FXSpotStream

FSS Rules of Engagement

Version 1.2.6

Document History

DOCUMENT VERSION NO.	DATE	AUTHOR	CHANGE DESCRIPTION

0.1	2012/03/15	Fabien Romeo	Initial Draft
0.2	2012/03/22	Rina RATSIMBAZAFY	Review
1.0.0	2012/03/22	Rina RATSIMBAZAFY	First version
1.0.1	2012/06/28	Eric Deshayes	Added supported currency pairs, their precision and settlement date. Changed currency (15) on NewOrderSingle to optional Fixed FullAmount Market Data Request example
1.0.2	2012/08/07	Eric Deshayes	Corrected Market Data Request section and its samples.
1.0.3	2012/09/17	Eric Deshayes	Added group hierarchy
1.0.4	2012/10/22	Fabien Romeo	Added Liquidity Provider ID table. Added NOS sending examples. Added workflow diagram. Added precision on ResetOnLogon for trading session.
1.0.5	2012/10/29	Fabien Romeo	Added 0 value for MarketDepth in MarketDataRequest
1.0.6	20012/11/05	Eric Deshayes	Fixed typo.
1.0.7	2012/11/20	Eric Deshayes	Added new currency pairs. Amended LastMkt reference in ExecutionReport
1.0.8	2012/12/10	Fabien Romeo Eric Deshayes	Fixed typo. QuickFIX/J sample Added new currency pairs.currencies
1.0.9	2012/12/13	Fabien Romeo	Added Account field in NewOrderSingle and ExecutionReport
1.0.10	2013/01/05	Eric Deshayes	Changed account description.
1.0.11	2013/01/08	Fabien Romeo	Added Market Data Incremental Message
1.0.12	2013/01/18	Eric Deshayes	Market Data Incremental feedbacks
1.0.13	2013/03/04	Eric Deshayes	MDEntryRefID unicitiy by side amendement
1.0.14	2013/03/11	Eric Deshayes	Clarification on LastMkt(30)
1.0.15	2013/04/09	Eric Deshayes Dan Smith	Addition of currency pairs
1.0.16	2013/04/10	Gael Hanquez	Addition of TraderID

1.0.17	2013/05/24	Fabien Romeo	Added UBS Added distribution
1.0.18	2013/06/18	Fabien Romeo	Added precisions for BAML & GS Add MDEntryPositionNo in type W
1.0.19	2013/06/19	Fabien Romeo	Fixed typo in type V with PartyRole
1.0.20	2013/07/15	Dan Smith	Fixed typos in Crossess matrix
1.0.21	2013/09/04	Fabien Romeo	Added Precious Metals Changed SecurityList for Precious Metals Updated FSS_FIX44.xml in appendix
1.0.22	2013/11/07	Eric Deshayes Fabien Romeo	Mid Rate term support Term currency trading Added note on MS book sweeping Fixed typos Removed fix dictionary
1.0.23	2014/01/06	Fabien Romeo	Added primary/secondary IP notice
1.0.24	2014/02/07	Fabien Romeo	Added CNH/SGD and SGD/CNH
1.0.25	2014/02/18	Fabien Romeo	Added SettlDate (64) on market data entries
1.0.26	2014/03/11	Fabien Romeo	Added SettlDate (64) on market data
1.0.27	2014/05/12	Fabien Romeo	Added BNP
1.0.28	2014/07/9	Dan Smith	Updated CCY pair matrix
1.1.0	2014/08/29	Fabien Romeo	Added RFS FWD SWAP
1.1.1	2014/10/30	Fabien Romeo	Removed ESP FWD Added back fix dictionary Added optional MDEntryOrigTime
1.1.2	2014/11/25	Fabien Romeo	Added ESP FWD
1.1.3	2014/12/08	Fabien Romeo	Amended ESP FWD instruments
1.1.4	2014/12/25	Fabien Romeo	Changed RFS broken date handling Added Slippage on NOS
1.1.5	2015/01/15	Fabien Romeo	Added SwapPoints in MassQuote
1.1.6	2015/01/27	Fabien Romeo	Added FIX message examples for RFS
1.1.7	2015/01/29	Fabien Romeo	Fixed typo (side values on massquote)
1.1.8	2015/02/13	Fabien Romeo	Added spotRate on NOS for FWD – Fixed typos
1.1.9	2015/02/20	Fabien Romeo	Added IOC and VWAP orders
1.1.10	2015/03/05	Houssem Maalej	Added note on Slippage offset

