



Спецификация протокола TWIME для ОТС системы срочного рынка

версия 5.6.0

Москва 2022

Содержание

1. Введение	5
1.1. Назначение документа	5
1.2. Общее описание	5
1.3. Термины и определения	5
2. Презентационный уровень	6
2.1. Типы данных	6
2.1.1. Целочисленные типы	6
2.1.2. Десятичные	6
2.1.3. Строковые	6
2.1.4. Дата и время	6
2.1.5. Перечисления	7
2.1.6. Битовые маски	8
2.2. Заголовок	9
2.3. Схема сообщений	9
3. Сессионный уровень	10
3.1. Поддерживаемые сообщения	10
3.1.1. Establish (message id=5100)	10
3.1.2. EstablishmentAck (message id=5101)	10
3.1.3. EstablishmentReject (message id=5102)	10
3.1.4. Terminate (message id=5103)	11
3.1.5. RetransmitRequest (message id=5104)	11
3.1.6. Retransmission (message id=5105)	12
3.1.7. Sequence (message id=5106)	12
3.1.8. FloodReject (message id=5107)	12
3.1.9. SessionReject (message id=5108)	12
3.2. Сценарии сессионного взаимодействия	13
3.2.1. Привязка и завершение сессии	13
3.2.2. Мониторинг состояния сессии	13
3.2.3. Нумерация сообщений	14
3.2.4. Запрос повторной отправки сообщений	14
3.2.5. Восстановление сессии после кратковременного сбоя	14
3.2.6. Полное восстановление сессии после сбоя	14
3.2.7. Сброс номеров сообщений	15
3.3. Контроль аномальной активности	15
3.4. Отключение при переполнении буфера на стороне WireGate	16
4. Прикладной уровень	17
4.1. Поддерживаемые сообщения	17
4.1.1. NewStream (message id=8007)	17
4.1.2. CancelStream (message id=8008)	18
4.1.3. RfsQuote (message id=8009)	18
4.1.4. RfsQuoteMassCancel (message id=8011)	19
4.1.5. RfsQuoteHit (message id=8012)	19
4.1.6. RfsConfirmation (message id=8013)	20
4.1.7. NewStreamResponse (message id=9011)	20
4.1.8. NewStreamReject (message id=9012)	21
4.1.9. CancelStreamResponse (message id=9013)	21
4.1.10. CancelStreamReject (message id=9014)	22
4.1.11. RfsQuoteResponse (message id=9015)	22
4.1.12. RfsQuoteReplaceResponse (message id=9016)	23
4.1.13. RfsQuoteReject (message id=9017)	24
4.1.14. RfsQuoteCancelResponse (message id=9018)	24
4.1.15. RfsQuoteMassCancelAck (message id=9020)	25
4.1.16. RfsBestQuoteUpdate (message id=9021)	25
4.1.17. RfsQuoteHitAck (message id=9022)	25
4.1.18. RfsConfirmationAck (message id=9023)	25
4.1.19. RfsExecutionReport (message id=9024)	26
4.1.20. EmptyBook (message id=9009)	27
4.1.21. SystemEvent (message id=9010)	27
4.1.22. Непрошенные сообщения	27
4.2. Сценарии торгового взаимодействия	28
4.2.1. Создание потока ликвидности потребителем ликвидности LC	28
4.2.2. Выставление котировки в поток провайдером ликвидности LP1, обновление лучших котировок для LC	28
4.2.3. Выставление котировки в поток провайдером ликвидности LP2, обновление лучших котировок для LC	29
4.2.4. Изменение котировки провайдером ликвидности LP2	30
4.2.5. Удаление котировок провайдером ликвидности LP	30
4.2.6. Удаление котировки провайдера по истечении времени жизни	31
4.2.7. Закрытие потока ликвидности	31
4.2.7.1. Закрытие потока ликвидности по команде от потребителя ликвидности LC	32
4.2.7.2. Автоматическое закрытие потока ликвидности	32

4.2.8. Выставление котировки в поток потребителем ликвидности LC	33
4.2.9. Сведение котировок в сделку и опциональная процедура Last Look	33
4.2.9.1. Запрос от LC на исполнение котировки (Quote Hit) в поток ликвидности без Last Look	34
4.2.9.2. Ошибка при выставлении адресной заявки LP в торговую систему срочного рынка	35
4.2.9.3. Ошибка при выставлении адресной заявки LC в торговую систему срочного рынка	35
4.2.9.4. Подтверждение квазисделки провайдером - процедура Last Look	36
4.2.10. Уникальность пользовательских идентификаторов котировок QuoteMsgID	37
5. Схема сообщений	38

История изменений

Дата	Версия	Изменения
14.10.2022	5.6.0	<ol style="list-style-type: none"> В TerminationCodeEnum добавлено новое значение: "SequenceReset" – Сброс номеров сообщений. Актуализирована схема сообщений.
15.01.2020	3.6.0	<ol style="list-style-type: none"> Спецификация переименована в Спецификация протокола TWIME для OTC системы срочного рынка. Добавлен новый раздел, описывающий сообщения прикладного уровня для системы RFS - см. раздел 4. В схему сообщений внесены изменения, связанные с сообщениями системы RFS - см. раздел 5. Из сообщения ExecutionReport для IQS удалено поле AggressorIndicator - см. ???.
31.10.2019	3.5.0	<ol style="list-style-type: none"> В разделы "2.1.6. Битовые маски" и "5. Схема сообщений" добавлено описание новых битов для 'FlagsSet'. В разделе "4.1.6. ExecutionReport (message id=9008)" добавлены новые значения для поля Flags: <ul style="list-style-type: none"> 0x200000000000 – Активная сторона в сделке. Котировка, приведшая к сделке при добавлении в стакан 0x400000000000 – Пассивная сторона в сделке. Котировка из стакана, участвующая в сделке

1. Введение

1.1. Назначение документа

В данном документе представлено описание протокола TWIME для подключения к OTC системе срочного рынка. Целью спецификации является описание презентационного, сессионного и прикладного уровней протокола. В данную спецификацию не входят административные и технические аспекты организации сетевого подключения, а также способы обеспечения его безопасности.

1.2. Общее описание

OTC система - это дополнительный модуль торговой системы срочного рынка, представляющий собой сервис для привлечения крупноблочной ликвидности (RFS), который позволяет участникам подавать запрос на ликвидность, и в случае удовлетворительного предложения забирать предоставленную ликвидность. Более подробную информацию об OTC системе срочного рынка можно получить из документа "Шлюз Plaza-2 для OTC системы срочного рынка" **OTC_P2Gate_ru.pdf** [http://ftp.moex.com/pub/OTC/RFS_IQS/prod/OTCGate/Docs/OTC_P2Gate_ru.pdf].

1.3. Термины и определения

В рамках настоящего документа используются следующие термины, определения и сокращения:

Термин	Определение
RFS (Request For Stream)	Система RFS срочного рынка - аукцион по требованию
LC (Liquidity Consumer)	Потребитель ликвидности в системе RFS
LP (Liquidity Provider)	Провайдер ликвидности в системе RFS
Заявка	Обязывающее торговое поручение, подаваемое в торговое ядро ТС Спектра.
Индикативная котировка (или просто котировка)	Торговое поручение, подаваемое в OTC систему без проверки обеспечения.
Квазисделка	Сделка, получившаяся в результате сведения индикативных котировок в OTC системе.
Сделка	Сделка, совершенная в результате сведения заявок в ТС Спектра.

2. Презентационный уровень

Протокол презентационного уровня разработан на основе FIX Simple Binary Encoding (<https://www.fixtrading.org/standards/sbe>); предполагается, что пользователь уже знаком с основами этого протокола.

2.1. Типы данных

В протоколе используются следующие типы данных.

2.1.1. Целочисленные типы

```
<type name="Int8" maxValue="126" minValue="-128" nullValue="127"
  presence="optional" primitiveType="int8"
  description="Целочисленное знаковое размером 1 байт"/>

<type name="Int16" maxValue="32766" minValue="-32768" nullValue="32767"
  presence="optional" primitiveType="int16"
  description="Целочисленное знаковое размером 2 байта"/>

<type name="Int32" maxValue="2147483646" minValue="-2147483648"
  nullValue="2147483647" presence="optional" primitiveType="int32"
  description="Целочисленное знаковое размером 4 байта"/>

<type name="Int64" maxValue="9223372036854775806" minValue="-9223372036854775808"
  nullValue="9223372036854775807" presence="optional"
  primitiveType="int64" description="Целочисленное знаковое размером 8 байт"/>

<type name="UInt8" maxValue="254" minValue="0" nullValue="255"
  presence="optional" primitiveType="uint8"
  description="Целочисленное беззнаковое размером 1 байт"/>

<type name="UInt16" maxValue="65534" minValue="0" nullValue="65535"
  presence="optional" primitiveType="uint16"
  description="Целочисленное беззнаковое размером 2 байта"/>

<type name="UInt32" maxValue="4294967294" minValue="0"
  nullValue="4294967295" presence="optional" primitiveType="uint32"
  description="Целочисленное беззнаковое размером 4 байта"/>

<type name="UInt64" maxValue="18446744073709551614" minValue="0"
  nullValue="18446744073709551615" presence="optional"
  primitiveType="uint64" description="Целочисленное беззнаковое размером 8 байт"/>
```

2.1.2. Десятичные

```
<composite name="Decimal5" description="Decimal">
  <type name="mantissa" description="mantissa" minValue="-9999999999999999"
    maxValue="9999999999999999" primitiveType="int64" presence="required" />
  <type name="exponent" description="exponent" presence="constant"
    primitiveType="int8">-5</type>
</composite>
```

2.1.3. Строковые

Строка символов фиксированной длины.

```
<type name="String7" length="7" primitiveType="char"/>

<type name="String20" length="20" primitiveType="char"/>

<type name="String25" length="25" primitiveType="char"/>
```

2.1.4. Дата и время

```
<type name="DeltaMillisecs"
  primitiveType="uint32"
  minValue="1000"
  maxValue="60000"
  presence="required" />
<type name="TimeStamp"
  primitiveType="uint64"
```

```

minValue="0"
maxValue="18446744073709551614"
nullValue="18446744073709551615"
presence="optional"
description="Time in number of nanoseconds since Unix epoch, UTC timezone" />

```

2.1.5. Перечисления

```

<enum name="BooleanEnum" encodingType="uint8">
  <validValue name="False" spectra_value="0">0</validValue>
  <validValue name="True" spectra_value="1">1</validValue>
</enum>

<enum name="TerminationCodeEnum" encodingType="uint8">
  <validValue name="Finished" >0</validValue>
  <validValue name="UnspecifiedError" >1</validValue>
  <validValue name="ReRequestOutOfBounds" >2</validValue>
  <validValue name="ReRequestInProgress" >3</validValue>
  <validValue name="TooFastClient" >4</validValue>
  <validValue name="TooSlowClient" >5</validValue>
  <validValue name="MissedHeartbeat" >6</validValue>
  <validValue name="InvalidMessage" >7</validValue>
  <validValue name="TCPFailure" >8</validValue>
  <validValue name="InvalidSequenceNumber">9</validValue>
  <validValue name="ServerShutdown" >10</validValue>
  <validValue name="SequenceReset" >11</validValue>
</enum>

<enum name="EstablishmentRejectCodeEnum" encodingType="uint8">
  <validValue name="Unnegotiated" >0</validValue>
  <validValue name="AlreadyEstablished" >1</validValue>
  <validValue name="SessionBlocked" >2</validValue>
  <validValue name="KeepaliveInterval" >3</validValue>
  <validValue name="Credentials" >4</validValue>
  <validValue name="Unspecified" >5</validValue>
</enum>

<enum name="SessionRejectReasonEnum" encodingType="uint8">
  <validValue name="ValueIsIncorrect" >5</validValue>
  <validValue name="Other" >99</validValue>
  <validValue name="SystemIsUnavailable" >100</validValue>
  <validValue name="QuoteMsgIDIsNotUnique">101</validValue>
</enum>

<enum name="TimeInForceEnum" encodingType="uint8">
  <validValue name="Day" spectra_value="1">0</validValue>
  <validValue name="IOC" spectra_value="2">3</validValue>
</enum>

<enum name="SideEnum" encodingType="uint8">
  <validValue name="Unavailable" spectra_value="0">0</validValue>
  <validValue name="Buy" spectra_value="1">1</validValue>
  <validValue name="Sell" spectra_value="2">2</validValue>
  <validValue name="BothSides" spectra_value="3">89</validValue>
</enum>

<enum name="ModeEnum" encodingType="uint8">
  <validValue name="DontChangeQuoteQty" spectra_value="0">0</validValue>
  <validValue name="ChangeQuoteQty" spectra_value="1">1</validValue>
  <validValue name="CheckQuoteQtyAndCancelQuote" spectra_value="2">2</validValue>
  <validValue name="FixStyleReplace" spectra_value="3">3</validValue>
</enum>

<enum name="MatchTypeEnum" encodingType="uint8">
  <validValue name="AutoMatch" spectra_value="1">4</validValue>
  <validValue name="AutoMatchWithLastLook" spectra_value="0">10</validValue>
</enum>

<enum name="TradSesEventEnum" encodingType="uint16">
  <validValue name="SessionDataReady" spectra_value="1">101</validValue>
  <validValue name="IntradayClearingFinished" spectra_value="2">102</validValue>
  <validValue name="IntradayClearingStarted" spectra_value="4">104</validValue>

```

```

        <validValue name="ClearingStarted"          spectra_value="5">105</validValue>
        <validValue name="ExtensionOfLimitsFinished" spectra_value="6">106</validValue>
        <validValue name="BrokerRecalcFinished"    spectra_value="8">108</validValue>
        <validValue name="OtcSessionInited"       spectra_value="10000">10100</validValue>
        <validValue name="OtcSessionStarted"      spectra_value="10001">10101</validValue>
        <validValue name="OtcSessionSuspended"   spectra_value="10002">10102</validValue>
        <validValue name="OtcSessionStoped"      spectra_value="10003">10103</validValue>
        <validValue name="OtcSessionFinished"    spectra_value="10004">10104</validValue>
    </enum>

    <enum name="SecurityTypeEnum" encodingType="uint8">
        <validValue name="Future"                spectra_value="0">0</validValue>
        <validValue name="Option"                spectra_value="1">1</validValue>
        <validValue name="Multileg"              spectra_value="2">2</validValue>
    </enum>

    <enum name="StatusEnum" encodingType="uint8">
        <validValue name="Matched"               spectra_value="0">0</validValue>
        <validValue name="WaitConfirm"          spectra_value="2">1</validValue>
        <validValue name="Confirmed"            spectra_value="4">2</validValue>
        <validValue name="Failed"               spectra_value="3,10,11,12,13,14">3</validValue>
        <validValue name="Success"              spectra_value="9">4</validValue>
    </enum>

    <enum name="RejectReasonEnum" encodingType="uint8">
        <validValue name="NotApplicable"        spectra_value="0,2,4,9">0</validValue>
        <validValue name="NotConfirmed"         spectra_value="3">1</validValue>
        <validValue name="ActiveSideError"      spectra_value="10,12,13,14">2</validValue>
        <validValue name="PassiveSideError"     spectra_value="11,12,13,14">3</validValue>
    </enum>

    <enum name="StreamExposureDurationEnum" encodingType="uint8">
        <validValue name="NotApplicable"        spectra_value="0">0</validValue>
        <validValue name="Duration30sec"         spectra_value="1">1</validValue>
        <validValue name="Duration60sec"         spectra_value="2">2</validValue>
        <validValue name="Duration90sec"        spectra_value="3">3</validValue>
        <validValue name="Duration120sec"       spectra_value="4">4</validValue>
    </enum>

    <enum name="SpeedBumpTypeEnum" encodingType="uint8">
        <validValue name="NotApplicable"        spectra_value="0">0</validValue>
        <validValue name="Duration200ms"         spectra_value="1">1</validValue>
        <validValue name="Duration500ms"        spectra_value="2">2</validValue>
        <validValue name="Duration1000ms"       spectra_value="3">3</validValue>
        <validValue name="Duration3000ms"       spectra_value="4">4</validValue>
    </enum>

    <enum name="CancelReasonEnum" encodingType="uint8">
        <validValue name="Deal"                  spectra_value="1">1</validValue>
        <validValue name="LCDoesntHaveEnoughMoney" spectra_value="2">2</validValue>
        <validValue name="CancelByLC"           spectra_value="3">3</validValue>
        <validValue name="TimeOut"              spectra_value="4">4</validValue>
        <validValue name="CancelByAdministrator" spectra_value="5">5</validValue>
        <validValue name="EndOfTradingSession"  spectra_value="6">6</validValue>
    </enum>

```

2.1.6. Битовые маски

```

<set name="FlagsSet" encodingType="uint64">
    <choice name="Day"                description="Quotes: Day"
    <choice name="IOC"                description="Quotes: IOC"
    <choice name="Replace"            description="Quotes: The record results from replacing the quote"
    <choice name="Cancel"             description="Quotes: The record results from cancelling the quote"
    <choice name="MassCancel"         description="Quotes: The record results from mass cancelling"
    <choice name="MultiLeg"           description="Quotes: Multi leg"
    <choice name="FineOperation"      description="Quotes: Flag of cancelling the quote because of fine"
    <choice name="ActiveSide"         description="Trades: Quote initiator is aggressor"
    <choice name="PassiveSide"        description="Trades: Quote initiator is passive"
    <choice name="TimeOut"            description="Quotes: Delete quote due timeout"
    <choice name="AutoMatch"         description="Quotes and trades: Auto match without last look"
</set>

```



```
<set name="StreamFlagsSet" encodingType="uint64">
  <choice name="AutoMatch" description="Auto match without last look">0</choice>
  <choice name="ClosedStream" description="Stream is closed">1</choice>
</set>
```

2.2. Заголовок

WireGate-сообщение состоит из стандартного заголовка и тела сообщения. Поля заголовка должны предшествовать полям тела сообщения. Имя заголовка - "messageHeader". Стандартный заголовок сообщения содержит следующие поля:

```
<composite name="messageHeader">
  <type name="blockLength" primitiveType="uint16" description="Длина тела сообщения"/>
  <type name="templateId" primitiveType="uint16" description="Идентификатор сообщения"/>
  <type name="schemaId" primitiveType="uint16" description="Идентификатор схемы сообщений"/>
  <type name="version" primitiveType="uint16" description="Версия схемы сообщений"/>
</composite>
```

2.3. Схема сообщений

```
<?xml version="1.0" encoding="UTF-8"?>
<sbe:messageSchema package="sbe" byteOrder="littleEndian" id="20809" version="0">
</sbe:messageSchema>
```

Атрибуты схемы:

Атрибут	Описание	Значение
id	Уникальный идентификатор схемы.	
version	Версия схемы.	
package	Наименование или категория схемы	"sbe"
byteOrder	Порядок байтов в полях.	"littleEndian"

3. Сессионный уровень

На сессионном уровне обеспечивается идентификация сторон взаимодействия, гарантированная доставка и последовательная обработка сообщений, контроль состояния соединения и возможность восстановления сессии в случае сбоев. Протокол сессионного уровня разработан на основе FIXP (<https://www.fixtrading.org/standards/fixp>); предполагается, что пользователь уже знаком с основами этого протокола.

