i>:double</i>
orderno	<i>:integer64</i>	quantity	<i>:integer</i>
board	<i>:string</i>	yield	<i>:double</i>
client	<i>:string</i>	accruedint	<i>:double</i>
buysell	<i>:string</i>	tradetype	<i>:string</i>
time	<i>:date</i>	settlecode	<i>:string</i>
brokerref	<i>:string</i>	currentpos	<i>:integer64</i>
value	<i>:double</i>		

Передается автоматически после установки соединения (для уже совершенных сделок), а так же по мере появления новых сделок.

4.12 Позиции клиента

```

<positions>
  <money_position>
    <markets>
      <market>Внутренний код рынка</market>
      ...
      <market>Внутренний код рынка</market>
    </markets>
    <asset>Код вида средств</asset>
    <client>Идентификатор клиента</client>
    <shortname>Наименование вида средств</shortname>
    <saldoin>Входящий остаток</saldoin>
    <bought>Куплено</bought>
    <sold>Продано</sold>
    <saldo>Текущее сальдо</saldo>
    <ordbuy>В заявках на покупку + комиссия</ordbuy>
    <ordbuycond>В условных заявках на покупку</ordbuycond>
    <comission>Сумма списанной комиссии</comission>
  </money_position>
  <sec_position>
    <secid>Код инструмента</secid>
    <client>Идентификатор клиента</client>
    <shortname>Наименование инструмента</shortname>
    <saldoin>Входящий остаток</saldoin>
    <saldomin>Неснижаемый остаток</saldomin>
    <bought>Куплено</bought>
    <sold>Продано</sold>
    <saldo>Текущее сальдо</saldo>
    <ordbuy>В заявках на покупку</ordbuy>
    <ordsell>В заявках на продажу</ordsell>
  </sec_position>
  <forts_position>
    <secid>Код инструмента</secid>
    <client>Идентификатор клиента</client>
    <startnet>Входящая позиция по инструменту</startnet>
    <openbuys>В заявках на покупку</openbuys>
    <opensells>В заявках на продажу</opensells>
    <totalnet>Текущая позиция по инструменту</totalnet>
    <todaybuy>Куплено</todaybuy>
    <todaybuy>Продано</todaybuy>
    <optmargin>Маржа для маржируемых опционов</optmargin>
    <varmargin>Вариационная маржа</varmargin>
    <expirationpos>Опционов в заявках на
    исполнение</expirationpos>
    <usedsellspotlimit>Объем использованного спот-лимита на
    продажу</usedsellspotlimit>
    <sellspotlimit>текущий спот-лимит на продажу,

```

```

        установленный Брокером</sellspotlimit>
        <netto>нетто-позиция по всем инструментам данного
        спота</netto>
        <kgo>коэффициент ГО для спота</kgo>
    </forts_position>
    <forts_money> - деньги ФОРТС
        <client>Идентификатор клиента</client>
        <markets>
            <market>Внутренний код рынка</market>
            ...
            <market>Внутренний код рынка</market>
        </markets>
        <shortname>Наименование вида средств</shortname>
        <current>Текущие</current>
        <blocked>Заблокировано</blocked>
        <free>Свободные</free>
        <varmargin>Опер. маржа</varmargin>
    </forts_money>
    <forts_collaterals> - залоги ФОРТС
        <client>Идентификатор клиента</client>
        <markets>
            <market>Внутренний код рынка</market>
            ...
            <market>Внутренний код рынка</market>
        </markets>
        <shortname>Наименование вида средств</shortname>
        <current>Текущие</current>
        <blocked>Заблокировано</blocked>
        <free>Свободно</free>
    </forts_collaterals>
    <spot_limit> - лимиты ФОРТС
        <client>Идентификатор клиента</client>
        <markets>
            <market>Внутренний код рынка</market>
            ...
            <market>Внутренний код рынка</market>
        </markets>
        <shortname>Наименование вида средств</shortname>
        <buylimit>Текущий лимит</buylimit>
        <buylimitused>Заблокировано лимита</buylimitused>
    </spot_limit>
</positions>