1.1.11	2015/03/19	Fabien Romeo	Addition of CS
1.1.12	2015/04/21	Fabien Romeo	Added note on IOC Changed MDEntryDate as optional on MDS
1.1.13	2015/05/27	Fabien Romeo	Added SCB Added LP capabilities
1.2.0	2015/06/29	Fabien Romeo	Added SCB precisions Addition of limit orders
1.2.1	2015/07/24	Fabien Romeo	New SCB precisions
1.2.2	2015/08/25	Fabien Romeo	Added new tenors Fixed currency to base ccy for limit orders Added new status for limit orders Added OrigClientID in ER
1.2.3	2015/09/04	Fabien Romeo	Added expired status Added ordType limit on MDR Fixed typos
1.2.4	2015/09/28	Fabien Romeo	JP aggressive DMA orders Remove UBS from DMA Updated FIX dictionary Minor fixes
1.2.5	2015/10/21	Fabien Romeo	Addition of BTMU
1.2.6	2016/01/21	Houssem Maalej	Addition of Trading Session List

Contents

Document History 1
1) Introduction 8
1.1. Purpose 8
2) Summary 8
2.1. Executable Streaming Pricing (ESP) 8
2.2. Request For Streams (RFS) 9
2.3. Liquidity Providers 11
2.3.1. Liquidity Providers Execution Capabilities 11
3) Connectivity 12
3.1. Simulator 12
3.2. UAT 12
3.3. Production 12
4) Product Information 13
4.1. Executable Streaming Prices (ESP) 13
4.1.1. Option 1: Full amount 13
4.1.2. Option 2: Passthrough 13
4.2. Request For Streams (RFS) 13
4.2.1. Forward 14
4.2.2. Swap 16
5) Tenors (for RFS Forwards and Swaps) 20
6) Currency pairs 21
6.1. Supported currency pairs 21
6.2. FX Currency Pairs 22
6.3. FX Currency Pairs with Tenors for ESP Forwards 23
6.4. FX Precisions 23
6.4.1. General FX Precisions (COBA, CITI, HSBC, JPMC, MS, UBS, BNP) 24
6.4.2. FX Precisions for BAML 26
6.4.3. FX Precisions for GS 26
6.4.4. FX Precisions for CS 27
6.4.5. FX Precisions for SCB 28
6.5. Precious Metals Currency Pairs and Precisions 30
7) Session 31

7.1. Logon Message Definition (Type A) 31
7.2. Logout Message Definition (Type 5) 31
8) Security 32
8.1. SecurityListRequest Message Definition (Type x) 32
8.2. SecurityList (Type y) 32
9) Market Data 33
9.1. Subscribe to Market Data 33
9.2. Unsubscribe from Market Data 33
9.3. Request for Trading: Limit Orders Case 34
9.4. MarketDataRequest Message Definition (Type V) 35
9.5. MarketDataRequestReject Message Definition (Type Y) 36
9.6. Streaming Market Data 36
9.7. MarketDataSnapshotFullRefresh Message Definition (Type W) 37
9.7.1. PreTrade Mid-Market Rate 38
9.8. MarketDataIncrementalRefresh Message Definition (Type X) 38
9.8.1. Pre-Trade Mid-Market Rate 39
9.9. Market Data Incremental Refresh Examples 39
10) Trading Session List (LP Status) 43
10.1. Trading Session List Request (Type BI) 43
10.2. Trading Session List (Type BJ) 44
10.3. Trading Session list Update Report (Type BS) 44
11) Orders 44
11.1. Submitting Orders 45
11.2. Previously-quoted Orders 45
11.3. Limit Orders 45
11.3.1. FSS Limit Orders 46
11.3.2. Direct Market Access Strategy (DMA) 46
11.4. NewOrderSingle Message Definition (Type D) 46
11.5. Submitting Order Examples 48
11.5.1. Buy order 48
11.5.2. Sell Order 48
11.5.3. Buy order on the term currency 48
11.5.4. Sell order on the term currency 48

11.6. Executions 49
11.7. ExecutionReport Message Definition (Type 8) 49
11.8. Order Execution Examples 52
11.8.1. Order Rejected 52
11.8.2. Order Filled 52
11.8.3. Order Partially Filled (IOC orders) 52
11.9. Canceling Orders 53
11.9.1. OrderCancelRequest (Type F) 53
11.9.2. OrderCancelReject (Type 9) 53
11.10. RFS 54
11.10.1. QuoteRequest (Type R) 54
11.10.2. MassQuote (Type i) 55
11.10.3. MassQuoteAcknowledgement (Type b) 56
11.10.4. QuoteResponse (Type AJ) 56
11.10.5. QuoteCancel (Type Z) 58
11.10.6. RFS Message Examples 58
12) Standard FIX Message Definitions 61
12.1. Standard Message Header 61
12.2. Standard Message Trailer 61
12.3. ResendRequest (Type 2) 61
12.4. SequenceReset (Type 4) 62
12.5. Reject (Type 3) 62
12.6. TestRequest Message Definition (Type 1) 62
12.7. Heartbeat Message Definition (Type 0) 62
13) Status and Error Message 62
13.1. Streaming Error 62
13.2. Trading Error 62
Appendix I. QuickFIX/J code sample 64

1) Introduction

1.1. Purpose

This document has been prepared with the purpose of providing a description of how FSS FIX Protocol engine is designed and how the connectivity with FSS FX Trading Platform can be established for the purpose of electronic Foreign Exchange Trading.