3.1. Поддерживаемые сообщения

- **Establish** - Иницирует привязку сессии к текущему TCP-соединению.
- **EstablishmentAck** - Подтверждает привязку сессии к TCP-соединению.
- **EstablishmentReject** - Извещает о том, что сессию не удалось привязать к TCP-соединению.
- **Terminate** - Завершение сессии.
- **RetransmitRequest** - Запрос повторной пересылки сообщений, начиная с указанного номера.
- **Retransmission** - Уведомление о повторной пересылке сообщений.
- **Sequence** - Используется для установки номера следующего сообщения. Также данное сообщение используется как Heartbeat.
- **FloodReject** - Присылается системой при превышении лимита на количество сообщений в единицу времени.
- **SessionReject** - Отправляется в ответ на невалидные сообщения от клиентов.

Ниже приведено описание полей сообщений. Для каждого поля указаны:

- **Tag** – уникальный идентификатор поля;
- **Поле** – имя поля;
- **Наличие** – признак, который показывает, является ли nullValue валидным значением для поля:
 - **Y** - обязательное поле, т.е. nullValue не используется;
 - **N** - опциональное поле, т.е. можно передавать nullValue;
 - **C** - поле заполняется не nullValue значением при определенном условии.
- **Тип** - тип поля;
- **Описание** - детальное описание поля.

3.1.1. Establish (message id=5100)

Сообщение иницирует привязку сессии к текущему TCP-соединению. Время ожидания сообщения Establish после установления TCP-соединения составляет 10 секунд, после которых TCP-соединение разрывается.

Tag	Поле	Наличие	Тип	Описание
<Header>		Y		
20204	Timestamp	Y	TimeStamp	Время отправки запроса.
20205	KeepaliveInterval	Y	DeltaMillisecs	Интервал передачи Heartbeat сообщений. Значение должно находиться в пределах от 1000 до 60000 миллисекунд включительно.
20206	Credentials	Y	String20	Идентификатор клиента (логин).

3.1.2. EstablishmentAck (message id=5101)

Сообщение подтверждает привязку сессии к TCP-соединению.

Tag	Поле	Наличие	Тип	Описание
<Header>		Y		
20207	RequestTimestamp	Y	TimeStamp	Время отправки из сообщения Establish.
20205	KeepaliveInterval	Y	DeltaMillisecs	Интервал передачи Heartbeat сообщений.
20208	NextSeqNo	Y	UInt64	Порядковый номер следующего сообщения.

3.1.3. EstablishmentReject (message id=5102)

Сообщение извещает о том, что сессию не удалось привязать к TCP-соединению.

Tag	Поле	Наличие	Тип	Описание
<Header>		Y		
20207	RequestTimestamp	Y	TimeStamp	Время отправки из сообщения Establish.
20209	EstablishmentRejectCode	Y	EstablishmentRejectCodeEnum	<p>Причина отклонения запроса:</p> <ul style="list-style-type: none"> "0" Unnegotiated - Зарезервировано в FIXP, но не может произойти в WireGate. "1" AlreadyEstablished - Соединение для этого клиента уже установлено. "2" SessionBlocked - Пользователь был заблокирован, следует обратиться в техническую поддержку за разъяснениями. "3" KeeraliveInterval - Интервал передачи Heartbeat сообщений находится вне допустимого диапазона. "4" Credentials - Либо такого пользователя нет, либо не с того ip-адреса. "5" Unspecified - Внутренняя ошибка сервера, следует обратиться в техническую поддержку.

3.1.4. Terminate (message id=5103)

Завершение сессии.

Tag	Поле	Наличие	Тип	Описание
<Header>		Y		
20210	TerminationCode	Y	TerminationCodeEnum	<p>Причина прерывания сессии:</p> <ul style="list-style-type: none"> "0" Finished - Сессия завершена по запросу клиента. "1" UnspecifiedError - Внутренняя ошибка сервера, следует обратиться в техническую поддержку. "2" ReRequestOutOfBounds - В ответ на RetransmitRequest. Запрашиваемые сообщения не могут быть пересланы. "3" ReRequestInProgress - Запрос RetransmitRequest на пересылку еще не завершился. "4" TooFastClient - Клиент присылает слишком много сообщений. Значительно превосходя свою квоту. "5" TooSlowClient - Клиент не вычитывает сообщения из TCP сокета. И количество таких сообщений значительно. "6" MissedHeartbeat - От клиента уже не приходят сообщения за период более KeeraliveInterval. "7" InvalidMessage - Либо сообщение нарушает протокол, либо не возможно распознать полученное сообщение. "8" TCPFailure - Ошибка на транспортном уровне. "9" InvalidSequenceNumber - Посылается в ответ на сообщение Sequence с некорректным SequenceNumber. "10" ServerShutdown - Сервер штатно завершает работу. "11" SequenceReset - Сброс номеров сообщений.

3.1.5. RetransmitRequest (message id=5104)

Запрос повторной пересылки сообщений в количестве Count, начиная с FromSeqNo.

Tag	Поле	Наличие	Тип	Описание
<Header>		Y		
20204	Timestamp	Y	TimeStamp	Время отправки запроса.
20211	FromSeqNo	Y	UInt64	Порядковый номер первого запрашиваемого сообщения.
20212	Count	Y	UInt32	Количество запрашиваемых сообщений.

3.1.6. Retransmission (message id=5105)

Сообщение уведомляет, что последующие сообщения в количестве Count - это пересылаемые сообщения в ответ на RetransmitRequest (message id=5104).

Tag	Поле	Наличие	Тип	Описание
<Header>		Y		
20208	NextSeqNo	Y	UInt64	Порядковый номер первого сообщения из сообщений, пересылаемых в ответ на RetransmitRequest.
20207	RequestTimestamp	Y	TimeStamp	Время отправки из сообщения RetransmitRequest.
20212	Count	Y	UInt32	Количество сообщений.

3.1.7. Sequence (message id=5106)

Используется только в качестве Heartbeat-сообщения. Клиенту следует отправлять на WireGate значение nullValue в поле NextSeqNo. WireGate отправляет клиенту в поле NextSeqNo номер следующего сообщения прикладного уровня.

Tag	Поле	Наличие	Тип	Описание
<Header>		Y		
20208	NextSeqNo	N	UInt64	Порядковый номер следующего сообщения.

3.1.8. FloodReject (message id=5107)

При превышении лимита сообщений система контроля аномальной активности посылает пользователю сообщение с уведомлением об отказе в обслуживании.

Tag	Поле	Наличие	Тип	Описание
<Header>		Y		
1166	QuoteMsgID	Y	UInt64	Пользовательский идентификатор отвергнутой котировки.
20213	QueueSize	Y	UInt32	Количество полученных от клиента сообщений, уровня приложения, за последнюю секунду.
20214	PenaltyRemain	Y	UInt32	Время в микросекундах, по прошествии которого будет успешно принято следующее сообщение.

3.1.9. SessionReject (message id=5108)

Отправляется в ответ на невалидные сообщения от клиентов.

Tag	Поле	Наличие	Тип	Описание
<Header>		Y		
1166	QuoteMsgID	Y	UInt64	Пользовательский идентификатор отвергнутой котировки.
371	RefTagID	N	UInt32	Идентификатор некорректного поля.
373	SessionRejectReason	Y	SessionRejectReasonEnum	Код причины отклонения сообщения: <ul style="list-style-type: none"> "5" ValueIsIncorrect - Некорректное значение поля. "99" Other - Иное. "100" SystemIsUnavailable - Торговая система не доступна. "101" QuoteMsgIDIsNotUnique - Не уникальный пользовательский идентификатор котировки.

3.2. Сценарии сессионного взаимодействия

3.2.1. Привязка и завершение сессии

Для привязки сессии к TCP-соединению клиент должен послать сообщение Establish. Если сообщение Establish корректное и пользователь авторизован, WireGate отправляет клиенту ответное сообщение EstablishmentAck, которое подтверждает привязку сессии. Если сообщение Establish не корректное или Система не авторизовала пользователя, WireGate отправляет клиенту сообщение EstablishmentReject с указанием причины отклонения запроса.

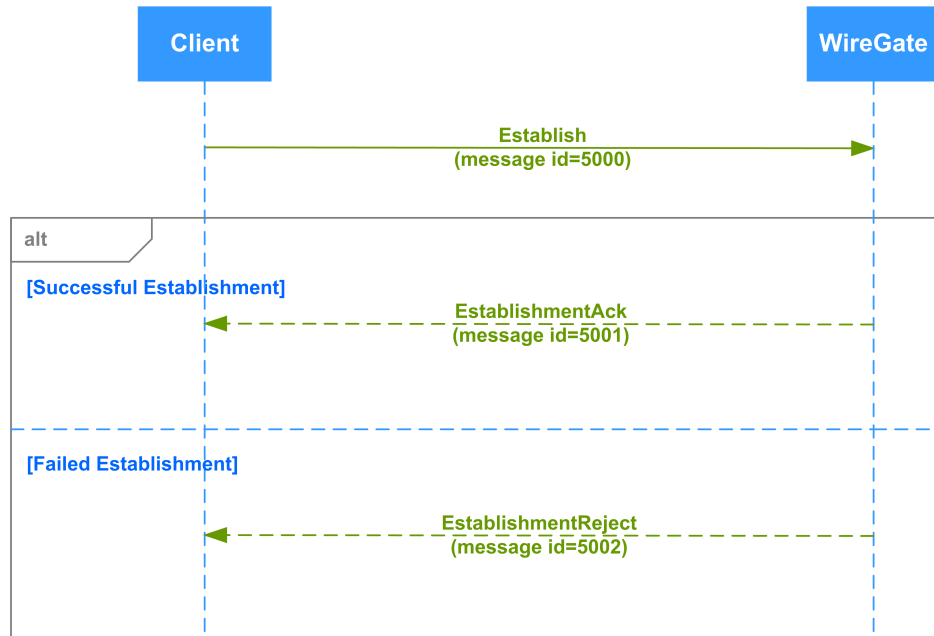


Рис. 1. Диаграмма. Привязка сессии

Для завершения сессии клиент должен отправить сообщение Terminate и дождаться от WireGate ответного сообщения Terminate с TerminationCode=0 (Finished).

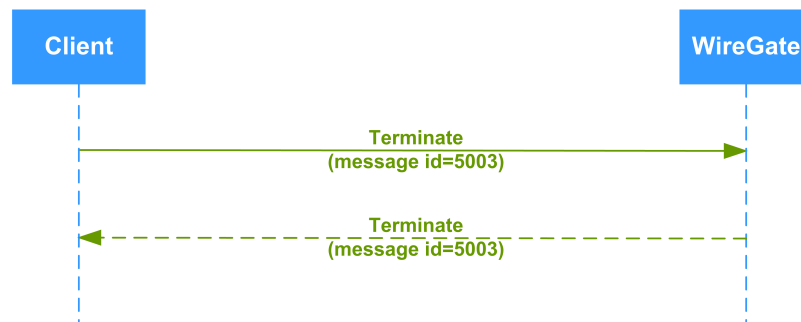


Рис. 2. Диаграмма. Завершение сессии

В случае переподключения клиента к WireGate повторное TCP-соединение с того же IP-адреса должно быть установлено не ранее, чем через 1 секунду после завершения предыдущего соединения. Если клиент попытается переподключиться раньше, то TCP-соединение будет отклонено.

При попытке двух и более одновременных TCP-соединений и отправки в них одного и того же идентификатора TWIME логина (Credentials в сообщении Establish), все такие соединения могут завершиться с ошибкой. Во все сессии WireGate отправит сообщение EstablishmentReject с указанием причины EstablishmentRejectCode=1 (AlreadyEstablished).

3.2.2. Мониторинг состояния сессии

Для мониторинга состояния сессии клиент и WireGate должны с заданной периодичностью обмениваться сообщениями типа Heartbeat (в нашем случае - это сообщение Sequence). Клиент передает свой интервал отправки Heartbeat-сообщений на WireGate в сообщении Establish в поле KeepaliveInterval. WireGate сообщает клиенту свой интервал отправки Heartbeat-сообщений в сообщении EstablishmentAck в поле KeepaliveInterval.

WireGate гарантирует, что он будет присылать сообщения не реже чем 1 раз за интервал, но эти сообщения не обязательно типа Heartbeat.

Если в течение заданного интервала (KeepaliveInterval) от клиента не пришло никакого сообщения, то WireGate отключает клиента. Отключение (срабатывание COD) происходит в период от одного до двух интервалов плюс время на передачу сообщения. Посылать Heartbeat-сообщения следует с учетом времени передачи сообщения от клиента до WireGate.

Клиент должен посылать на WireGate не более трёх Heartbeat-сообщений в секунду, четвёртое Heartbeat-сообщение за секунду приведет к разрыву соединения с отправкой клиенту сообщения Terminate с причиной TooFastClient.

3.2.3. Нумерация сообщений

Для последовательной нумерации сообщений клиент должен поддерживать счетчик сообщений, входящих на него с WireGate. Начальный номер входящих сообщений клиент получает от WireGate в сообщении EstablishmentAck при установке соединения. В дальнейшем каждое сообщение прикладного уровня, полученное клиентом от WireGate, должно увеличивать счетчик на единицу. Сообщения сессионного уровня не должны увеличивать счетчик.

3.2.4. Запрос повторной посылки сообщений

В случае обнаружения пропущенных сообщений клиент может запросить повторную посылку сообщений от WireGate с помощью команды RetransmitRequest, в которой указывает порядковый номер первого запрашиваемого сообщения и количество запрашиваемых сообщений. В результате после подтверждения Retransmission производится повторная посылка сообщений.

Во время отправки повторных сообщений WireGate не отправляет клиенту новых сообщений. Их отправка выполняется после завершения обработки команды RetransmitRequest.