```

<money_position>

market	<i>:integer</i>
asset	<i>:string</i>
client	<i>:string</i>
shortname	<i>:string</i>
saldoin	<i>:double</i>

bought	<i>:double</i>
sold	<i>:double</i>
saldo	<i>:double</i>
ordbuy	<i>:double</i>
ordbuycond	<i>:double</i>
commission	<i>:double</i>
<sec_position>	
secid	<i>:integer</i>
client	<i>:string</i>
shortname	<i>:string</i>
saldoin	<i>:integer64</i>
saldomin	<i>:integer64</i>
bought	<i>:integer64</i>
sold	<i>:integer64</i>
saldo	<i>:integer64</i>
ordbuy	<i>:integer64</i>
ordsell	<i>:integer64</i>
<forts_position>	
secid	<i>:integer</i>
client	<i>:string</i>
startnet	<i>:integer</i>
openbuys	<i>:integer</i>
opensells	<i>:integer</i>
totalnet	<i>:integer</i>
todaybuy	<i>:integer</i>
todaysell	<i>:integer</i>
optmargin	<i>:double</i>
varmargin	<i>:double</i>
expirationpos	<i>:integer64</i>
usedsellspotlimit	<i>:double</i>
sellspotlimit	<i>:double</i>
netto	<i>:double</i>
kgo	<i>:double</i>
<forts_money>	
client	<i>:string</i>
market	<i>:integer</i>
shortname	<i>:string</i>
current	<i>:double</i>
blocked	<i>:double</i>
free	<i>:double</i>
varmargin	<i>:double</i>
<forts_collaterals>	
client	<i>:integer</i>
market	<i>:integer</i>
shortname	<i>:string</i>
current	<i>:double</i>
blocked	<i>:double</i>
free	<i>:double</i>

<spot_limit>	
client	<i>:integer</i>
market	<i>:integer</i>
shortname	<i>:string</i>
buylimit	<i>:double</i>
buylimitused	<i>:double</i>

Передается автоматически после установки соединения, а так же по мере появления новых позиций и изменении какого-либо из полей. В последнем случае передаются только изменившиеся поля.

Элемент `markets` в структуре `money_position` устанавливает зависимость между денежными позициями клиента и рынками, на которых их можно использовать.

`Forts_position` возвращается для фьючерсов, опционов, позиций по инструментам РТС Стандарт.

Теги `kgo`, `netto`, `sellspotlimit` и `usedsellspotlimit` отдаются только для позиций по инструментам являющимся «главными спотами» РТС Стандарт.

В `forts_position` возвращаются всегда все поля для любых инструментов. Поля, которые не имеют смысла для инструмента (например, `varmargin` для опционов), возвращаются с нулевым значением.

4.13 Лимиты клиента на срочном рынке

```
<clientlimits client="клиент">
  <cbplimit>стоимостной лимит открытых позиций (СЛОП срочн.
рынок ММВБ) </cbplimit>
  <cbplused>стоимостная оценка текущих чистых позиций (СОЧП
срочн. рынок ММВБ) </cbplused>
  <cbplplanned>СОЧП с учетом активных заявок (срочный рынок
ММВБ) </cbplplanned>
  <fob_varmargin>Вар. маржа срочного рынка
ММВБ</fob_varmargin>
  <coverage>Обеспеченность срочного портфеля
(FORTS)</coverage>
  <liquidity_c>Коэффициент ликвидности(FORTS)</liquidity_c>
  <profit>Доход(FORTS)</profit>
  <money_current>Деньги текущие. То же, что current в
forts_money</money_current >
  <money_blocked>Деньги заблокированные. То же, что blocked в
forts_money </money_blocked >
  <money_free>Деньги свободные. То же, что free в forts_money
</money_free >
  <options_premium>Премии по
опционам(FORTS)</options_premium>
  <exchange_fee>Биржевой сбор(FORTS)</exchange_fee>
  <forts_varmargin>Вар. маржа текущая (FORTS)</forts_varmargin>
  <varmargin>Операционная маржа. То же, что varmargin в
forts_money </varmargin>
  <pclmargin> Перечисленная в пром.клиринге вариационная
маржа(FORTS)</pclmargin>
  <options_vm> Вар. маржа по опционам(FORTS) </ options_vm >
  <spot_buy_limit>Лимит на покупку спот </spot_buy_limit>
  <used_stop_buy_limit> Лимит на покупку спот
использованный</used_spot_buy_limit>
  <collat_current>Залоги текущие. То же, что current в
forts_collaterals </collat_current>
  <collat_blocked>Залоги заблокированные. То же, что blocked в
forts_collaterals </collat_blocked>
  <collat_free> Залоги свободные. То же, free в forts_collaterals
</collat_free>
</clientlimits>
```

client	<i>:string</i>	options_premium	<i>:double</i>
cbplimit	<i>:double</i>	exchange_fee	<i>:double</i>
cbplused	<i>:double</i>	forts_varmargin	<i>:double</i>
cbplplanned	<i>:double</i>	varmargin	<i>:double</i>
fob_varmargin	<i>:double</i>	pclmargin	<i>:double</i>
coverage	<i>:double</i>	options_vm	<i>:double</i>
liquidity_c	<i>:double</i>	spot_buy_limit	<i>:double</i>
profit	<i>:double</i>	used_stop_buy_limit	<i>:double</i>
money_current	<i>:double</i>	collat_current	<i>:double</i>
money_blocked	<i>:double</i>	collat_blocked	<i>:double</i>
money_free	<i>:double</i>	collat_free	<i>:double</i>

4.14 Режим кредитования

```
<overnight status="true/false"/>
```

status ***:string***

Ночной или дневной режим кредитования (см. поля ml_intraday и ml_overnight в структуре client).