2) Summary

FSS FIX sessions are based on FIX 4.4 and follow an order driven market model where clients request tradable FX price streams, and submit orders with a reference to a previously sent price or limit orders.

Two methods are supported for receiving prices and executing trades:

- Executable Streaming Pricing (ESP)
- Request For Streams (RFS)

Each method requires dedicated FIX sessions. In other words, it is not possible to request quotes (RFS) on ESP sessions, and conversely, it is not possible to subscribe to an ESP on RFS sessions.

2.1. Executable Streaming Pricing (ESP)

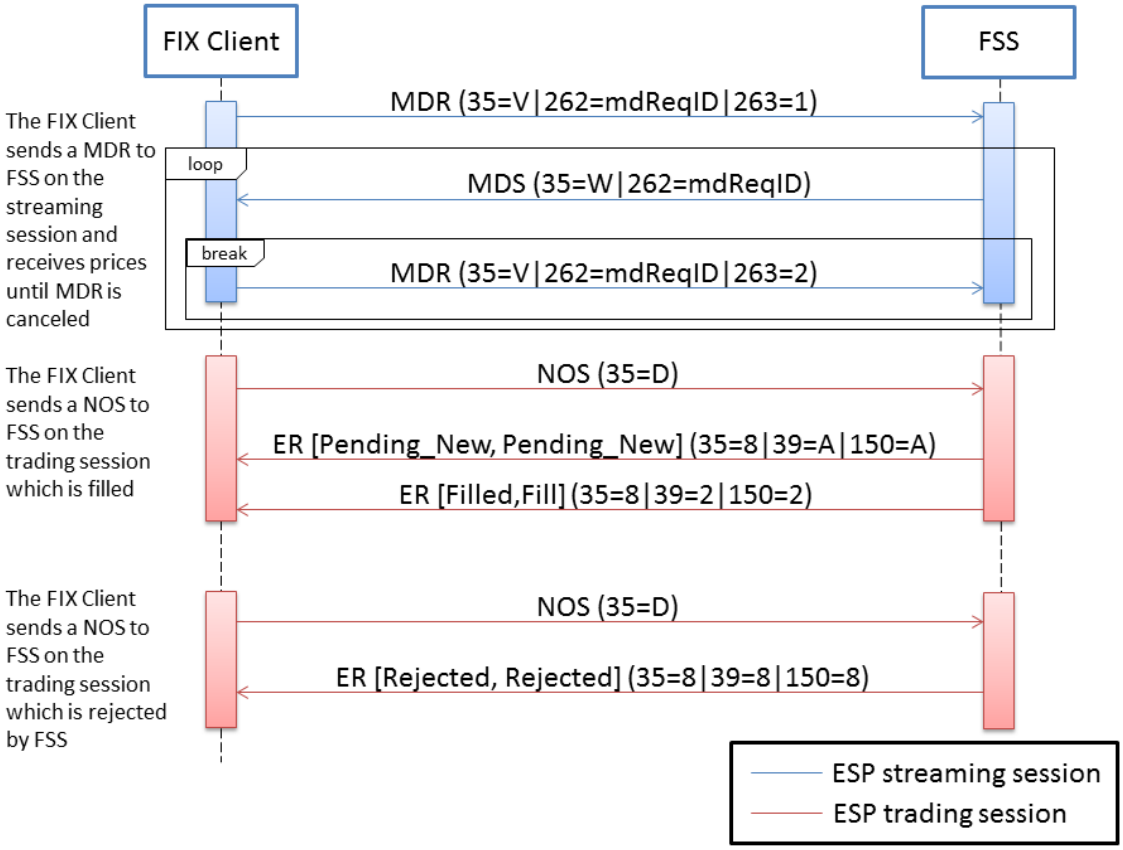
The following product is supported in the ESP mode:

- Spot
- Forward (only a predefined set of currency pairs and tenors – see section 6.3)

For ESP, there are two separate FIX sessions:

- **Market Data/Streaming Session:** Session for sending/receiving security information and rates.
- **Orders/Trading Session:** Session for order submission and trade execution. Messages are transactional and persistent with no lost messages allowed. The server resends order session messages in response to a resend request from the client. In order to enable this latter feature, the client must logon on (35=A) with tag 141=N (see section 7.1).