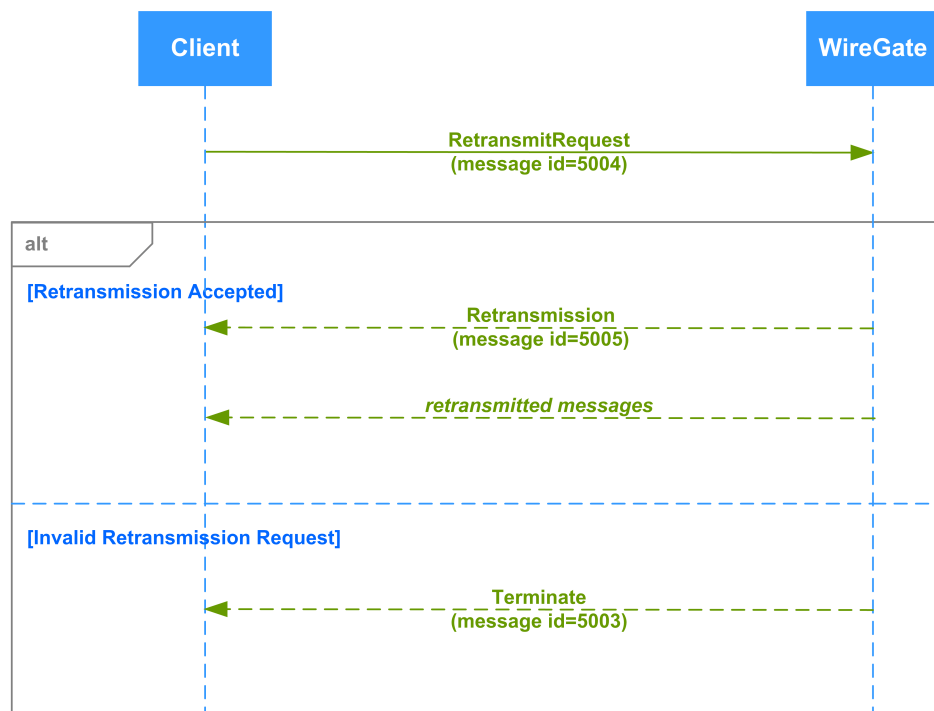


Рис. 3. Диаграмма. Запрос повторной посылки сообщений

3.2.5. Восстановление сессии после кратковременного сбоя

Для восстановления состояния клиент при установке нового соединения должен сравнить номер счетчика входящих сообщений с номером, пришедшим в сообщении EstablishmentAck. Если обнаружится, что NextSeqNo в EstablishmentAck больше его счетчика входящих сообщений, то клиент должен перезапросить пропущенные сообщения с помощью команды RetransmitRequest. WireGate отправит в ответ сообщения прикладного уровня. Одной командой RetransmitRequest клиент может запросить не более 1000 сообщений.

3.2.6. Полное восстановление сессии после сбоя

Для получения всех сообщений по сессии клиент при установке нового соединения должен получить номер счетчика входящих сообщений в сообщении EstablishmentAck в поле NextSeqNo. Затем клиент должен перезапросить сообщения с помощью команды RetransmitRequest. Следует учитывать, что если клиент запросил очень много данных, но читает их медленно, то WireGate будет вынужден отключить клиента с ошибкой TooSlowClient (см. раздел "раздел 3.4"). Одной командой RetransmitRequest клиент может запросить не более 1000 сообщений.

Далее клиенту следует подключиться к WireGate и выполнить шаги из раздела "раздел 3.2.5".

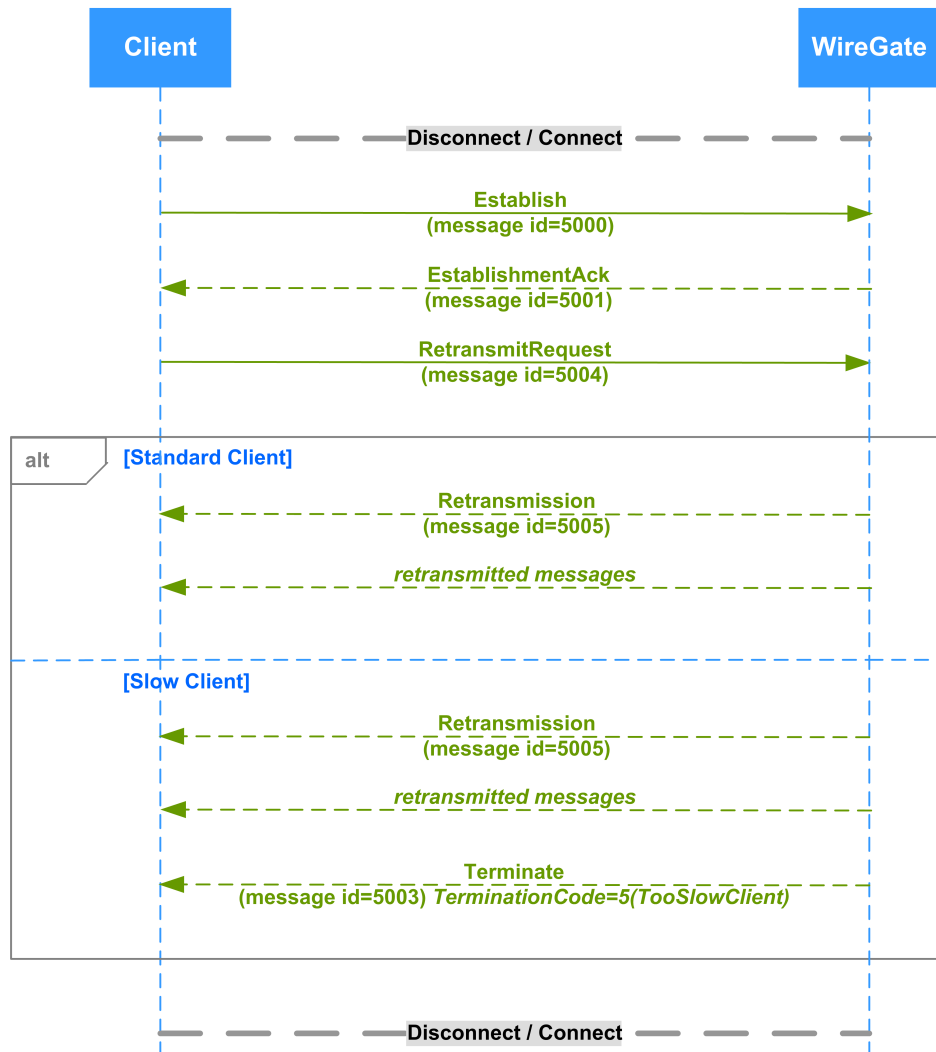


Рис. 4. Диаграмма. Полное восстановление сессии после сбоя

3.2.7. Сброс номеров сообщений

В период с 23:50 до 10:00 следующего дня происходит отключение клиентов от WireGate, очистка WireGate от сообщений торговой сессии предыдущего дня и сброс номеров сообщений. После сброса номеров и восстановления клиентом соединения к WireGate, клиент получит в сообщении EstablishmentAck в поле NextSeqNo новый номер, меньший или равный тому, что был до сброса номеров. Клиент должен новым номером из поля NextSeqNo инициализировать свой счетчик сообщений. После сброса номеров сообщений в WireGate для восстановления остаются доступны только сообщения прикладного уровня вечерней торговой сессии.

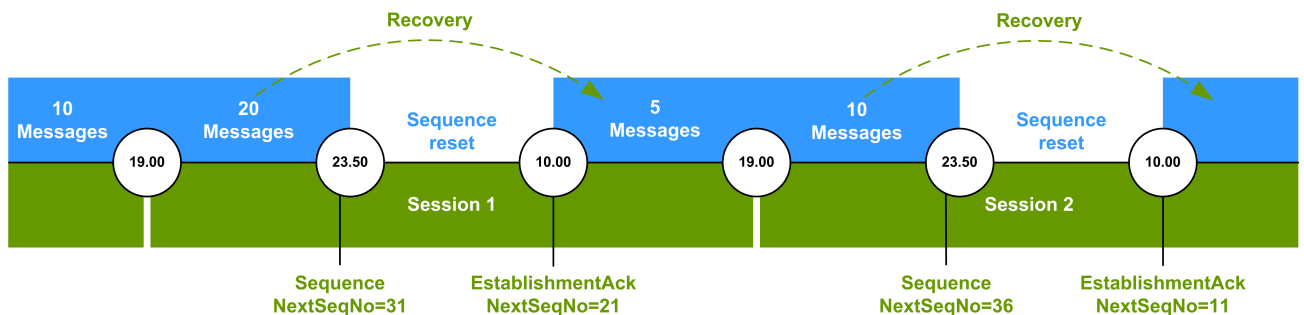


Рис. 5. Диаграмма. Сброс номеров сообщений

3.3. Контроль аномальной активности

В WireGate действует система ограничения аномальной активности клиентских приложений. Она не позволяет приложению пользователя (в рамках одной сессии) присылать более оговорённого в заявке на подключение количества сообщений в единицу вре-

мени. В настоящий момент можно получить логин в систему с ограничением 30, 60, 90 и т. д. (но не более 3000) торговых операций в секунду.

При превышении лимита сообщений система контроля аномальной активности посылает пользователю сообщение FloodReject с уведомлением об отказе в обслуживании.

Количество сообщений за истекшую секунду оценивается при приёме КАЖДОГО сообщения. Это значит, что если пользователь постоянно присылает запросы с частотой, больше, чем ему разрешено, то его сообщения перестают обрабатываться совсем.

Если же пользователь превысит отведенную ему квоту более чем в два раза, WireGate вообще разорвет соединение (сообщение Terminate с TerminationCode=4).

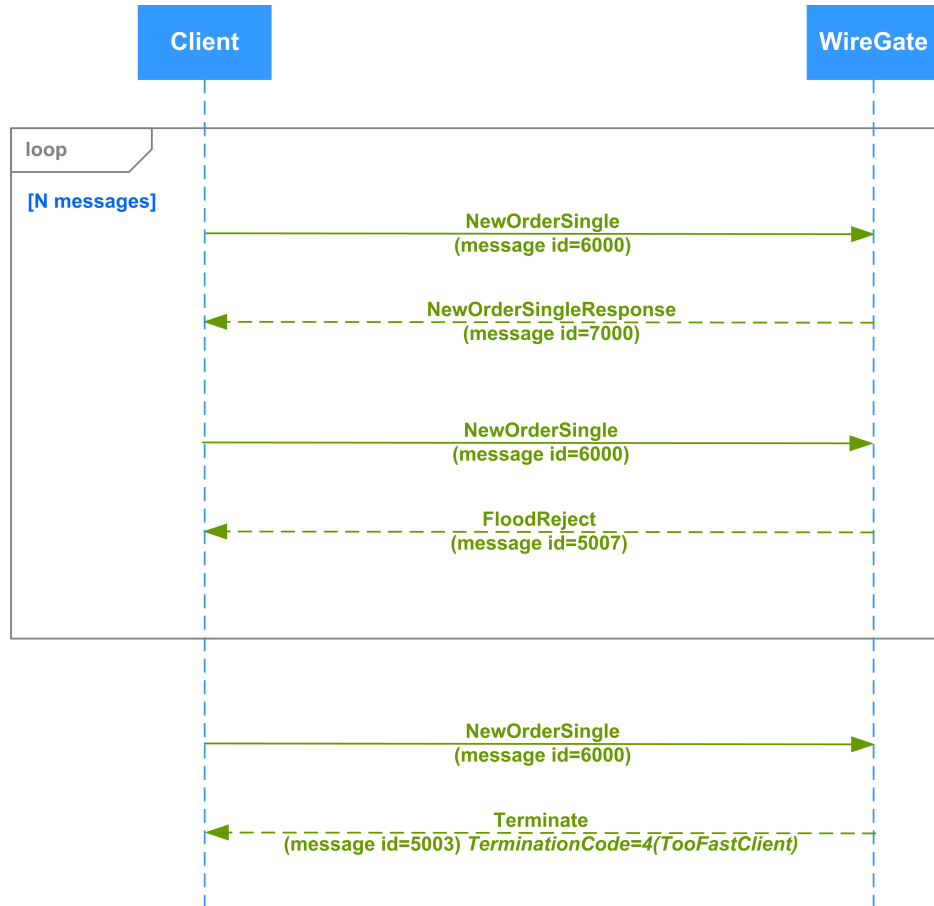


Рис. 6. Диаграмма. Контроль аномальной активности

3.4. Отключение при переполнении буфера на стороне WireGate

Если клиент в силу каких-либо причин не успевает вычитывать сообщения из TCP сокета, то при переполнении исходящего TCP буфера на стороне WireGate такой клиент будет отключен (сообщение Terminate с TerminationCode=5).

4. Прикладной уровень

4.1. Поддерживаемые сообщения

Транслируются от клиента к WireGate:

- **NewStream** – Создание потока на запрос ликвидности (выполняет потребитель ликвидности LC)
- **CancelStream** – Закрытие потока ликвидности потребителем ликвидности
- **RfsQuote** – Добавление/изменение котировок провайдерами ликвидности (LP)
- **RfsQuoteMassCancel** – Удаление котировок по типу провайдером ликвидности
- **RfsQuoteHit** – Подтверждение котировки потребителем ликвидности
- **RfsConfirmation** – Подтверждение квазисделки провайдером ликвидности

Транслируются от WireGate клиенту:

- **NewStreamResponse** – Успешное создание потока для потребителя ликвидности. Уведомление о создании потока для провайдеров ликвидности.
- **NewStreamReject** – Отклонение запроса на добавление нового потока ликвидности для LC
- **CancelStreamResponse** – Успешное удаление потока для LC, Уведомление о закрытии потока для провайдеров ликвидности.
- **CancelStreamReject** – Отклонение запроса на удаление потока для LC
- **RfsQuoteResponse** – Успешное добавление котировки (ответ для LP)
- **RfsQuoteCancelResponse** – Успешное снятие котировки (ответ для LP)
- **RfsQuoteReplaceResponse** – Успешное перемещение котировки (ответ для LP)
- **RfsQuoteReject** – Отклонение запроса на добавление/перемещение котировки (ответ для LP)
- **RfsQuoteMassCancelAck** – Сообщение о результатах массового снятия котировок (ответ для LP)
- **RfsBestQuoteUpdate** – Лучшая котировка провайдеров для LC
- **RfsQuoteHitAck** – Ответ на подтверждение котировки для LC
- **RfsConfirmationAck** – Уведомление об успешном или неудачном подтверждении квазисделки
- **RfsExecutionReport** – Сведения котировок в квазисделку и уведомления LC и LP о статусе сделки
- **EmptyBook** – Очистка контейнера активных котировок на клиентах
- **SystemEvent** – События в системе OTC

4.1.1. NewStream (message id=8007)

Потребитель ликвидности имеет возможность подать в систему OTC запрос на открытие потока ликвидности, в который провайдеры могут добавлять свои котировки. Операция доступна только LC.

Tag	Поле	Наличие	Тип	Описание
<Header>		Y		
1166	QuoteMsgID	Y	UInt64	Пользовательский идентификатор приказа на создание потока.
110	MinQty	Y	UInt64	Объем запроса (количество единиц инструмента). Объем имеет ограничение снизу, зависящее от базового актива. Пользователь не может задать значение, меньше минимально допустимого объема.
20027	ExternalID	N	UInt64	Внешний идентификатор потока.
48	SecurityID	Y	UInt32	Идентификатор инструмента.
54	Side	Y	SideEnum	Направление: <ul style="list-style-type: none"> • "1" – Покупка. • "2" – Продажа. • "89" – Оба направления.
21011	StreamExposureDuration	Y	StreamExposureDurationEnum	Длительность потока:

Tag	Поле	Наличие	Тип	Описание
				<ul style="list-style-type: none"> "1" – 30 секунд. "2" – 60 секунд. "3" – 90 секунд. "4" – 120 секунд.
574	MatchType	Y	MatchTypeEnum	Типы котировок, допустимые в потоке: <ul style="list-style-type: none"> "4" – Только твердые котировки, подтверждаемые автоматически (без Last Look). "10" – Любые котировки; как индикативные, требующие подтверждения от провайдера (есть Last Look), так и твердые, подтверждаемые автоматически (без Last Look). "
21000	SpeedBumpType	Y	SpeedBumpTypeEnum	Ограничение на частоту обновления котировок провайдером ликвидности, чтобы потребитель успевал реагировать на котировки. Время, в течение которого невозможны операции изменения (move) или отмены (cancel) котировки после ее выставления. Возможные значения: <ul style="list-style-type: none"> "0" – Нет ограничений. "1" – 200 миллисекунд. "2" – 500 миллисекунд. "3" – 1000 миллисекунд. "4" – 3000 миллисекунд.
1	Account	Y	String7	7-ми символьный код клиента.
21001	TextToLP	N	String20	Комментарий для LP.
58	Text	N	String20	Комментарий LC.

4.1.2. CancelStream (message id=8008)

Закрытие потока ликвидности. Операция доступна только LC.

Tag	Поле	Наличие	Тип	Описание
<Header>		Y		
1166	QuoteMsgID	Y	UInt64	Пользовательский идентификатор приказа на закрытие потока.
21002	AuctionID	Y	UInt64	Идентификатор потока ликвидности, назначенный системой RFS, уникален в течении торгового дня..
1	Account	Y	String7	7-ми символьный код клиента.