4.15 Возможность рыночных заявок

```
<marketord secid="id" permit="yes/no" />
```

secid ***:integer***
 permit ***:string***

Приходит в случае, если во время работы Коннектора изменится возможность выставления или снятия рыночной заявки по инструменту с идентификатором id.

4.16 Максимальная покупка/продажа и плечо

```

<leverage_control
  client="код клиента"
  leverage_plan="плановое плечо"
  leverage_fact="фактическое плечо">
  <security secid="код бумаги" maxbuy="макс.покупка"
  maxsell="мак.продажа" />
  <security secid="код бумаги" maxbuy="макс.покупка"
  maxsell="мак.продажа" />
  .....
  <security secid="код бумаги" maxbuy="макс.покупка"
  maxsell="мак.продажа" />
</leverage_control>

```

client	<i>:string</i>	secid	<i>:integer</i>
leverage_plan	<i>:double</i>	maxbuy	<i>:integer64</i>
leverage_fact	<i>:double</i>	maxsell	<i>:integer64</i>

4.17 Тиковые данные

```

<ticks>
  <tick>
    <secid>идентификатор бумаги</secid>
    <tradeno>номер сделки</tradeno>
    <tradetime>время сделки</tradetime>
    <price>цена</price>
    <quantity>количество лотов (контрактов)</quantity>
    <period>торговый период (О - открытие, N - торги, С -
    закрытие; передается только для ММВБ)</period>
    <buysell>В - покупка, S - продажа (с точки зрения того, кто
    инициировал сделку, приняв условия выставленной ранее
    заявки - передается только когда есть такая
    информация)</buysell>
    <openinterest></openinterest>
  </tick>
</ticks>

```

secid	<i>:integer</i>
tradeno	<i>:integer64</i>
tradetime	<i>:date</i>
price	<i>:double</i>
quantity	<i>:integer</i>
period	<i>:string</i>
buysell	<i>:string</i>
openinterest	<i>:integer</i>

4.18 Информация по инструменту

```

<sec_info sec_id="идентификатор бумаги">
  <secname> полное наименование инструмента</secname>
  <pname> единицы измерения цены</pname>
  <mat_date> дата погашения</mat_date>
  <clearing_price>цена последнего клиринга (только FORTS)
</clearing_price>
  <minprice>минимальная цена (только FORTS)</minprice>
  <maxprice>максимальная цена (только FORTS)</maxprice>
  <buy_deposit>ГО покупателя (фьючерсы FORTS,
руб.)</buy_deposit>
  <sell_deposit>ГО продавца (фьючерсы FORTS,
руб.)</sell_deposit>
  <bgo_c>ГО покрытой позиции (опционы FORTS, руб.)</bgo_c>
  <bgo_nc>ГО непокрытой позиции (опционы FORTS,
руб.)</bgo_nc>
  <accruedint>текущий НКД, руб</accruedint>
  <coupon_value>размер купона, руб</coupon_value>
  <coupon_date>дата погашения купона </coupon_date>
  <coupon_period>период выплаты купона, дни</coupon_period>
  <facevalue>номинал облигации или акции, руб</facevalue>
  <put_call>тип опциона Call(C)/Put(P)</put_call>
  <opt_type>маржинальный(M)/премия(P)</opt_type>
</sec_info>

```

sec_id	<i>:integer</i>	bgo_c	<i>:double</i>
secname	<i>:string</i>	bgo_nc	<i>:double</i>
pname	<i>:string</i>	accruedint	<i>:double</i>
mat_date	<i>:date</i>	coupon_value	<i>:double</i>
clearing_price	<i>:integer</i>	coupon_date	<i>:date</i>
price_min	<i>:integer</i>	coupon_period	<i>:integer</i>
price_max	<i>:integer</i>	facevalue	<i>:double</i>
buy_deposit	<i>:double</i>	put_call	<i>:string</i>
sell_deposit	<i>:double</i>	opt_type	<i>:string</i>

В структуру включаются только те элементы, значение которых отлично от нуля.