The following diagram depicts the basic workflow:



2.2. Request For Streams (RFS)

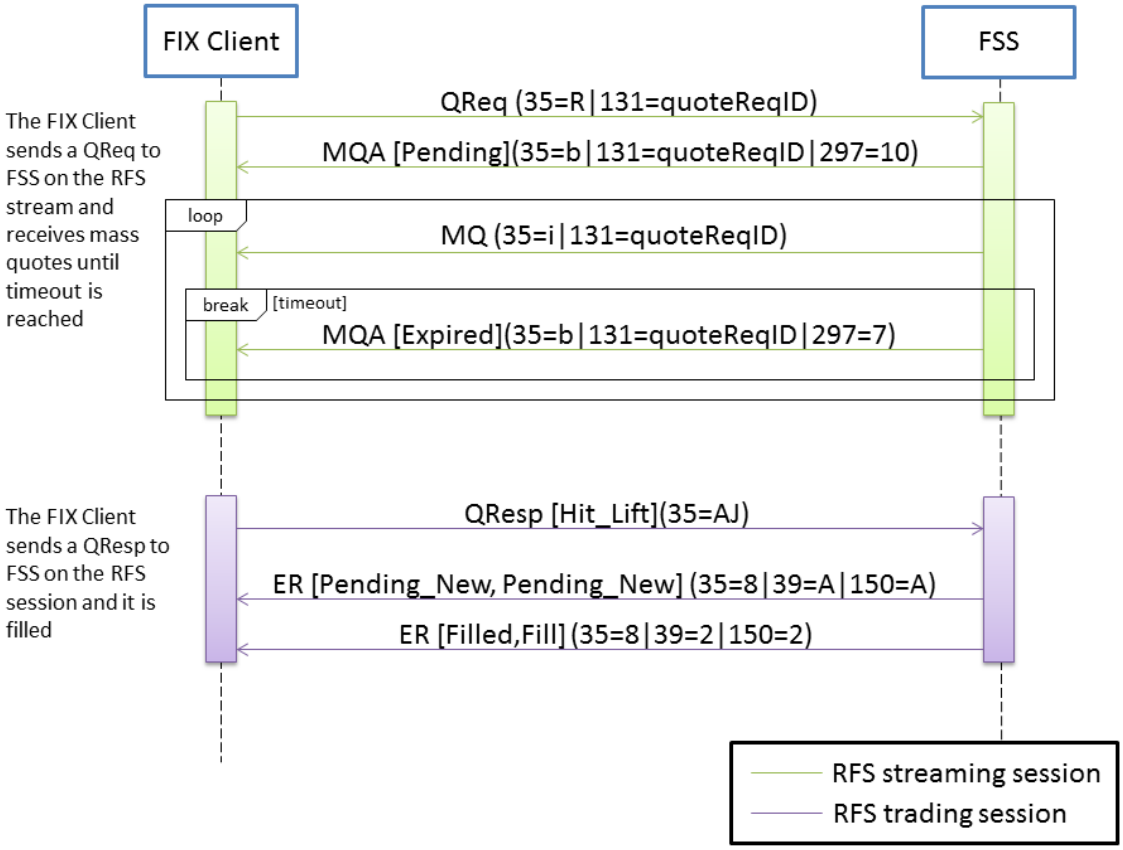
The RFS mode supports the following products:

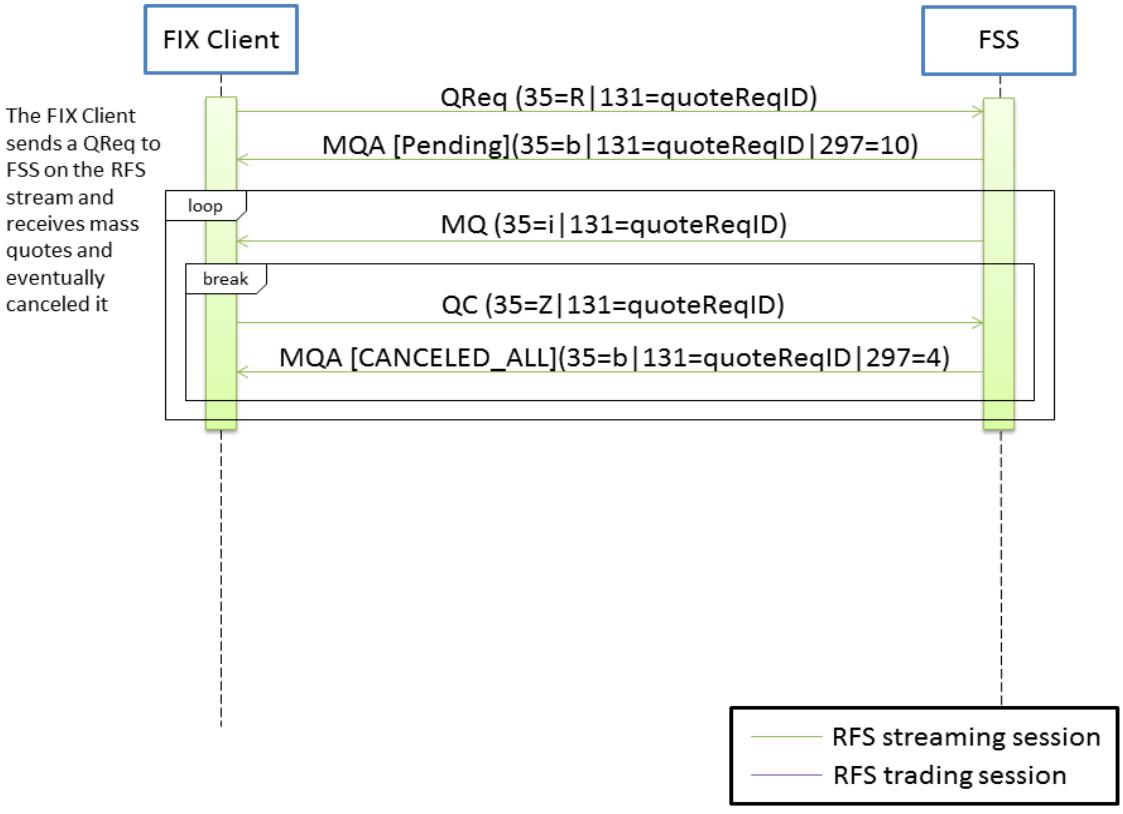
- Forward
- Swap

For RFS, prices and trades are exchanged on two separate sessions:

- **RFS Streaming Session**: Session for requesting/receiving quotes.
- **RFS Trading Session**: Session for trading on quotes. Messages are transactional and persistent with no lost messages allowed. The server resends order session messages in response to a resend request from the client. In order to enable this latter feature, the client must logon on (35=A) with tag 141=N (see section 7.1).

The following diagrams depict the basic workflows:


FIX Client
FSS
The FIX Client sends a QReq to FSS on the RFS stream and receives mass quotes until timeout is reached
QReq (35=R|131=quoteReqID)
MQA [Pending](35=b|131=quoteReqID|297=10)
loop
MQ (35=i|131=quoteReqID)
break
[timeout]
MQA [Expired](35=b|131=quoteReqID|297=7)
The FIX Client sends a QResp to FSS on the RFS session and it is filled
QResp [Hit_Lift](35=AJ)
ER [Pending_New, Pending_New] (35=8|39=A|150=A)
ER [Filled,Fill] (35=8|39=2|150=2)
RFS streaming session
RFS trading session


FIX Client
FSS
The FIX Client sends a QReq to FSS on the RFS stream and receives mass quotes and eventually canceled it
QReq (35=R|131=quoteReqID)
MQA [Pending](35=b|131=quoteReqID|297=10)
loop
MQ (35=i|131=quoteReqID)
break
QC (35=Z|131=quoteReqID)
MQA [CANCELED_ALL](35=b|131=quoteReqID|297=4)
RFS streaming session
RFS trading session

2.3. Liquidity Providers

FSS FX Trading Platform provides liquidity from the following banks:

Liquidity Provider ID	Liquidity Provider Name
BAML	Bank of America
BNP	BNP Paribas
BTMU	Bank of Tokyo-Mitsubishi UFJ
CITI	Citi
COBA	CommerzBank
CS	Credit Suisse
GS	Goldman Sachs
HSBC	HSBC
JPMC	JPMorgan Chase
MS	Morgan Stanley
SCB	Standard Chartered
UBS	UBS

The codes above are used to identify the Liquidity Providers in the different FIX messages.

2.3.1. Liquidity Providers Execution Capabilities

Previously-Quoted Orders

The following table lists the different execution capabilities supported by the liquidity providers for previously-quoted orders:

LP	FOK	IOC[3]	Slippage	VWAP
BAML	✓		✓	
BNP	✓			
BTMU	✓			
CITI	✓			✓
COBA	✓			✓
CS	✓			✓
GS	✓			
HSBC	✓		✓	✓
JPMC	✓			✓
MS	✓	✓	✓	✓
SCB	✓			
UBS	✓	✓		✓

- FOK: Fill-Or-Kill orders, the execution is either fully-filled or rejected.
- IOC: Immediate-Or-Cancel orders, the execution can be either fully-filled, partially-filled or rejected. Note that multiple partially-filled execution reports can be received.
- Slippage: allows the client to specify a discretion offset up to which he will accept the execution in his disfavor.
- VWAP: in passthrough mode, allows the client to send a unique order with a VWAP prices calculated on the target liquidity provider's entries.

For the technical details of each capability, please refer to section 11).

Limit Orders (DMA strategy)

The liquidity providers supporting Limit Orders via DMA strategy are listed in the following table with the available Time-In-Forces:

LP	Aggressive orders		Resting orders	
	FOK	IOC	GTC	GTD
GS			✓	✓
JPMC	✓			
MS		✓	✓	✓

- Aggressive orders:
  - FOK: Fill-Or-Kill orders, the execution is either fully-filled or rejected.
  - IOC: Immediate-Or-Cancel orders, the execution can be either fully-filled, partially-filled or rejected.
- Resting orders:
  - GTC: Good-Till-Cancel orders rest on the LP side until cancelation or execution.
  - GTD: Good-Till-Date orders rest on the LP side until the given date or execution.