4.1.3. RfsQuote (message id=8009)

Добавление (изменение) котировок провайдером ликвидности. Операция доступна только LP.

Tag	Поле	Наличие	Тип	Описание
<Header>		Y		
1166	QuoteMsgID	Y	UInt64	Пользовательский идентификатор приказа на добавление/изменение котировки.
21002	AuctionID	Y	UInt64	Идентификатор потока ликвидности, назначенный системой RFS, уникален в течении торгового дня.
133	OfferPx	Y	Decimal5	Цена котировки на продажу.
20025	OfferExternalID	N	UInt64	Внешний идентификатор котировки на продажу.
132	BidPx	Y	Decimal5	Цена котировки на покупку.
20024	BidExternalID	N	UInt64	Внешний идентификатор котировки на покупку.

Tag	Поле	Наличие	Тип	Описание
1629	ExposureDuration	Y	UInt64	Время жизни котировки в микросекундах.
574	MatchType	Y	MatchTypeEnum	Тип котировки: <ul style="list-style-type: none"> • "4" – Твердая котировка, подтверждаемая автоматически (без Last Look). • "10" – Индикативная котировка, требующая подтверждения от провайдера (есть Last Look).
54	Side	Y	SideEnum	Направление: <ul style="list-style-type: none"> • "1" – Покупка. • "2" – Продажа. • "89" – Оба направления.
1	Account	Y	String7	7-ми символьный код клиента.
21003	OfferText	N	String20	Комментарий в котировке на продажу.
21004	BidText	N	String20	Комментарий в котировке на покупку.

4.1.4. RfsQuoteMassCancel (message id=8011)

Удаление котировок провайдером.

Провайдер выполняет массовое удаление котировок, посылая сообщение RfsQuoteMassCancel, заполнив одно из полей:

- SecurityID - удаление котировок с заданным идентификатором инструмента;
- Account - удаление котировок с заданным кодом клиента;
- ExternalID - удаление котировок с заданным внешним идентификатором.

Провайдер выполняет удаление одной котировки, отправляя сообщение RfsQuoteMassCancel с заполненными полями AuctionID и Side.

Tag	Поле	Наличие	Тип	Описание
<Header>		Y		
1166	QuoteMsgID	Y	UInt64	Пользовательский идентификатор приказа на удаление котировок.
21002	AuctionID	N	UInt64	Идентификатор потока ликвидности, назначенный системой RFS, уникален в течении торгового дня.
20027	ExternalID	N	UInt64	Внешний идентификатор котировки.
48	SecurityID	N	Int32	Идентификатор инструмента.
54	Side	N	SideEnum	Направление котировки: <ul style="list-style-type: none"> • "1" – Котировка на покупку. • "2" – Котировка на продажу. • "89" – Все котировки.
1	Account	N	String7	7-ми символьный код клиента.

4.1.5. RfsQuoteHit (message id=8012)

Запрос на исполнение котировки. Операция доступна только LC.

Tag	Поле	Наличие	Тип	Описание
<Header>		Y		
1166	QuoteMsgID	Y	UInt64	Пользовательский идентификатор запроса на исполнение котировки.
21002	AuctionID	Y	UInt64	Идентификатор потока ликвидности, назначенный системой RFS, уникален в течении торгового дня.
44	Price	Y	Decimal5	Цена котировки.
54	Side	Y	SideEnum	Направление: <ul style="list-style-type: none"> • "1" – Покупка.

Tag	Поле	Наличие	Тип	Описание
				<ul style="list-style-type: none"> "2" – Продажа. "89" – Оба направления.
58	Text	N	String20	Комментарий LC.

4.1.6. RfsConfirmation (message id=8013)

Подтверждение квазисделки провайдером ликвидности. В процессе заключения квазисделок в системе OTC предусмотрен этап, когда провайдер может отказаться от заключения квазисделки по выставленной котировке, либо подтвердить условия сделки ("Last Look"). Если провайдер отказывается от заключения квазисделки, его котировка снимается и ему начисляются штрафные баллы. Чтобы подтвердить квазисделку следует в ответ на приход сообщения **RfsExecutionReport** со статусом квазисделки "1" (ожидается подтверждение от пассивной стороны), отправить команду **RfsConfirmation** с идентификатором этой квазисделки. Время на ответ ограничено, превышение трактуется как отказ от подтверждения.

Провайдер при выставлении котировки может отказаться от этапа подтверждения, указав в сообщении **RfsQuote** в поле MatchType (Tag=574) признак "Котировка с автоподтверждением" (MatchType=4), тогда такая котировка считается подтвержденной по умолчанию.

Tag	Поле	Наличие	Тип	Описание
<Header>		Y		
1166	QuoteMsgID	Y	UInt64	Пользовательский идентификатор приказа на подтверждения квазисделки.
17	ExecID	Y	UInt64	Идентификатор квазисделки.

4.1.7. NewStreamResponse (message id=9011)

Успешное добавление потока ликвидности.

Tag	Поле	Наличие	Тип	Описание
<Header>		Y		
20204	Timestamp	Y	TimeStamp	Дата и время операции на сервере.
1166	QuoteMsgID	Y	UInt64	Пользовательский идентификатор из приказа на добавление потока.
21002	AuctionID	Y	UInt64	Идентификатор потока ликвидности, назначенный системой RFS, уникален в течении торгового дня.
110	MinQty	Y	UInt64	Объем (количество единиц инструмента).
20027	ExternalID	N	UInt64	Внешний идентификатор потока.
48	SecurityID	Y	UInt32	Идентификатор инструмента.
336	TradingSessionID	Y	Int32	Идентификатор торговой сессии.
21005	StreamFlags	Y	StreamFlagsSet	Поле признаков. Представляет собой битовую маску: <ul style="list-style-type: none"> 0x1 – В потоке допустимы только котировки с автоподтверждением ("LastLook" отсутствует). 0x2 – Поток закрыт.
167	SecurityType	Y	SecurityTypeEnum	Тип инструмента: <ul style="list-style-type: none"> "0" – Фьючерс. "1" – Опцион. "2" – Связка.
54	Side	Y	SideEnum	Направление: <ul style="list-style-type: none"> "1" – Покупка. "2" – Продажа. "89" – Оба направления.
21011	StreamExposureDuration	Y	StreamExposureDurationEnum	Длительность потока: <ul style="list-style-type: none"> "1" – 30 секунд. "2" – 60 секунд.

Tag	Поле	Наличие	Тип	Описание
				<ul style="list-style-type: none"> "3" – 90 секунд. "4" – 120 секунд.
21000	SpeedBumpType	Y	SpeedBumpTypeEnum	<p>Ограничение на частоту обновления котировок провайдерами ликвидности, чтобы потребитель успевал реагировать на котировки. Время, в течение которого невозможны операции изменения или отмены котировки после ее выставления. Возможные значения:</p> <ul style="list-style-type: none"> "0" – Нет ограничений. "1" – 200 миллисекунд. "2" – 500 миллисекунд. "3" – 1000 миллисекунд. "4" – 3000 миллисекунд.
21001	TextToLP	N	String20	Комментарий для LP.
58	Text	N	String20	Комментарий LC.
21007	TagOfLC	N	String64	Тэг потребителя (дополнительный идентификатор).

4.1.8. NewStreamReject (message id=9012)

Отклонение запроса на создание потока ликвидности.

Tag	Поле	Наличие	Тип	Описание
<Header>		Y		
20204	Timestamp	Y	TimeStamp	Дата и время операции на сервере.
1166	QuoteMsgID	Y	UInt64	Пользовательский идентификатор из приказа на добавление потока.
300	QuoteRejectReason	Y	Int32	Код ошибки.

4.1.9. CancelStreamResponse (message id=9013)

Успешное закрытие потока ликвидности.

Tag	Поле	Наличие	Тип	Описание
<Header>		Y		
20204	Timestamp	Y	TimeStamp	Дата и время операции на сервере.
1166	QuoteMsgID	Y	UInt64	Пользовательский идентификатор из приказа на закрытие потока.
21002	AuctionID	Y	UInt64	Идентификатор потока ликвидности, назначенный системой RFS, уникален в течении торгового дня.
110	MinQty	Y	UInt64	Объем (количество единиц инструмента).
20027	ExternalID	N	UInt64	Внешний идентификатор потока.
17	ExecID	Y	UInt64	Идентификатор квазисделки.
48	SecurityID	Y	UInt32	Идентификатор инструмента.
336	TradingSessionID	Y	Int32	Идентификатор торговой сессии.
54	Side	Y	SideEnum	<p>Направление:</p> <ul style="list-style-type: none"> "1" – Покупка. "2" – Продажа. "89" – Оба направления.
21011	StreamExposureDuration	Y	StreamExposureDurationEnum	<p>Длительность потока:</p> <ul style="list-style-type: none"> "1" – 30 секунд. "2" – 60 секунд. "3" – 90 секунд.

Tag	Поле	Наличие	Тип	Описание
				<ul style="list-style-type: none"> "4" – 120 секунд.
21005	StreamFlags	Y	StreamFlagsSet	Поле признаков. Представляет собой битовую маску: <ul style="list-style-type: none"> 0x1 - В потоке только котировки с автоподтверждением ("LastLook" отсутствует). 0x2 - Поток закрыт.
167	SecurityType	Y	SecurityTypeEnum	Тип инструмента: <ul style="list-style-type: none"> "0" – Фьючерс. "1" – Опцион. "2" – Связка.
21000	SpeedBumpType	Y	SpeedBumpTypeEnum	Ограничение на частоту обновления котировок провайдерами ликвидности, чтобы потребитель успевал реагировать на котировки. Время, в течение которого невозможны операции изменения или отмены котировки после ее выставления. Возможные значения: <ul style="list-style-type: none"> "0" – Нет ограничений. "1" – 200 миллисекунд. "2" – 500 миллисекунд. "3" – 1000 миллисекунд. "4" – 3000 миллисекунд.
21008	CancelReason	Y	CancelReasonEnum	Причина закрытия потока: <ul style="list-style-type: none"> "1" – Сделка совершена. "2" – Сделка не совершена по причине отсутствия средств у LC. "3" – По запросу LC. "4" – По времени. "5" – Закрыт администратором торгов. "6" – Окончание торговой сессии.
21001	TextToLP	N	String20	Комментарий для LP.
58	Text	N	String20	Комментарий LC.
21007	TagOfLC	N	String64	Тэг потребителя (дополнительный идентификатор).

4.1.10. CancelStreamReject (message id=9014)

Отклонение запроса на закрытие потока.

Tag	Поле	Наличие	Тип	Описание
<Header>		Y		
20204	Timestamp	Y	TimeStamp	Дата и время операции на сервере.
1166	QuoteMsgID	Y	UInt64	Пользовательский идентификатор из приказа на закрытие потока.
300	QuoteRejectReason	Y	Int32	Код ошибки.

4.1.11. RfsQuoteResponse (message id=9015)

Сообщение об успешном выставлении котировки LP.

Tag	Поле	Наличие	Тип	Описание
<Header>		Y		
20204	Timestamp	Y	TimeStamp	Дата и время операции на сервере.
1166	QuoteMsgID	Y	UInt64	Пользовательский идентификатор из приказа на добавление котировки.

Tag	Поле	Наличие	Тип	Описание
21002	AuctionID	Y	UInt64	Идентификатор потока ликвидности, назначенный системой RFS, уникален в течении торгового дня.
1751	SecondaryQuoteID	Y	UInt64	Уникальный идентификатор котировки, назначаемый биржей.
20026	QuoteSize	Y	UInt64	Объём котировки.
44	Price	Y	Decimal5	Цена котировки.
20027	ExternalID	N	UInt64	Внешний идентификатор котировки.
1629	ExposureDuration	Y	UInt64	Время жизни котировки в микросекундах.
20215	Flags	Y	FlagsSet	Поле признаков. Представляет собой битовую маску: <ul style="list-style-type: none"> • 0x1 - Котировочная (Day) • 0x2 - Встречная (IOC) • 0x8000000 - Связка • 0x4000000000000000 - Котировка с автоподтверждением ("LastLook" отсутствует)
48	SecurityID	Y	Int32	Идентификатор инструмента.
336	TradingSessionID	Y	Int32	Идентификатор торговой сессии.
167	SecurityType	Y	SecurityTypeEnum	Тип инструмента: <ul style="list-style-type: none"> • "0" – Фьючерс. • "1" – Опцион. • "2" – Связка.
54	Side	Y	SideEnum	Направление котировки: <ul style="list-style-type: none"> • "1" – Котировка на покупку. • "2" – Котировка на продажу.
21009	CodeOfLP	Y	String20	Код LP.
58	Text	N	String20	Комментарий из котировки.

4.1.12. RfsQuoteReplaceResponse (message id=9016)

Сообщение об успешном перемещении котировки LP.

Tag	Поле	Наличие	Тип	Описание
<Header>		Y		
20204	Timestamp	Y	TimeStamp	Дата и время операции на сервере.
1166	QuoteMsgID	Y	UInt64	Пользовательский идентификатор из приказа на перемещение котировки.
21002	AuctionID	Y	UInt64	Идентификатор потока ликвидности, назначенный системой RFS, уникален в течении торгового дня.
1751	SecondaryQuoteID	Y	UInt64	Уникальный идентификатор котировки, назначаемый биржей.
20026	QuoteSize	Y	UInt64	Объём котировки.
44	Price	Y	Decimal5	Цена котировки.
20027	ExternalID	N	UInt64	Внешний идентификатор котировки.
20034	PrevSecondaryQuoteID	Y	UInt64	Идентификатор предыдущий котировки.
1629	ExposureDuration	Y	UInt64	Время жизни котировки в микросекундах.
20215	Flags	Y	FlagsSet	Поле признаков. Представляет собой битовую маску: <ul style="list-style-type: none"> • 0x1 - Котировочная (Day) • 0x2 - Встречная (IOC) • 0x1000000 - Запись является результатом операции перемещения котировки • 0x80000000 - Связка

Tag	Поле	Наличие	Тип	Описание
				• 0x4000000000000000 - Котировка с автоподтверждением ("LastLook" отсутствует)
48	SecurityID	Y	Int32	Идентификатор инструмента.
336	TradingSessionID	Y	Int32	Идентификатор торговой сессии.
167	SecurityType	Y	SecurityTypeEnum	Тип инструмента: <ul style="list-style-type: none"> • "0" – Фьючерс. • "1" – Опцион. • "2" – Связка.
54	Side	Y	SideEnum	Направление котировки: <ul style="list-style-type: none"> • "1" – Котировка на покупку. • "2" – Котировка на продажу.
21009	CodeOfLP	Y	String20	Код LP.

4.1.13. RfsQuoteReject (message id=9017)

Отклонение запроса на добавление котировки.

Tag	Поле	Наличие	Тип	Описание
<Header>		Y		
1166	QuoteMsgID	Y	UInt64	Пользовательский идентификатор из приказа на добавление котировки.
20204	Timestamp	Y	TimeStamp	Дата и время операции на сервере.
300	QuoteRejectReason	Y	Int32	Код ошибки.
54	Side	Y	SideEnum	Направление котировки: <ul style="list-style-type: none"> • "1" – Котировка на покупку. • "2" – Котировка на продажу.

4.1.14. RfsQuoteCancelResponse (message id=9018)

Сообщение об успешном снятии котировки LP.

Tag	Поле	Наличие	Тип	Описание
<Header>		Y		
1166	QuoteMsgID	Y	UInt64	Пользовательский идентификатор из приказа на снятие котировки.
20204	Timestamp	Y	TimeStamp	Дата и время операции на сервере.
21002	AuctionID	Y	UInt64	Идентификатор потока ликвидности, назначенный системой RFS, уникален в течении торгового дня.
1751	SecondaryQuoteID	Y	UInt64	Уникальный идентификатор котировки, назначаемый биржей.
20026	QuoteSize	Y	UInt64	Объём котировки.
20027	ExternalID	N	UInt64	Внешний идентификатор котировки.
20215	Flags	Y	FlagsSet	Поле признаков. Представляет собой битовую маску: <ul style="list-style-type: none"> • 0x1 - Котировочная (Day) • 0x2 - Встречная (IOC) • 0x100000 - Запись является результатом операции перемещения котировки • 0x200000 - Запись является результатом операции удаления котировки • 0x400000 - Запись является результатом массового удаления • 0x8000000 - Связка

Tag	Поле	Наличие	Тип	Описание
				<ul style="list-style-type: none"> 0x1000000000 - Признак удаления котировки в результате применения штрафа 0x20000000000000 - Признак удаления по Timeout 0x40000000000000 - Котировка с автоподтверждением ("LastLook" отсутствует)
336	TradingSessionID	Y	Int32	Идентификатор торговой сессии.

4.1.15. RfsQuoteMassCancelAck (message id=9020)

Сообщение о результатах массового снятия котировок.