4.19 Возможные ошибки

В случае возникновения внутренних ошибок dll, они будут возвращены в callback-функцию в следующем виде:

```
<error>Описание ошибки</error>
```

```
error :string
```

5. Получение новостей в TXmlConnector

После подключения к серверу внешнее приложение будет автоматически получать заголовки свежих новостей в виде структур `news_header` (см. пункт **5.2.2**).

Для получения заголовков более старых новостей необходимо выполнять команду `"get_old_news"` (см. пункт **5.1.2**).

Каждая новость имеет уникальный целочисленный идентификатор. Более свежие новости имеют большие значения идентификаторов.

Чтобы получить тело новости для известного заголовка необходимо выполнить команду `"get_news_body"` (см. пункт **5.1.3**).

ВНИМАНИЕ! Команду `"get_news_body"` можно вызывать для получения только тех новостей, заголовки которых были получены внешним приложением в текущем сеансе подключения к серверу *Transaq*.

ВНИМАНИЕ! При реализации хранения новостей во внешнем приложении следует учитывать, что на каждом сервере *Transaq* ведется своя нумерация новостей. Для получения идентификатора сервера используйте команду `"get_server_id"`.

5.1 Спецификация команд

5.1.1 Получение идентификатора сервера

```
<command id = "get_server_id">
```

Запрос на получение идентификатора сервера. Выполняется синхронно. В ответ на данную команду внешнему приложению будет передано сообщение `"current_server"` (см. пункт **5.2.1**).

5.1.2 Получение заголовков более старых новостей

```
<command id = "get_old_news" count = "OLD_NEWS_COUNT"/>
```

Значение атрибута `"count"` определяет максимальное количество новостей, которое необходимо получить, и не может превышать 100.

В ответ на данную команду внешнему приложению будет отдано запрошенное количество заголовков новостей (структур `"news_header"`), если они существуют на сервере (см. пункт **5.2.2**).

5.1.3 Получение тела новости

```
<command id = "get_news_body" news_id = "NEWS_ID"/>
```

Значение атрибута " news_id " задает номер новости из полученного ранее заголовка, для которой необходимо получить тело. В ответ на данную команду внешнему приложению будет передано сообщение "news_body" (см. пункт **5.2.3**).

5.2 Список возможных сообщений, передаваемых callback функции

5.2.1 Идентификатор сервера

```
<current_server id = "SERVER_ID"/>
```

id ***:integer***

Здесь: id = "SERVER_ID" – идентификатор сервера, с которым установлено текущее соединение.

5.2.2 Заголовок новости

```
<news_header>
  <id> порядковый номер новости </id>
  <time_stamp> дата-время новости (от источника)</time_stamp>
  <source><![CDATA[ источник новости ]]></source>
  <title><![CDATA[ заголовок новости ]]></title>
</news_header >
```

id ***:integer***
time_stamp ***:date***
source ***:string***
title ***:string***

5.2.3 Тело новости

```
<news_body>
  <id> порядковый номер новости </id>
  <text><![CDATA[ текст новости в виде обычного текста
  ]]></text>
</news_body >
```

id ***:integer***
text ***:string***

6. Приложения

Таблица 1. Возможные значения для поля status в теге quotation.

Символ	Значение
A	Операции разрешены
S	Операции запрещены

Таблица 2. ММВБ/РТС. Возможные значения для tradingstatus в теге quotation.

Символ	Значение
N	Недоступно для торгов (ММВБ)
O	Период открытия (ММВБ)
C	Торги закрыты (ММВБ)
F	Период закрытия (ММВБ)
B	Перерыв (ММВБ)
T	Торговая сессия (ММВБ)
L	Период послеторгового аукциона (ММВБ)
0	Сессия назначена. Нельзя ставить заявки, но можно удалять (FORTS)
1	Сессия идет. Можно ставить и удалять заявки (FORTS)
2	Приостановка торгов по всем инструментам. Нельзя ставить заявки, но можно удалять (FORTS)
3	Сессия принудительно завершена. Нельзя ставить и удалять заявки (FORTS)
4	Сессия завершена по времени. Нельзя ставить и удалять заявки (FORTS)

Таблица 3. XETRA. Возможные значения для tradingstatus в теге quotation.