3) Connectivity

To connect to FSS FIX engine, client can use either a VPN connection or an extranet via extranet provider (Radianz, TNS …).

3.1. Simulator

This environment should be used for the purposes of testing on the FSS Simulators.

3.2. UAT

This environment should be used for the purposes of testing with FSS FIX Engine and the Bank UAT environment.

3.3. Production

This environment should be used by client to connect with FSS FIX Engine and Bank Production environment. The production environment is composed of a primary server and a secondary/redondant server.

In the event that the primary server fails on the FXSpotStream network, we will automatically fail over to the secondary/redundant server. During the onboarding process, clients are provided with IP information for both the primary and secondary servers. These IPs should be configured on the client's network, and it is recommended that automatic (non-manual) failover procedures are put in place in an effort minimize the potential downtime.

4) Product Information

Trading on the platform follows an order driven market model where clients request tradable price streams and submit orders with a reference to a previously quoted price.

There are a number of products that are supported on the platform, and the following is an overview on how to trade them using FIX. There are further details and examples in the appropriate sections.

4.1. Executable Streaming Prices (ESP)

4.1.1. Option 1: Full amount

Full amount trading is done by clients requesting for market data on a specified currency pair and a specified set of quantities.

- To subscribe to full amount streams, clients should submit a "MarketDataRequest" (Type V) message with the interested currency pair and the interested quantity(ies). Depending on the available liquidity and on the characteristics of the client's account profile, clients should begin receiving "MarketDataSnapshotFullRefresh" (Type W) messages with prices and sizes for the interested currency pair and the interested quantity(ies). The price for each requested quantity is calculated by aggregating available quantities for each stream on which the client is allowed to trade, and then by selecting the best price.

- To trade on a quoted price, clients should submit a "NewOrderSingle" message with the "MDEntryID" tag set to the respective "MarketDataSnapshotFullRefresh" "MDEntryID" tag. Order responses will be encapsulated in "ExecutionReport" messages.

4.1.2. Option 2: Passthrough

Passthrough is done by clients requesting for market data on a specified currency pair and a specified depth.

- To subscribe to pass-through streams, clients should submit a "MarketDataRequest" (Type V) message with the interested currency pair and the interested depth. Depending on the available liquidity and on the characteristics of the client's account profile, clients should start receiving "MarketDataSnapshotFullRefresh" (Type W) messages with prices and sizes for the interested currency pair. The best prices are shown first. The depth corresponds to the maximum number of entries that can be shown.

- To trade on a quoted price, clients should submit a "NewOrderSingle" message with the "QuoteID" tag set to the respective "MarketDataSnapshotFullRefresh" "MDEntryID" tag. Order responses will be encapsulated in "ExecutionReport" messages.

4.2. Request For Streams (RFS)

RFS allows clients to request Forwards and Swaps on a specified currency pair.

The QuoteRequest can be either one-way or two-way.

- To subscribe to a quote stream, clients should submit a “QuoteRequest” (Type R) message with the interested currency pair and the interested amount and the settlement type as defined in section 5). The stream is open for a maximum of 120s or until a “QuoteResponse” is sent by the client.

- To trade on quote, clients should submit a “QuoteResponse” message “Hit/Lift” with both “QuoteReqID” and “QuoteID”. Only one response could be sent per quote resquest. After the response has been sent the quote stream is closed and no new quote will be sent to the client.

4.2.1. Forward

As a client, I want to either do one-way or two-way quotes.

For two way quotes, the currency corresponds to the quantity.

As a client, I want to send a quote request to:

- either buy 1M EUR and sell the equivalent in USD
- either sell 1M EUR and buy the equivalent in USD

Symbol(55)=EUR/USD, Currency(15)=EUR, Quantity=1M

-> I will receive both bid and offer quotes with prices for quantities in EUR.

- on the bid entries, As a client, I can either

-> sell 500k EUR and buy the equivalent amount in USD: Symbol(55)=EUR/USD, Side(54)=Sell(2), Currency(15)=EUR, Quantity=500k

-> buy 500k USD and sell the equivalent amount in EUR: Symbol(55)=EUR/USD, Side(54)=Buy(1), Currency(15)=USD, Quantity=500k

- on the offer entries, As a client, I can either

-> buy 500k EUR and sell the equivalent amount in USD: Symbol(55)=EUR/USD, Side(54)=Buy(1), Currency(15)=EUR, Quantity=500k

-> sell 500k USD and buy the equivalent amount in EUR: Symbol(55)=EUR/USD, Side(54)=Sell(2), Currency(15)=USD, Quantity=500k

As a client, I want to send a quote request to:

- either buy 1M USD and sell the equivalent in EUR
- either sell 1M USD and buy the equivalent in EUR

Symbol(55)=EUR/USD, Currency(15)=USD, Quantity=1M

-> I will receive both bid and offer quotes with prices for quantities in USD.