Tag	Поле	Наличие	Тип	Описание
<Header>		Y		
1166	QuoteMsgID	Y	UInt64	Пользовательский идентификатор из приказа на массовое снятие котировок.
20204	Timestamp	Y	TimeStamp	Дата и время операции на сервере.
1168	TotNoCxlIdQuotes	N	Int32	Количество снятых котировок.
21012	TotNoSpeedBumpQuotes	N	Int32	Количество не снятых котировок из-за ограничения 'Speed bump'.
300	QuoteRejectReason	Y	Int32	Код ошибки.

4.1.16. RfsBestQuoteUpdate (message id=9021)

Лучшая котировка. Присылается потребителю ликвидности при обновлении лучших цен покупки/продажи в потоке. Значения '0' в полях SecondaryQuoteID, QuoteSize, Price обозначает удаление лучшей котировки в потоке ликвидности.

Tag	Поле	Наличие	Тип	Описание
<Header>		Y		
21002	AuctionID	Y	UInt64	Идентификатор потока ликвидности, назначенный системой RFS, уникален в течении торгового дня.
1751	SecondaryQuoteID	Y	UInt64	Уникальный идентификатор котировки, назначаемый биржей.
20026	QuoteSize	Y	UInt64	Объём котировки.
44	Price	Y	Decimal5	Цена котировки.
54	Side	Y	SideEnum	Направление котировки: <ul style="list-style-type: none"> "1" – Котировка на покупку. "2" – Котировка на продажу.
574	MatchType	Y	MatchTypeEnum	Тип котировки: <ul style="list-style-type: none"> "4" – Твердая котировка, подтверждаемая автоматически (без Last Look). "10" – Индикативная котировка, требующая подтверждения от провайдера (есть Last Look).

4.1.17. RfsQuoteHitAck (message id=9022)

Сообщение об успешном или неудачном выполнении запроса на исполнение котировки.

Tag	Поле	Наличие	Тип	Описание
<Header>		Y		
1166	QuoteMsgID	Y	UInt64	Пользовательский идентификатор из приказа на добавление котировки потребителя ликвидности.
20204	Timestamp	Y	TimeStamp	Дата и время операции на сервере.
1751	SecondaryQuoteID	Y	UInt64	Уникальный идентификатор котировки, назначаемый биржей.
300	QuoteRejectReason	Y	Int32	Код ошибки.

4.1.18. RfsConfirmationAck (message id=9023)

Уведомление об успешном или неудачном подтверждении квазисделки.

Tag	Поле	Наличие	Тип	Описание
<Header>		Y		
1166	QuoteMsgID	Y	UInt64	Пользовательский идентификатор из приказа на добавление котировки потребителя ликвидности.
20204	Timestamp	Y	TimeStamp	Дата и время операции на сервере.
17	ExecID	Y	UInt64	Уникальный идентификатор котировки, назначаемый биржей.
300	QuoteRejectReason	Y	Int32	Код ошибки.

4.1.19. RfsExecutionReport (message id=9024)

Сведение котировок в квазисделку. Данное сообщение присылается в двух случаях:

- при формировании квазисделок в RFS;
- при изменении статуса квазисделок в RFS.

Tag	Поле	Наличие	Тип	Описание
<Header>		Y		
1166	QuoteMsgID	Y	UInt64	Пользовательский идентификатор из приказа на добавление котировки потребителя ликвидности.
20204	Timestamp	Y	TimeStamp	Дата и время операции на сервере.
21002	AuctionID	Y	UInt64	Идентификатор потока ликвидности, назначенный системой RFS, уникален в течении торгового дня.
1751	SecondaryQuoteID	Y	Int64	Уникальный идентификатор котировки, назначаемый биржей.
31	LastPx	Y	Decimal5	Цена заключённой сделки.
32	LastQty	Y	UInt32	Количество единиц инструмента в сделке.
1629	ExposureDuration	Y	UInt64	Время жизни котировки в микросекундах.
20027	ExternalID	N	UInt64	Внешний идентификатор котировки.
177	ExecID	Y	UInt64	Идентификатор квазисделки.
880	TrdMatchID	Y	UInt64	Идентификатор сделки в Spectra.
37	OrderID	Y	UInt64	Идентификатор заявки в Spectra.
336	TradingSessionID	Y	Int32	Идентификатор торговой сессии.
48	SecurityID	Y	Int32	Идентификатор инструмента.
103	OrdRejReason	N	Int32	Код причины отклонения приказа на добавление адресной заявки. Коды ошибок, транслируемые в поле OrdRejReason, можно посмотреть в документе p2gate_ru.pdf (ftp://ftp.moex.com/pub/ClientsAPI/Spectra/CGate/docs/). Перед поиском описания ошибки в списке, значение поля OrdRejReason необходимо уменьшить на 100.
167	SecurityType	Y	SecurityTypeEnum	Тип инструмента: <ul style="list-style-type: none"> • "0" – Фьючерс. • "1" – Опцион. • "2" – Связка.
54	Side	Y	SideEnum	Направление: <ul style="list-style-type: none"> • "1" – Покупка. • "2" – Продажа.
20030	Status	Y	StatusEnum	Статус квазисделки: <ul style="list-style-type: none"> • "0" – Квазисделка обрабатывается. • "1" – Ожидается подтверждение от провайдера. • "2" – Подтверждение от провайдера получено. • "3" – Сделка не заключена по причине ошибки. • "4" – Сделка заключена.

Tag	Поле	Наличие	Тип	Описание
20033	RejectReason	Y	RejectReasonEnum	Код ошибки из RFS. Возможные значения: <ul style="list-style-type: none"> "0" – Не используется. "1" – Квазисделка не подтверждена. "2" – Ошибка при добавлении заявки для потребителя. "3" – Ошибка при добавлении заявки для провайдера.
21009	CodeOfLP	Y	String20	Код LP.
58	Text	N	String20	Комментарий из котировки.

4.1.20. EmptyBook (message id=9009)

Очистка контейнера активных котировок на клиентах.

Tag	Поле	Наличие	Тип	Описание
<Header>		Y		
20204	Timestamp	Y	TimeStamp	Дата и время операции на сервере.
336	TradingSessionID	Y	Int32	Идентификатор торговой сессии.

4.1.21. SystemEvent (message id=9010)

События в торговой системе.

Tag	Поле	Наличие	Тип	Описание
<Header>		Y		
20204	Timestamp	Y	TimeStamp	Дата и время операции на сервере.
336	TradingSessionID	Y	Int32	Идентификатор торговой сессии.
1368	TradSesEvent	Y	TradSesEventEnum	Тип события: <ul style="list-style-type: none"> "101" SessionDataReady - Закончена загрузка данных из клиринговой системы в торговую перед началом новой торговой сессии. "102" IntradayClearingFinished - Все расчетные процедуры в промклиринге закончены. "104" IntradayClearingStarted - Начало промклиринга. "105" ClearingStarted - Начало основного клиринга "106" ExtensionOfLimitsFinished - Раздвижка лимитов закончена "108" BrokerRecalcFinished - Денежные средства после промклиринга пересчитаны "10100" OtcSessionInited - Сессия в OTC назначена "10101" OtcSessionStarted - Сессия в OTC идет "10102" OtcSessionSuspended - Сессия в OTC приостановлена "10103" OtcSessionStoped - Сессия в OTC остановлена "10104" OtcSessionFinished - Сессия в OTC завершена

4.1.22. Непрошенные сообщения

Котировки, выставленные одним пользователем (TWIME логином), могут удаляться или изменяться другими пользователями (через WireGate, P2Gate, терминал). Если котировка, выставленная TWIME логином, удаляется/изменяется другим логином, в сессию выставившего котировку логина, направляется "непрошеное сообщение". Непрошеное сообщение - это обычное сообщение об удалении (**RfsQuoteCancelResponse**) или изменении (**RfsQuoteReplaceResponse**) котировки со значением 'nullValue' в поле **QuoteMsgID**.

4.2. Сценарии торгового взаимодействия

4.2.1. Создание потока ликвидности потребителем ликвидности LC

Потребитель ликвидности (LC) посылает в торговую систему запрос на создание нового потока ликвидности - сообщение **NewStream**.

При успешном создании потока ликвидности:

- WireGate отправляет LC сообщение **NewStreamResponse**, содержащее идентификатор потока **AuctionID**. В сообщении **NewStreamResponse** поля **Account** и **Text** транслируются только LC;
- всем провайдерам ликвидности (LP) транслируется непрошеное сообщение **NewStreamResponse**. Если LC, создавший поток ликвидности, является одновременно и LP, то ему такое непрошеное сообщение не отправляется.

В случае ошибки при создании потока WireGate отправляет LC сообщение **NewStreamReject**.

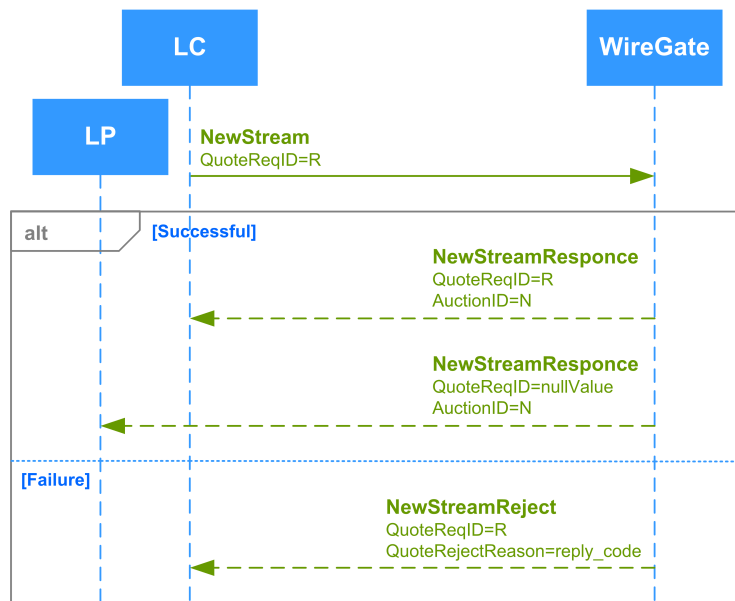


Рис. 7. Диаграмма. Создание потока ликвидности

4.2.2. Выставление котировки в поток провайдером ликвидности LP1, обновление лучших котировок для LC

LP1 добавляет новую котировку в пустой поток ликвидности, отправляя в WireGate сообщение **RfsQuote**.

При успешном добавлении котировки WireGate отправляет LP1 сообщение **RfsQuoteResponse**.

После успешного добавления котировки WireGate отправляет LC два сообщения **RfsBestQuoteUpdate** с обновлением лучших котировок на покупку и продажу.

В случае ошибки WireGate отправляет LP1 сообщение **RfsQuoteReject**.

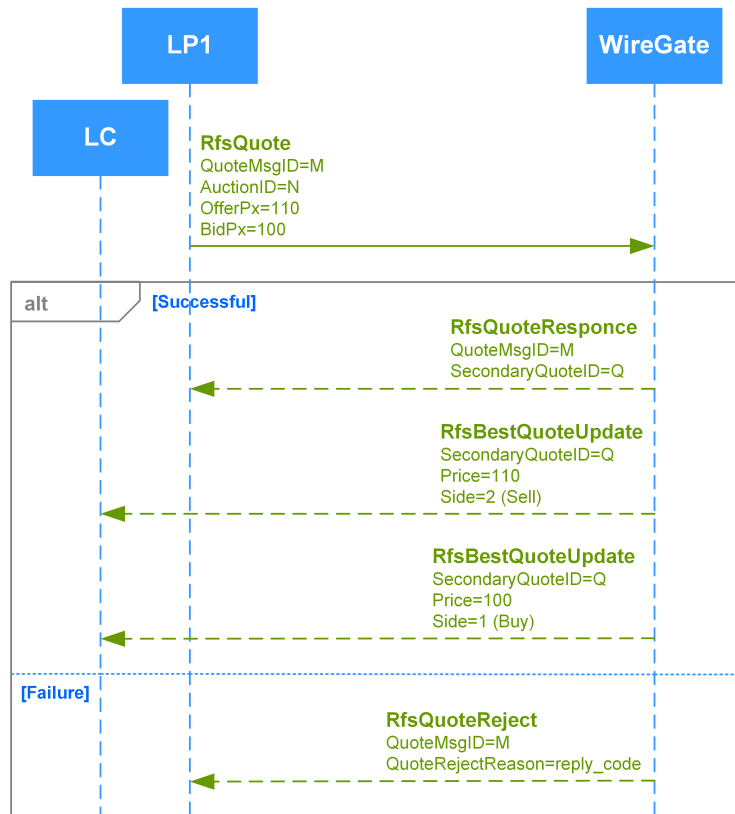


Рис. 8. Диаграмма. Выставление котировки LP1

4.2.3. Выставление котировки в поток провайдером ликвидности LP2, обновление лучших котировок для LC

В рамках одного потока ликвидности от провайдера может стоять только одна котировка (котировка может быть двухсторонней). Провайдеры не могут видеть котировки друг друга.

LC-владелец потока не может видеть все предложения от провайдеров, он видит только лучшую цену на покупку и продажу - TOP1. При поступлении от провайдеров котировки с лучшей ценой, чем поданная ранее в поток, котировка в потоке от LP заменяется на лучшую.

LP2 добавляет новую котировку, лучшую, чем имеющаяся котировка от LP1, отправляя в WireGate сообщение **RfsQuote**.

WireGate отправляет LP2 сообщение **RfsQuoteResponse**.

WireGate отправляет LC одно сообщение **RfsBestQuoteUpdate** с обновлением лучшей котировки. Сообщение одно, потому что LP2 улучшил только Bid (на схеме).

В случае ошибки WireGate отправляет LP2 сообщение **RfsQuoteReject**.

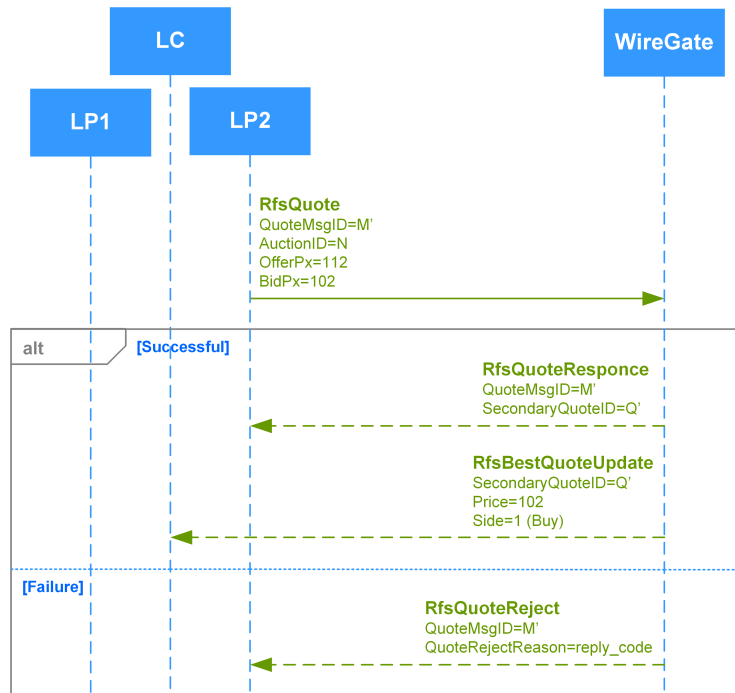


Рис. 9. Диаграмма. Выставление котировки LP2

4.2.4. Изменение котировки провайдером ликвидности LP2

При изменении котировки старая котировка удаляется и на ее место выставляется новая.

Провайдер ликвидности может менять свои котировки не чаще, чем заданное в потоке минимальное время жизни котировки. Но новая котировка будет принята системой в любом случае, если в ней указана лучшая цена, чем в выставленной ранее котировке.

LP2 меняет свою котировку, отправляя в WireGate сообщение **RfsQuote**.

При успешном изменении котировки WireGate отправляет LP2 сообщение **RfsQuoteReplaceResponse**.

В случае ошибки WireGate отправляет LP2 сообщение **RfsQuoteReject**.