Символ	Значение
S	Start
g	Pre Trading
B	In Between Auction
G	Post Trading
A	Add
D	Delete
#	End Of Trade
T	Continuous Trading
V	Volatility Interruption (VOLA) of Continuous Trading
H	Halt
U	Suspend
q	Quote Driven Pre-Auction Call
Q	Quote Driven Auction Call
O	Opening Auction Call
I	Intra Day Auction Call
E	End-of-day Auction Call
K	Closing Auction Call
L	Opening Auction Pre-Orderbook Balancing
L	Intra Day Auction Orderbook Balancing
L	End-of-day Auction Pre-Orderbook Balancing
L	Closing Auction Pre-Orderbook Balancing
L	Opening Auction Orderbook Balancing
L	Intra Day Auction Orderbook Balancing
L	End-of-day Auction Orderbook Balancing
L	Closing Auction Orderbook Balancing
O	Opening Auction IPO Call
Z	Opening Auction IPO Freeze
I	Intra Day Auction IPO Call
Z	Intra Day Auction IPO Freeze
F	Equity Fast Market On
f	Equity Fast Market Off
N	Bond Fast Market On
n	Bond Fast Market Off
W	Warrant Fast Market On
w	Warrant Fast Market Off
I	Manual Intraday Auction Call
Z	Manual Intraday Auction Freeze
P	Continuous Auction Pre-Call
C	Continuous Auction Call
Z	Continuous Auction Freeze
@	Market reset in specific SetId

Таблица 4. Атрибуты заявки

Наименование	Описание
Bymarket / По рынку	Заявка по рынку. При наличии тега bymarket, тег price игнорируется и может отсутствовать.
Usecredit / Использовать кредит	Чтобы воспользоваться кредитом при заключении сделки, установите параметр «Использовать кредит» в поручении на выставление заявки.
Nosplit / По одной цене	При наличии встречных заявок с пересекающимися ценами, сделка совершается по цене и в объеме лучшей встречной заявки. Неисполненная часть подаваемой заявки помещается в очередь заявок с ценой, равной цене совершенной сделки.
PutInQueue / Поставить в очередь	При поступлении заявки в торговую систему Биржи производится проверка наличия в очереди встречных заявок, цены которых совпадают или пересекаются с ценой подаваемой заявки. При наличии таких заявок в системе производятся сделки по цене ранее поданной заявки. Затем количество лотов инструмента, указанное в заявке, уменьшается на количество лотов в совершенной сделке и производится аналогичная процедура сопоставления новой заявки с оставшимися встречными заявками. Неисполненная часть заявки помещается в очередь заявок.
ImmOrCancel / Немедленно или отклонить	Аналогично <i>Поставить в очередь</i> , но сделки совершаются только в том случае, если заявка может быть удовлетворена полностью. В противном случае заявка не выставляется
CancelBalance / Снять остаток	Аналогично <i>Поставить в очередь</i> , но неисполненная часть заявки снимается с торгов

Таблица 5. Статусы ордеров

Идентификатор	Значение
none	
active	Активная
cancelled	Снята трейдером (заявка уже попала на рынок и была отменена)
denied	Отклонена Брокером
disabled	Прекращена трейдером (условная заявка, которую сняли до наступления условия)
expired	Время действия истекло
failed	Не удалось выставить на биржу
forwarding	Выставляется на биржу
inactive	Статус не известен из-за проблем со связью с биржей
matched	Исполнена
refused	Отклонена контрагентом
rejected	Отклонена биржей
removed	Аннулирована биржей
wait	Не наступило время активации
watching	Ожидает наступления условия

Таблица 6. Статусы стоп-заявок

Идентификатор	Значение
cancelled	Снята трейдером (заявка уже попала на рынок и была отменена)
denied	Отклонена Брокером
disabled	Прекращена трейдером (стоп-заявка, которую сняли до наступления условия)
expired	Время действия истекло
failed	Не удалось выставить на биржу
linkwait	Ожидает исполнения связанной заявки
rejected	Отклонена биржей
sl_executed	Выполнена (Stop Loss)
sl_forwarding	Стокп выставляется на биржу (Stop Loss)
sl_guardtime	Стокп ожидает исполнения в защитном периоде
tp_correction	Ожидает исполнения в режиме коррекции (Take Profit)
tp_correction_guardtime	Стокп ожидает исполнения в защитном режиме после коррекции (Take Profit)
tp_executed	Выполнен (Take Profit)
tp_forwarding	Стокп выставляется на биржу (Take Profit)
tp_guardtime	Стокп ожидает исполнения в защитном периоде (Take Profit)
watching	Ожидает наступления условия

Таблица 7. Параметры торговых серверов Transaq

IP-адрес	Порт
<i>Тестовый сервер</i>	
78.41.194.72 (новый)	3939
<i>Торговый сервер</i>	
213.247.141.133	3900
89.249.28.197	3900