- on the bid entries, As a client, I can either

-> sell 500k EUR and buy the equivalent amount in USD: Symbol(55)=EUR/USD, Side(54)=Sell(2), Currency(15)=EUR, Quantity=500k

-> buy 500k USD and sell the equivalent amount in EUR: Symbol(55)=EUR/USD, Side(54)=Buy(1), Currency(15)=USD, Quantity=500k

- on the offer entries, As a client, I can either

-> buy 500k EUR and sell the equivalent amount in USD: Symbol(55)=EUR/USD, Side(54)=Buy(1), Currency(15)=EUR, Quantity=500k

-> sell 500k USD and buy the equivalent amount in EUR: Symbol(55)=EUR/USD, Side(54)=Sell(2), Currency(15)=USD, Quantity=500k

For a one-way quote, I will provide a Side that corresponds to the action I want to do on the specified currency:

As a client, I want to send a quote request to buy 1M EUR and sell the equivalent in USD: Symbol(55)=EUR/USD, Side(54)=Buy(1), Currency(15)=EUR, Quantity=1M

-> I will receive offer entries with a price for a quantity defined in EUR.

-> As a client, I want to buy 500k EUR and sell the equivalent USD: Symbol(55)=EUR/USD, Side(54)=Buy(1), Currency(15)=EUR, Quantity=500k

-> As a client, I want to sell 500k USD and buy the equivalent EUR: Symbol(55)=EUR/USD, Side(54)=Sell(2), Currency(15)=USD, Quantity=500k

As a client, I want to send a quote request to buy 1M USD and sell the equivalent in EUR: Symbol(55)=EUR/USD, Side(54)=Buy(1), Currency(15)=USD, Quantity=1M

-> I will receive bid entries with a price for a quantity defined in USD.

-> As a client, I want to sell 500k EUR and buy the equivalent USD: Symbol(55)=EUR/USD, Side(54)=Sell(2), Currency(15)=EUR, Quantity=500k

-> As a client, I want to buy 500k USD and sell the equivalent EUR: Symbol(55)=EUR/USD, Side(54)=Buy(1), Currency(15)=USD, Quantity=500k

As a client, I want to send a quote request to sell 1M EUR and buy the equivalent in USD: Symbol(55)=EUR/USD, Side(54)=Sell(2), Currency(15)=EUR, Quantity=1M

-> I will receive bid entries with a price for a quantity defined in EUR.

-> As a client, I want to sell 500k EUR and buy the equivalent USD: Symbol(55)=EUR/USD, Side(54)=Sell(2), Currency(15)=EUR, Quantity=500k

-> As a client, I want to buy 500k USD and sell the equivalent EUR: Symbol(55)=EUR/USD, Side(54)=Buy(1), Currency(15)=USD, Quantity=500k

As a client, I want to send a quote request to sell 1M USD and buy the equivalent in EUR: Symbol(55)=EUR/USD, Side(54)=Sell(2), Currency(15)=USD, Quantity=1M

-> I will receive offer entries with a price for a quantity defined in USD.

-> As a client, I want to buy 500k EUR and sell the equivalent USD: Symbol(55)=EUR/USD, Side(54)=Buy(1), Currency(15)=EUR, Quantity=500k

-> As a client, I want to sell 500k USD and buy the equivalent EUR: Symbol(55)=EUR/USD, Side(54)=Sell(2), Currency(15)=USD, Quantity=500k

4.2.2. Swap

As a client, I want to either do one-way or two-way quotes.

For two way quotes, the currency corresponds to the OrderQty.

The action ALWAYS refers to the far leg from a client pint of view.

As a client, I want to send a quote request to :

- either buy 1M EUR Far and sell the equivalent in USD Far, Sell 2M in EUR Near and Buy the equivalent in USD Near
- either sell 1M EUR and buy the equivalent in USD, Buy 2M in EUR Near and Sell the equivalent in USD Near

Symbol(55)=EUR/USD, Currency(15)=EUR, Quantity=1M, Quantity2=2M

-> I will receive both bid and offer quotes with prices for quantities in EUR.