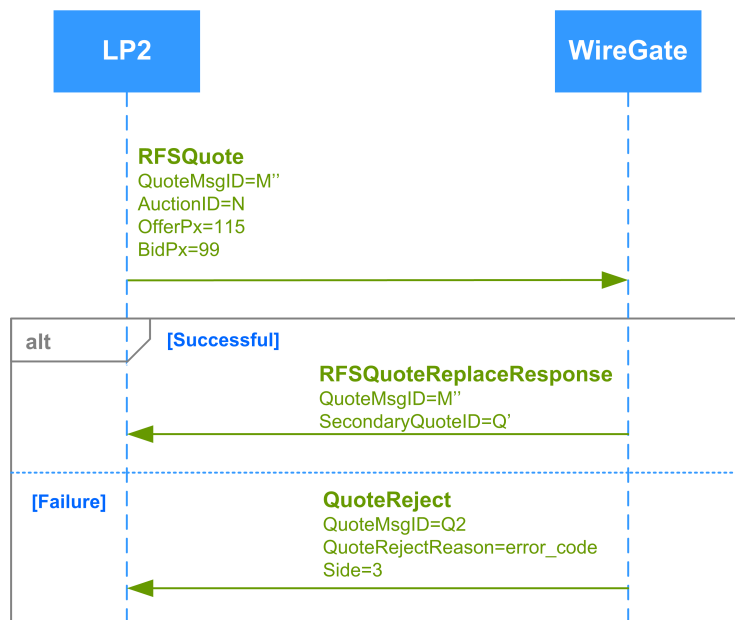


Рис. 10. Диаграмма. Изменение котировки LP2

4.2.5. Удаление котировок провайдером ликвидности LP

LP может выполнить массовое удаление котировок, посылая в WireGate сообщение **RfsQuoteMassCancel**. Массовое удаление производится по типу:

- по коду инструмента - поле **SecurityID**;
- по коду клиента - три последние символа в семизначном коде из поля **Account**;
- по внешнему номеру - поле **ExternalID**.

LP может выполнить удаление одной котировки, отправив в WireGate сообщение **RfsQuoteMassCancel** с заполненными полями **AuctionID** и **Side**.

Если в результате удаления котировок удаляются лучшие котировки в потоке для LC, то WireGate отправляет LC сообщение **RfsBestQuoteUpdate** со значениями полей:

- AuctionID = stream_id;
- SecondaryQuoteID = 0;
- QuoteSize = 0;
- Price = 0;
- Side = dir;
- MatchType - значение не определено.

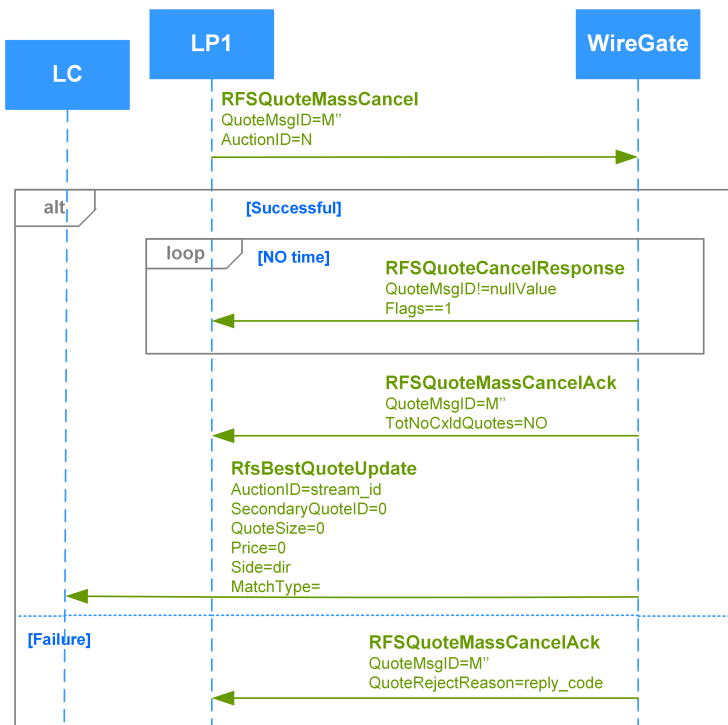


Рис. 11. Диаграмма. Удаление котировок LP

4.2.6. Удаление котировки провайдера по истечении времени жизни

При создании котировки LP в сообщении **RfsQuote** передаёт в поле **ExposureDuration** положительное значение - это время жизни котировки. По истечении времени жизни котировка удаляется из торговой системы и WireGate транслирует LP сообщение **RfsQuoteCancelResponse** (QuoteMsgID!=nullValue).

4.2.7. Закрытие потока ликвидности

Закрытие потока ликвидности может происходить по нескольким сценариям:

- при недостаточности средств у LC;
- при заключении сделки - поток закрывается автоматически;
- автоматическое закрытие при истечении времени жизни потока. Продолжительность потока задается LC при его открытии. По умолчанию продолжительность потока определяется торговой сессией - по ее окончании поток закрывается;
- по запросу потребителя ликвидности. LC может закрывать только свои потоки ликвидности. К своим относятся потоки, открытые от своего логина и от других логинов своих клиентов. Закрывать чужие потоки ликвидности нельзя.

4.2.7.1. Закрытие потока ликвидности по команде от потребителя ликвидности LC

LC посылает в WireGate сообщение **CancelStream**.

При успешном удалении потока WireGate отправляет:

- сообщение **CancelStreamResponse** - LC. В сообщении поля **Account** и **Text** транслируются только для LC.
- непрошеное сообщение **CancelStreamResponse** - всем пользователям, являющимся LP. Если LC - создатель потока ликвидности является одновременно и LP, то WireGate не отправляет ему непрошеное сообщение. WireGate транслирует LP пустые строки в полях **Account** и **Text**.
- сообщение **RfsQuoteCancelResponse** - LP-владельцу котировки (при наличии котировок в потоке).

В случае ошибки WireGate отправляет LC сообщение **CancelStreamReject**.

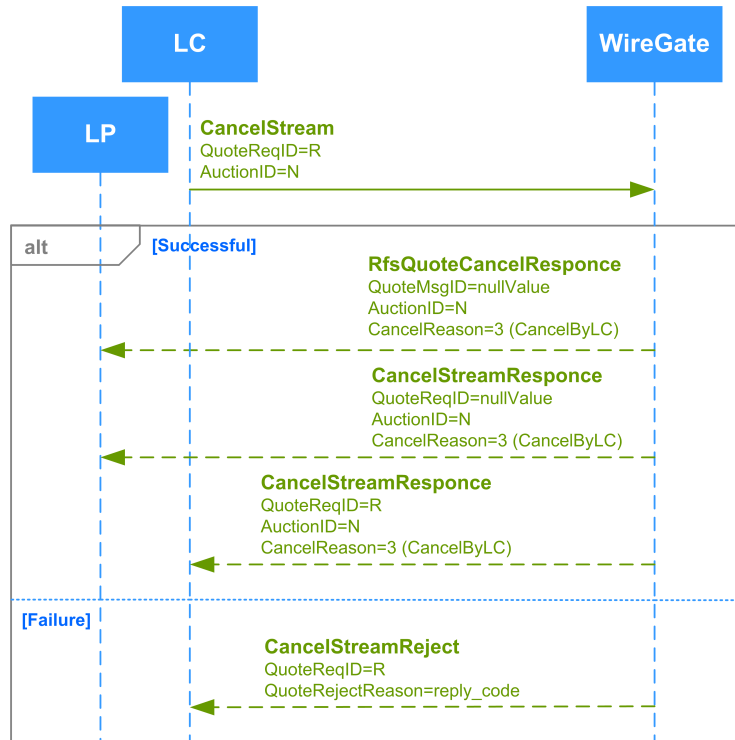


Рис. 12. Диаграмма. Закрытие потока по команде LC

4.2.7.2. Автоматическое закрытие потока ликвидности

Поток ликвидности может быть закрыт автоматически при наступлении следующих событий:

- по окончании торговой сессии, если точно не задано время жизни потока;
- по истечении времени жизни потока (StreamExposureDuration);
- при заключении сделки в Spectra по этому потоку;
- при недостаточности средств у LC в Spectra.

При закрытии потока WireGate отправляет LC сообщение **CancelStreamResponse**.

Всем пользователям, являющимся LP, WireGate транслирует непрошеное сообщение **CancelStreamResponse**.

При наличии котировок в потоке отправляет LP-владельцу котировки сообщение **RfsQuoteCancelResponse**.

При закрытии потока по окончании торговой сессии или по истечении времени жизни потока в поле **Flags** сообщения **RfsQuoteCancelResponse** взведен бит 0x20000000000000.

В сообщении **CancelStreamResponse** поле **CancelReason** может принимать следующие значения:

- 6 (EndOfTradingSession) - окончание торговой сессии;
- 4 (TimeOut) - окончание времени жизни потока;

- 1 (Deal) - заключение сделки по потоку;
- 2(LCDoesntHaveEnoughMoney) - недостаточно средств у LC;
- 5 (CancelByAdministrator) - решение администратора торгов.

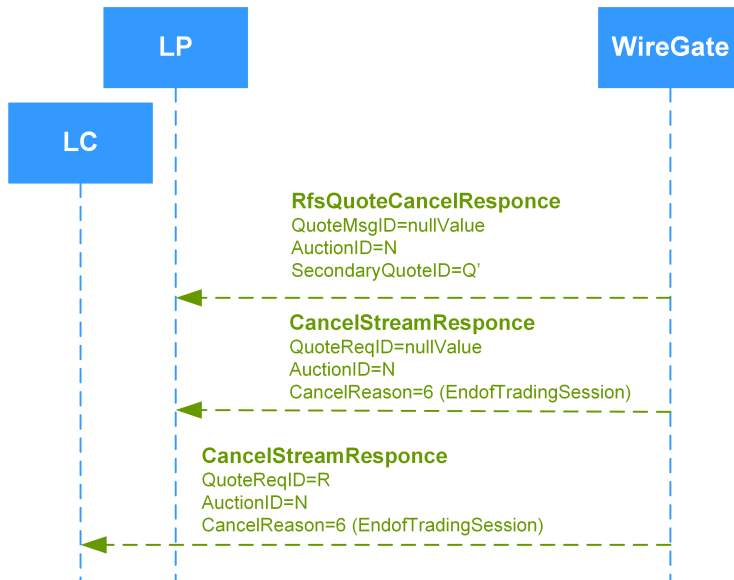


Рис. 13. Диаграмма. Автоматическое закрытие потока ликвидности

4.2.8. Выставление котировки в поток потребителем ликвидности LC

LC не может видеть все предложения от провайдеров LP. Он видит только лучшую цену на покупку и продажу и, если она его устраивает, может послать встречную котировку, отвечая на предложение.

Отвечая на лучшую котировку LC посылает в WireGate сообщение **RfsQuoteHit**.

При успешном сведении котировки в квазисделку WireGate отправляет LC сообщение **RfsQuoteHitAck** с полем **QuoteRejectReason = 0**.

В случае ошибки WireGate отправляет LC сообщение **RfsQuoteHitAck** с полем **QuoteRejectReason != 0**.

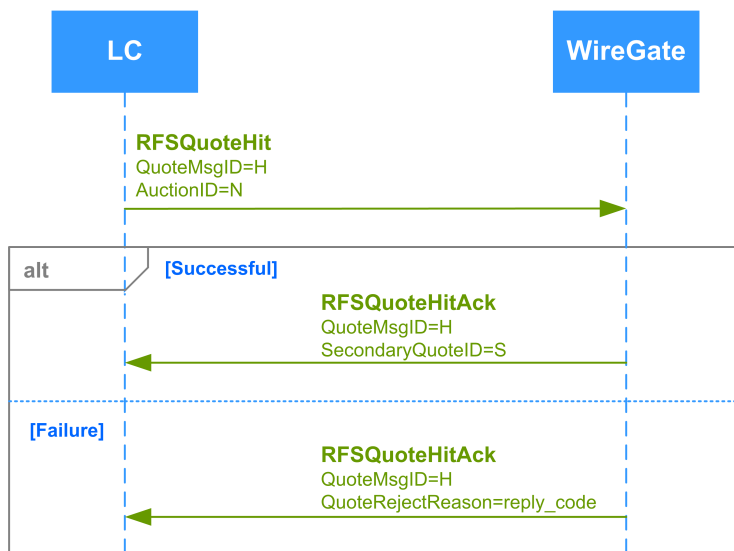


Рис. 14. Диаграмма. Встречная котировка LC

4.2.9. Сведение котировок в сделку и опциональная процедура Last Look

Принимая предложение провайдера LC выставляет встречную котировку. При этом в системе формируется квазисделка.

Если котировка от LP была с автоподтверждением, то сделка будет заключена автоматически, без запроса подтверждения от LP. Если котировка от LP была без автоподтверждения, до в процесс заключения сделки добавляется дополнительная итерация -

подтверждение от провайдера - процедура Last Look (см. раздел 4.2.9.4). Открывая поток ликвидности LC может настроить его так, что от провайдеров будут приниматься только котировки с автоподтверждением.

Далее в ТКС Спектра последовательно выставляются две заявки, которые сводятся в сделку. После этого, процесс сведения RFS котировок в сделку считается завершенным, и поток ликвидности закрывается.

Все этапы процесса сведения котировок сопровождаются соответствующим изменением состояния (статуса) квазисделки.

Алгоритм сведения котировок следующий:

1. После выставления котировки LC в RFS системе формируется квазисделка. Клиенту присылается сообщение **RfsExecutionReport** с **Status=0** (квазисделка обрабатывается).
2. Далее в ТКС Спектра выставляется адресная заявка от LC, с параметрами, указанными в квазисделке. При постановке заявки осуществляются все стандартные проверки, включая и проверку на достаточность обеспечения. Клиенту присылается сообщение **RfsExecutionReport** с **Status=1** (ожидается подтверждение от провайдера).
3. Система RFS в течение заданного (настраивается администратором) времени ожидает от провайдера ликвидности (LP) подтверждения квазисделки.
4. После получения подтверждения клиенту присылается сообщение **RfsExecutionReport** с **Status=2** (подтверждение от провайдера получено), и в ТКС Спектра выставляется заявка от провайдера ликвидности (LP), с параметрами, указанными в квазисделке.
5. После сведения двух заявок в ТКС Спектра в сделку, клиенту присылается сообщение **RfsExecutionReport** об изменении статуса квазисделки на **Status=4** (сделка заключена).
6. Если на каком-либо этапе сведения котировок происходит ошибка или сделка не подтверждена, то клиенту присылается сообщение **RfsExecutionReport** с **Status=3** (сделка не заключена по причине ошибки) с соответствующим кодом ошибки в поле **RejectReason**.

4.2.9.1. Запрос от LC на исполнение котировки (Quote Hit) в поток ликвидности без Last Look

Если LC устраивает котировка от LP, выставленная в поток, то он отправляет запрос на исполнение этой котировки, посылая в WireGate сообщение **RfsQuoteHit**.

При успешном исполнении котировки WireGate отправляет:

- LC сообщение **RfsExecutionReport**;
- LP, чья котировка исполнилась, сообщение **RfsExecutionReport**;
- всем пользователям, являющимся LP, сообщение **CancelStreamResponse** со значением поля **CancelReason = 1** (Deal).

В случае отказа WireGate отправляет LC сообщение **RfsQuoteHitAck** с ненулевым кодом отклонения в поле **QuoteRejectReason**.

На диаграмме показан случай успешного исполнения котировки в системе RFS и успешного заключения сделки в системе Spectra.

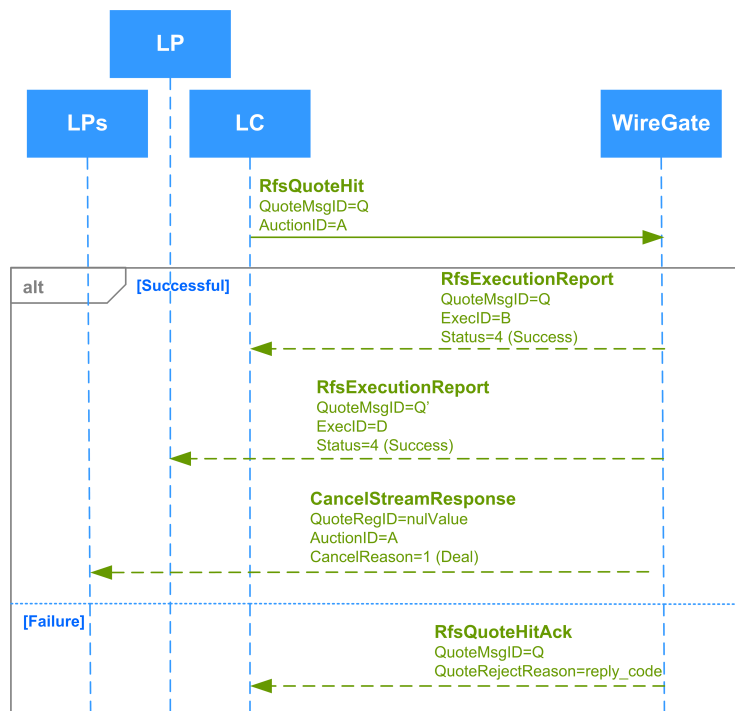


Рис. 15. Диаграмма. Успешное исполнение котировки и заключение сделки в Spectra

4.2.9.2. Ошибка при выставлении адресной заявки LP в торговую систему срочного рынка

Причиной не совершения сделки может быть нехватка средств у LP. При этом происходят следующие события:

- поток ликвидности продолжается;
- котировка LP удаляется из списка предложений от провайдеров;
- WireGate посылает LC сообщение **RfsExecutionReport** со значением поля **Status = 3 (Failed)**;
- WireGate посылает LP, чья котировка участвовала в квазисделке, сообщение **RfsExecutionReport** со значением поля **Status = 3 (Failed)**;
- если произошло изменение лучшей котировки в потоке, то WireGate транслирует LC сообщение **RfsBestQuoteUpdate** с обновлением лучших котировок (на диаграмме этот вариант не показан).

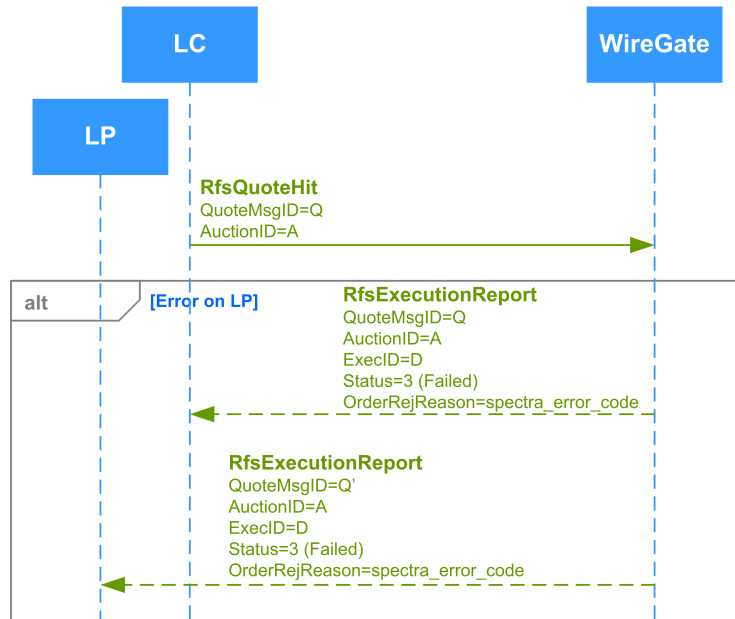


Рис. 16. Диаграмма. Ошибка при выставлении адресной заявки LP

4.2.9.3. Ошибка при выставлении адресной заявки LC в торговую систему срочного рынка

Причиной не совершения сделки может быть нехватка средств у LC. При этом происходят следующие события:

- WireGate посылает LC сообщение **RfsExecutionReport** со значением поля **Status = 3 (Failed)**;
- WireGate посылает LP, чья котировка участвовала в квазисделке, сообщение **RfsExecutionReport** со значением поля **Status = 3 (Failed)**;
- поток ликвидности закрывается;
- WireGate посылает LC сообщение **CancelStreamResponse** со значением поля **CancelReason = 2 (LCDoesntHaveEnoughMoney)**;
- WireGate посылает всем пользователям, являющимся LP, сообщение **CancelStreamResponse** со значением поля **CancelReason = 2 (LCDoesntHaveEnoughMoney)**;
- котировки всех провайдеров, поданные в этот поток, удаляются из торговой системы.

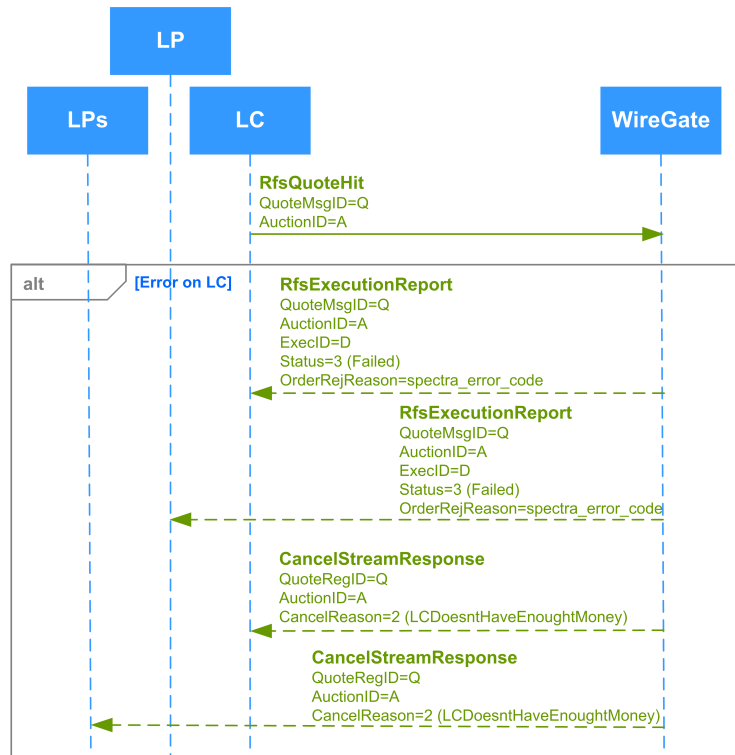


Рис. 17. Диаграмма. Ошибка при выставлении адресной заявки LC

4.2.9.4. Подтверждение квазисделки провайдером - процедура Last Look

LC отправляет запрос на исполнение котировки, посылая в WireGate сообщение **RfsQuoteHit**.

При успешном прохождении всех проверок для котировки LC WireGate отправляет LP сообщение **ExecutionReport** со значением поля **Status = 1** (WaitConfirm).

Далее возможно три варианта развития событий:

1. Провайдер подтверждает квазисделку:
 - LP отправляет в WireGate сообщение **RfsConfirmation**;
 - WireGate отправляет LC сообщение **RfsExecutionReport** со значением поля **Status = 2** (Confirmed);
 - WireGate отправляет LP сообщение **RfsExecutionReport** со значением поля **Status = 2** (Confirmed).
2. Провайдер не подтверждает квазисделку:
 - LP отправляет в WireGate сообщение **RfsConfirmation**;
 - WireGate отправляет LP сообщение **RfsConfirmationAck** с ненулевым кодом отклонения в поле **QuoteRejectReason**.
3. Если в течении таймаута LP не отправляет в WireGate сообщение **RfsConfirmationAck**, то система RFS считает, что LP отказался от сделки. В этом случае:
 - WireGate отправляет LC сообщение **RfsExecutionReport** со значением поля **Status = 3** (NotConfirmed);
 - WireGate отправляет LP сообщение **RfsExecutionReport** со значением поля **Status = 3** (NotConfirmed).

Если квазисделка не подтверждена провайдером или по таймауту, то в этом случае:

- поток ликвидности продолжается;
- котировка LP удаляется из списка котировок провайдеров в этом потоке;
- если произошло изменение лучших котировок в потоке, то WireGate транслирует LC сообщение **RfsBestQuoteUpdate** с обновлением лучших котировок (это сообщение не показано на схеме для простоты восприятия).

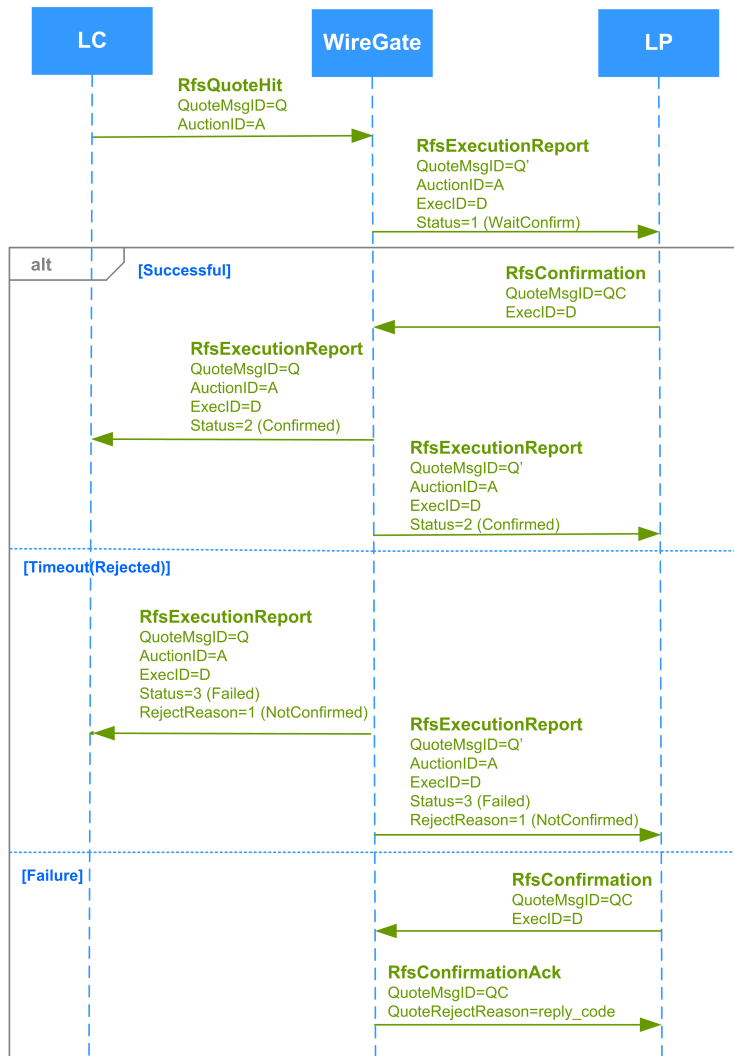


Рис. 18. Диаграмма. Процедура Last Look

4.2.10. Уникальность пользовательских идентификаторов котировок QuoteMsgID

Шлюз WireGate контролирует уникальность пользовательских идентификаторов котировок. От клиента требуется гарантировать уникальность **QuoteMsgID** на протяжении торговой сессии для всех котировок с временем жизни в одну торговую сессию. Если клиент присылает не уникальный **QuoteMsgID**, то WireGate отклоняет такое сообщение - клиенту отправляет сообщение **SessionReject** с кодом ошибки **101** (QuoteMsgIDIsNotUnique).

5. Схема сообщений

Ниже приведена актуальная схема сообщений.

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet href="sbe_schema.xsl" type="text/xsl"?>
<sbe:messageSchema package="moex_spectra_rfs_twime" byteOrder="littleEndian" id="20809" version="1"
xmlns:sbe="http://fixprotocol.io/2016/sbe"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://fixprotocol.io/2016/sbe sbe.xsd">
  <types>
    <type name="Int8" primitiveType="int8" minValue="-128"
      maxValue="126" nullValue="127" presence="optional" />
    <type name="Int16" primitiveType="int16" minValue="-32768"
      maxValue="32766" nullValue="32767" presence="optional" />
    <type name="Int32" primitiveType="int32" minValue="-2147483648"
      maxValue="2147483646" nullValue="2147483647" presence="optional" />
    <type name="Int64" primitiveType="int64" minValue="-9223372036854775808"
      maxValue="9223372036854775806" nullValue="9223372036854775807" presence="optional" />

    <type name="UInt8" primitiveType="uint8" minValue="0" maxValue="254"
      nullValue="255" presence="optional" />
    <type name="UInt16" primitiveType="uint16" minValue="0" maxValue="65534"
      nullValue="65535" presence="optional" />
    <type name="UInt32" primitiveType="uint32" minValue="0" maxValue="4294967294"
      nullValue="4294967295" presence="optional" />
    <type name="UInt64" primitiveType="uint64" minValue="0" maxValue="18446744073709551614"
      nullValue="18446744073709551615" presence="optional" />

    <type name="String7" primitiveType="char" length="7"/>
    <type name="String20" primitiveType="char" length="20"/>
    <type name="String25" primitiveType="char" length="25"/>
    <type name="String64" primitiveType="char" length="64"/>

    <type name="DeltaMillisecs" primitiveType="uint32" minValue="1000" maxValue="60000"
      presence="required" />
    <type name="TimeStamp" primitiveType="uint64" minValue="0" maxValue="18446744073709551614"
      nullValue="18446744073709551615" presence="optional"
      description="Time in number of nanoseconds since Unix epoch, UTC timezone" />

    <enum name="BooleanEnum" encodingType="uint8">
      <validValue name="False" >0</validValue>
      <validValue name="True" >1</validValue>
    </enum>

    <enum name="TerminationCodeEnum" encodingType="uint8">
      <validValue name="Finished" >0</validValue>
      <validValue name="UnspecifiedError" >1</validValue>
      <validValue name="ReRequestOutOfBounds" >2</validValue>
      <validValue name="ReRequestInProgress" >3</validValue>
      <validValue name="TooFastClient" >4</validValue>
      <validValue name="TooSlowClient" >5</validValue>
      <validValue name="MissedHeartbeat" >6</validValue>
      <validValue name="InvalidMessage" >7</validValue>
      <validValue name="TCPFailure" >8</validValue>
      <validValue name="InvalidSequenceNumber" >9</validValue>
      <validValue name="ServerShutdown" >10</validValue>
      <validValue name="SequenceReset" >11</validValue>
    </enum>

    <enum name="EstablishmentRejectCodeEnum" encodingType="uint8">
      <validValue name="Unnegotiated" >0</validValue>
      <validValue name="AlreadyEstablished" >1</validValue>
      <validValue name="SessionBlocked" >2</validValue>
      <validValue name="KeepaliveInterval" >3</validValue>
      <validValue name="Credentials" >4</validValue>
      <validValue name="Unspecified" >5</validValue>
    </enum>

    <enum name="SessionRejectReasonEnum" encodingType="uint8">
      <validValue name="ValueIsIncorrect" >5</validValue>
      <validValue name="Other" >99</validValue>
    </enum>
  </types>
</sbe:messageSchema>
```

```

        <validValue name="SystemIsUnavailable" >100</validValue>
        <validValue name="QuoteMsgIDIsNotUnique">101</validValue>
    </enum>

    <enum name="TimeInForceEnum" encodingType="uint8">
        <validValue name="Day" >0</validValue>
        <validValue name="IOC" >3</validValue>
    </enum>

    <enum name="SideEnum" encodingType="uint8">
        <validValue name="Unavailable" >0</validValue>
        <validValue name="Buy" >1</validValue>
        <validValue name="Sell" >2</validValue>
        <validValue name="BothSides" >89</validValue>
    </enum>

    <enum name="ModeEnum" encodingType="uint8">
        <validValue name="DontChangeQuoteQty" >0</validValue>
        <validValue name="ChangeQuoteQty" >1</validValue>
        <validValue name="CheckQuoteQtyAndCancelQuote" >2</validValue>
        <validValue name="FixStyleReplace" >3</validValue>
    </enum>

    <enum name="MatchTypeEnum" encodingType="uint8">
        <validValue name="AutoMatch" >4</validValue>
        <validValue name="AutoMatchWithLastLook" >10</validValue>
    </enum>

    <enum name="TradSesEventEnum" encodingType="uint16">
        <validValue name="SessionDataReady" >101</validValue>
        <validValue name="IntradayClearingFinished" >102</validValue>
        <validValue name="IntradayClearingStarted" >104</validValue>
        <validValue name="ClearingStarted" >105</validValue>
        <validValue name="ExtensionOfLimitsFinished" >106</validValue>
        <validValue name="BrokerRecalcFinished" >108</validValue>
        <validValue name="OtcSessionInited" >10100</validValue>
        <validValue name="OtcSessionStarted" >10101</validValue>
        <validValue name="OtcSessionSuspended" >10102</validValue>
        <validValue name="OtcSessionStoped" >10103</validValue>
        <validValue name="OtcSessionFinished" >10104</validValue>
    </enum>

    <enum name="SecurityTypeEnum" encodingType="uint8">
        <validValue name="Future" >0</validValue>
        <validValue name="Option" >1</validValue>
        <validValue name="Multileg" >2</validValue>
    </enum>

    <enum name="StatusEnum" encodingType="uint8">
        <validValue name="Matched" >0</validValue>
        <validValue name="WaitConfirm" >1</validValue>
        <validValue name="Confirmed" >2</validValue>
        <validValue name="Failed" >3</validValue>
        <validValue name="Success" >4</validValue>
    </enum>

    <enum name="RejectReasonEnum" encodingType="uint8">
        <validValue name="NotApplicable" >0</validValue>
        <validValue name="NotConfirmed" >1</validValue>
        <validValue name="ActiveSideError" >2</validValue>
        <validValue name="PassiveSideError" >3</validValue>
    </enum>

    <enum name="StreamExposureDurationEnum" encodingType="uint8">
        <validValue name="NotApplicable" >0</validValue>
        <validValue name="Duration30sec" >1</validValue>
        <validValue name="Duration60sec" >2</validValue>
        <validValue name="Duration90sec" >3</validValue>
        <validValue name="Duration120sec" >4</validValue>
    </enum>

    <enum name="SpeedBumpTypeEnum" encodingType="uint8">

```

```

    <validValue name="NotApplicable" >0</validValue>
    <validValue name="Duration200ms" >1</validValue>
    <validValue name="Duration500ms" >2</validValue>
    <validValue name="Duration1000ms" >3</validValue>
    <validValue name="Duration3000ms" >4</validValue>
</enum>

<enum name="CancelReasonEnum" encodingType="uint8">
    <validValue name="Deal" >1</validValue>
    <validValue name="LCDoesntHaveEnoughMoney" >2</validValue>
    <validValue name="CancelByLC" >3</validValue>
    <validValue name="TimeOut" >4</validValue>
    <validValue name="CancelByAdministrator" >5</validValue>
    <validValue name="EndOfTradingSession" >6</validValue>
</enum>

<set name="FlagsSet" encodingType="uint64">
    <choice name="Day" description="Quotes: Day" >0</choice>
    <choice name="IOC" description="Quotes: IOC" >1</choice>
    <choice name="Replace" description="Quotes:
        The record results from replacing the quote" >20</choice>
    <choice name="Cancel" description="Quotes:
        The record results from cancelling the quote" >21</choice>
    <choice name="MassCancel" description="Quotes:
        The record results from mass cancelling" >22</choice>
    <choice name="MultiLeg" description="Quotes: Multi leg"
        >27</choice>
    <choice name="FineOperation" description="Quotes:
        Flag of cancelling the quote because of fine" >36</choice>
    <choice name="ActiveSide" description="Trades: Quote initiator is aggressor"
        >41</choice>
    <choice name="PassiveSide" description="Trades: Quote initiator is passive"
        >42</choice>
    <choice name="TimeOut" description="Quotes: Delete quote due timeout"
        >49</choice>
    <choice name="AutoMatch" description="Quotes and trades:
        Auto match without last look" >50</choice>
</set>

<set name="StreamFlagsSet" encodingType="uint64">
    <choice name="AutoMatch" description="Auto match without last look">0</choice>
    <choice name="ClosedStream" description="Stream is closed">1</choice>
</set>

<composite name="Decimal5" description="Decimal">
    <type name="mantissa" description="mantissa" minValue="-9999999999999999"
        maxValue="9999999999999999" primitiveType="int64" presence="required" />
    <type name="exponent" description="exponent" presence="constant"
        primitiveType="int8">-5</type>
</composite>

<!-- do not change this one as it is hardcoded in serializer -->
<composite name="messageHeader" description="Template ID and length of message root">
    <type name="blockLength" primitiveType="uint16" />
    <type name="templateId" primitiveType="uint16" />
    <type name="schemaId" primitiveType="uint16" />
    <type name="version" primitiveType="uint16" />
</composite>
</types>

<!-- Session layer messages -->
<sbe:message name="Establish" id="5100">
    <field name="Timestamp" id="20204" type="TimeStamp" />
    <field name="KeepaliveInterval" id="20205" type="DeltaMillisecs" />
    <field name="Credentials" id="20206" type="String20" />
</sbe:message>

<sbe:message name="EstablishmentAck" id="5101">
    <field name="RequestTimestamp" id="20207" type="TimeStamp" />
    <field name="KeepaliveInterval" id="20205" type="DeltaMillisecs" />
    <field name="NextSeqNo" id="20208" type="UInt64" />
</sbe:message>

```



```

<sbe:message name="EstablishmentReject" id="5102">
  <field name="RequestTimestamp" id="20207" type="TimeStamp" />
  <field name="EstablishmentRejectCode" id="20209" type="EstablishmentRejectCodeEnum" />
</sbe:message>

<sbe:message name="Terminate" id="5103">
  <field name="TerminationCode" id="20210" type="TerminationCodeEnum" />
</sbe:message>

<sbe:message name="RetransmitRequest" id="5104">
  <field name="Timestamp" id="20204" type="TimeStamp" />
  <field name="FromSeqNo" id="20211" type="UInt64" />
  <field name="Count" id="20212" type="UInt32" />
</sbe:message>

<sbe:message name="Retransmission" id="5105">
  <field name="NextSeqNo" id="20208" type="UInt64" />
  <field name="RequestTimestamp" id="20207" type="TimeStamp" />
  <field name="Count" id="20212" type="UInt32" />
</sbe:message>

<sbe:message name="Sequence" id="5106">
  <field name="NextSeqNo" id="20208" type="UInt64" />
</sbe:message>

<sbe:message name="FloodReject" id="5107">
  <field name="QuoteMsgID" id="1166" type="UInt64"/>
  <field name="QueueSize" id="20213" type="UInt32"/>
  <field name="PenaltyRemain" id="20214" type="UInt32"/>
</sbe:message>

<sbe:message name="SessionReject" id="5108">
  <field name="QuoteMsgID" id="1166" type="UInt64"/>
  <field name="RefTagID" id="371" type="UInt32"/>
  <field name="SessionRejectReason" id="373" type="SessionRejectReasonEnum"/>
</sbe:message>

<!-- Application layer messages -->
<sbe:message name="NewStream" id="8007">
  <field name="QuoteMsgID" id="1166" type="UInt64" />
  <field name="MinQty" id="110" type="UInt64" />
  <field name="ExternalID" id="20027" type="UInt64" />
  <field name="SecurityID" id="48" type="Int32" />
  <field name="Side" id="54" type="SideEnum" />
  <field name="StreamExposureDuration" id="21011" type="StreamExposureDurationEnum"/>
  <field name="MatchType" id="574" type="MatchTypeEnum" />
  <field name="SpeedBumpType" id="21000" type="SpeedBumpTypeEnum" />
  <field name="Account" id="1" type="String7" />
  <field name="TextToLP" id="21001" type="String20" />
  <field name="Text" id="58" type="String20" />
</sbe:message>

<sbe:message name="CancelStream" id="8008">
  <field name="QuoteMsgID" id="1166" type="UInt64" />
  <field name="AuctionID" id="21002" type="UInt64" />
  <field name="Account" id="1" type="String7" />
</sbe:message>

<sbe:message name="RfsQuote" id="8009">
  <field name="QuoteMsgID" id="1166" type="UInt64" />
  <field name="AuctionID" id="21002" type="UInt64" />
  <field name="OfferPx" id="133" type="Decimal5" />
  <field name="OfferExternalID" id="20025" type="UInt64" />
  <field name="BidPx" id="132" type="Decimal5" />
  <field name="BidExternalID" id="20024" type="UInt64" />
  <field name="ExposureDuration" id="1629" type="UInt64" />
  <field name="MatchType" id="574" type="MatchTypeEnum" />
  <field name="Side" id="54" type="SideEnum" />
  <field name="Account" id="1" type="String7" />
  <field name="OfferText" id="21003" type="String20" />
  <field name="BidText" id="21004" type="String20" />

```

```

</sbe:message>

<sbe:message name="RfsQuoteMassCancel" id="8011">
  <field name="QuoteMsgID" id="1166" type="UInt64" />
  <field name="AuctionID" id="21002" type="UInt64" />
  <field name="ExternalID" id="20027" type="UInt64" />
  <field name="SecurityID" id="48" type="Int32" />
  <field name="Side" id="54" type="SideEnum" />
  <field name="Account" id="1" type="String7" />
</sbe:message>

<sbe:message name="RfsQuoteHit" id="8012">
  <field name="QuoteMsgID" id="1166" type="UInt64" />
  <field name="AuctionID" id="21002" type="UInt64" />
  <field name="Price" id="44" type="Decimal5" />
  <field name="Side" id="54" type="SideEnum" />
  <field name="Text" id="58" type="String20" />
</sbe:message>

<sbe:message name="RfsConfirmation" id="8013">
  <field name="QuoteMsgID" id="1166" type="UInt64" />
  <field name="ExecID" id="17" type="UInt64" />
</sbe:message>

<sbe:message name="NewStreamResponse" id="9011">
  <field name="QuoteMsgID" id="1166" type="UInt64" />
  <field name="Timestamp" id="20204" type="TimeStamp" />
  <field name="AuctionID" id="21002" type="UInt64" />
  <field name="MinQty" id="110" type="UInt64" />
  <field name="ExternalID" id="20027" type="UInt64" />
  <field name="SecurityID" id="48" type="Int32" />
  <field name="TradingSessionID" id="336" type="Int32" />
  <field name="StreamFlags" id="21005" type="StreamFlagsSet" />
  <field name="SecurityType" id="167" type="SecurityTypeEnum" />
  <field name="Side" id="54" type="SideEnum" />
  <field name="StreamExposureDuration" id="21011" type="StreamExposureDurationEnum" />
  <field name="SpeedBumpType" id="21000" type="SpeedBumpTypeEnum" />
  <field name="TextToLP" id="21001" type="String20" />
  <field name="Text" id="58" type="String20" />
  <field name="TagOfLC" id="21007" type="String64" />
</sbe:message>

<sbe:message name="NewStreamReject" id="9012">
  <field name="QuoteMsgID" id="1166" type="UInt64" />
  <field name="Timestamp" id="20204" type="TimeStamp" />
  <field name="QuoteRejectReason" id="300" type="Int32" />
</sbe:message>

<sbe:message name="CancelStreamResponse" id="9013">
  <field name="QuoteMsgID" id="1166" type="UInt64" />
  <field name="Timestamp" id="20204" type="TimeStamp" />
  <field name="AuctionID" id="21002" type="UInt64" />
  <field name="MinQty" id="110" type="UInt64" />
  <field name="ExternalID" id="20027" type="UInt64" />
  <field name="ExecID" id="17" type="UInt64" />
  <field name="SecurityID" id="48" type="Int32" />
  <field name="TradingSessionID" id="336" type="Int32" />
  <field name="Side" id="54" type="SideEnum" />
  <field name="StreamExposureDuration" id="21011" type="StreamExposureDurationEnum" />
  <field name="StreamFlags" id="21005" type="StreamFlagsSet" />
  <field name="SecurityType" id="167" type="SecurityTypeEnum" />
  <field name="SpeedBumpType" id="21000" type="SpeedBumpTypeEnum" />
  <field name="CancelReason" id="21008" type="CancelReasonEnum" />
  <field name="TextToLP" id="21001" type="String20" />
  <field name="Text" id="58" type="String20" />
  <field name="TagOfLC" id="21007" type="String64" />
</sbe:message>

<sbe:message name="CancelStreamReject" id="9014">
  <field name="QuoteMsgID" id="1166" type="UInt64" />
  <field name="Timestamp" id="20204" type="TimeStamp" />
  <field name="QuoteRejectReason" id="300" type="Int32" />

```

```

</sbe:message>

<sbe:message name="RfsQuoteResponse" id="9015">
  <field name="QuoteMsgID" id="1166" type="UInt64" />
  <field name="Timestamp" id="20204" type="TimeStamp" />
  <field name="AuctionID" id="21002" type="UInt64" />
  <field name="SecondaryQuoteID" id="1751" type="UInt64" />
  <field name="QuoteSize" id="20026" type="UInt64" />
  <field name="Price" id="44" type="Decimal5" />
  <field name="ExternalID" id="20027" type="UInt64" />
  <field name="ExposureDuration" id="1629" type="UInt64" />
  <field name="Flags" id="20215" type="FlagsSet" />
  <field name="SecurityID" id="48" type="Int32" />
  <field name="TradingSessionID" id="336" type="Int32" />
  <field name="SecurityType" id="167" type="SecurityTypeEnum" />
  <field name="Side" id="54" type="SideEnum" />
  <field name="CodeOfLP" id="21009" type="String20" />
  <field name="Text" id="58" type="String20" />
</sbe:message>

<sbe:message name="RfsQuoteReplaceResponse" id="9016">
  <field name="QuoteMsgID" id="1166" type="UInt64" />
  <field name="Timestamp" id="20204" type="TimeStamp" />
  <field name="AuctionID" id="21002" type="UInt64" />
  <field name="SecondaryQuoteID" id="1751" type="UInt64" />
  <field name="QuoteSize" id="20026" type="UInt64" />
  <field name="Price" id="44" type="Decimal5" />
  <field name="ExternalID" id="20027" type="UInt64" />
  <field name="PrevSecondaryQuoteID" id="20034" type="UInt64" />
  <field name="ExposureDuration" id="1629" type="UInt64" />
  <field name="Flags" id="20215" type="FlagsSet" />
  <field name="SecurityID" id="48" type="Int32" />
  <field name="TradingSessionID" id="336" type="Int32" />
  <field name="SecurityType" id="167" type="SecurityTypeEnum" />
  <field name="Side" id="54" type="SideEnum" />
  <field name="CodeOfLP" id="21009" type="String20" />
</sbe:message>

<sbe:message name="RfsQuoteReject" id="9017">
  <field name="QuoteMsgID" id="1166" type="UInt64" />
  <field name="Timestamp" id="20204" type="TimeStamp" />
  <field name="QuoteRejectReason" id="300" type="Int32" />
  <field name="Side" id="54" type="SideEnum" />
</sbe:message>

<sbe:message name="RfsQuoteCancelResponse" id="9018">
  <field name="QuoteMsgID" id="1166" type="UInt64" />
  <field name="Timestamp" id="20204" type="TimeStamp" />
  <field name="AuctionID" id="21002" type="UInt64" />
  <field name="SecondaryQuoteID" id="1751" type="UInt64" />
  <field name="QuoteSize" id="20026" type="UInt64" />
  <field name="ExternalID" id="20027" type="UInt64" />
  <field name="Flags" id="20215" type="FlagsSet" />
  <field name="TradingSessionID" id="336" type="Int32" />
</sbe:message>

<sbe:message name="RfsQuoteMassCancelAck" id="9020">
  <field name="QuoteMsgID" id="1166" type="UInt64" />
  <field name="Timestamp" id="20204" type="TimeStamp" />
  <field name="TotNoCxldQuotes" id="1168" type="Int32" />
  <field name="TotNoSpeedBumpQuotes" id="21012" type="Int32" />
  <field name="QuoteRejectReason" id="300" type="Int32" />
</sbe:message>

<sbe:message name="RfsBestQuoteUpdate" id="9021">
  <field name="AuctionID" id="21002" type="UInt64" />
  <field name="SecondaryQuoteID" id="1751" type="UInt64" />
  <field name="QuoteSize" id="20026" type="UInt64" />
  <field name="Price" id="44" type="Decimal5" />
  <field name="Side" id="54" type="SideEnum" />
  <field name="MatchType" id="574" type="MatchTypeEnum" />
</sbe:message>

```

```

<sbe:message name="RfsQuoteHitAck" id="9022">
  <field name="QuoteMsgID" id="1166" type="UInt64" />
  <field name="Timestamp" id="20204" type="TimeStamp" />
  <field name="SecondaryQuoteID" id="1751" type="UInt64" />
  <field name="QuoteRejectReason" id="300" type="Int32" />
</sbe:message>

<sbe:message name="RfsConfirmationAck" id="9023">
  <field name="QuoteMsgID" id="1166" type="UInt64" />
  <field name="Timestamp" id="20204" type="TimeStamp" />
  <field name="ExecID" id="17" type="UInt64" />
  <field name="QuoteRejectReason" id="300" type="Int32" />
</sbe:message>

<sbe:message name="RfsExecutionReport" id="9024">
  <field name="QuoteMsgID" id="1166" type="UInt64" />
  <field name="Timestamp" id="20204" type="TimeStamp" />
  <field name="AuctionID" id="21002" type="UInt64" />
  <field name="SecondaryQuoteID" id="1751" type="UInt64" />
  <field name="LastPx" id="31" type="Decimal5" />
  <field name="LastQty" id="32" type="UInt64" />
  <field name="ExposureDuration" id="1629" type="UInt64" />
  <field name="ExternalID" id="20027" type="UInt64" />
  <field name="ExecID" id="17" type="UInt64" />
  <field name="TrdMatchID" id="880" type="Int64" />
  <field name="OrderID" id="37" type="Int64" />
  <field name="TradingSessionID" id="336" type="Int32" />
  <field name="SecurityID" id="48" type="Int32" />
  <field name="OrdRejReason" id="103" type="Int32" />
  <field name="SecurityType" id="167" type="SecurityTypeEnum" />
  <field name="Side" id="54" type="SideEnum" />
  <field name="Status" id="20030" type="StatusEnum" />
  <field name="RejectReason" id="20033" type="RejectReasonEnum" />
  <field name="CodeOfLP" id="21009" type="String20" />
  <field name="Text" id="58" type="String20" />
</sbe:message>

<sbe:message name="EmptyBook" id="9009">
  <field name="Timestamp" id="20204" type="TimeStamp" />
  <field name="TradingSessionID" id="336" type="Int32" />
</sbe:message>

<sbe:message name="SystemEvent" id="9010">
  <field name="Timestamp" id="20204" type="TimeStamp" />
  <field name="TradingSessionID" id="336" type="Int32" />
  <field name="TradSesEvent" id="1368" type="TradSesEventEnum"/>
</sbe:message>
</sbe:messageSchema>

```