- on the bid entries, As a client, I can either

-> sell 500k EUR Far and buy the equivalent amount in USD Far, buy 700k EUR Near and sell the equivalent amount in USD Near: Symbol(55)=EUR/USD, Side(54)=Sell(2), Currency(15)=EUR, Quantity=500k, Quantity2=700K

-> buy 500k USD Far and sell the equivalent amount in EUR Far, sell 700k USD Near and buy the equivalent amount in USD Near: Symbol(55)=EUR/USD, Side(54)=Buy(1), Currency(15)=USD, Quantity=500k, Quantity2=700K

- on the offer entries, As a client, I can either

-> buy 500k EUR Far and sell the equivalent amount in USD Far, sell 700k EUR Near and buy the equivalent amount in USD Near: Symbol(55)=EUR/USD, Side(54)=Buy(1), Currency(15)=EUR, Quantity=500k, Quantity2=700K

-> sell 500k USD Far and buy the equivalent amount in EUR Far, buy 700k USD Near and sell the equivalent amount in EUR Near: Symbol(55)=EUR/USD, Side(54)=Sell(2), Currency(15)=USD, Quantity=500k, Quantity2=700K

As a client, I want to send a quote request to:

- either buy 1M USD and sell the equivalent in EUR
- either sell 1M USD and buy the equivalent in EUR

Symbol(55)=EUR/USD, Currency(15)=USD, Quantity=1M

-> I will receive both bid and offer quotes with prices for quantities in USD.

- on the bid entries, As a client, I can either

-> sell 500k EUR and buy the equivalent amount in USD: Symbol(55)=EUR/USD, Side(54)=Sell(2), Currency(15)=EUR, Quantity=500k

-> buy 500k USD and sell the equivalent amount in EUR: Symbol(55)=EUR/USD, Side(54)=Buy(1), Currency(15)=USD, Quantity=500k

- on the offer entries, As a client, I can either

-> buy 500k EUR and sell the equivalent amount in USD: Symbol(55)=EUR/USD, Side(54)=Buy(1), Currency(15)=EUR, Quantity=500k

-> sell 500k USD and buy the equivalent amount in EUR: Symbol(55)=EUR/USD, Side(54)=Sell(2), Currency(15)=USD, Quantity=500k

For a one-way quote, I will provide a Side that corresponds to the action I want to do on the specified currency for the far leg:

As a client, I want to send a quote request to buy 1M EUR Far and sell the equivalent in USD Far, sell 2M EUR Near and buy the equivalent in USD Near: Symbol(55)=EUR/USD, Side(54)=Buy(1), Currency(15)=EUR, Quantity=1M, Quantity2=2M

-> I will receive offer entries with a price for a OrderQty defined in EUR.

-> As a client, I want to buy 500k EUR Far and sell the equivalent USD Far, sell 700k EUR Near and buy the equivalent USD Near: Symbol(55)=EUR/USD, Side(54)=Buy(1), Currency(15)=EUR, Quantity=500k, Quantity2=700K

-> As a client, I want to sell 500k USD Far and buy the equivalent EUR Far, buy 500k USD Near and sell the equivalent EUR Near: Symbol(55)=EUR/USD, Side(54)=Sell(2), Currency(15)=USD, Quantity=500k, Quantity2=700K

As a client, I want to send a quote request to buy 1M USD Far and sell the equivalent in EUR Far ,buy 2M USD Near and buy the equivalent in EUR Near: Symbol(55)=EUR/USD, Side(54)=Buy(1), Currency(15)=USD, Quantity=1M, Quantity2=2M

-> I will receive bid entries with a price for a OrderQty defined in USD.

-> As a client, I want to sell 500k EUR Far and buy the equivalent USD Far, buy 700k EUR Near and sell the equivalent USD Near: Symbol(55)=EUR/USD, Side(54)=Sell(2), Currency(15)=EUR, Quantity=500k, Quantity2=700K

-> As a client, I want to buy 500k USD Far and sell the equivalent EUR Far, sell 700k USD Near and buy the equivalent EUR: Symbol(55)=EUR/USD, Side(54)=Buy(1), Currency(15)=USD, Quantity=500k, Quantity2=700K

As a client, I want to send a quote request to sell 1M EUR Far and buy the equivalent in USD Far, buy 1M EUR Near and sell the equivalent in USD Near: Symbol(55)=EUR/USD, Side(54)=Sell(2), Currency(15)=EUR, Quantity=1M, Quantity2=2M

-> I will receive bid entries with a price for a OrderQty defined in EUR.

-> As a client, I want to sell 500k EUR Far and buy the equivalent USD Far, buy 700k EUR Near and sell the equivalent USD Near: Symbol(55)=EUR/USD, Side(54)=Sell(2), Currency(15)=EUR, Quantity=500k, Quantity2=700K

-> As a client, I want to buy 500k USD Far and sell the equivalent EUR Far, sell 700k USD Near and buy the equivalent EUR Near: Symbol(55)=EUR/USD, Side(54)=Buy(1), Currency(15)=USD, Quantity=500k, Quantity2=700K

As a client, I want to send a quote request to sell 1M USD Far and buy the equivalent in EUR Far, buy 1M USD Near and sell the equivalent in EUR Near: Symbol(55)=EUR/USD, Side(54)=Sell(2), Currency(15)=USD, Quantity=1M, Quantity2=2M

-> I will receive offer entries with a price for a OrderQty defined in USD.

-> As a client, I want to buy 500k EUR Far and sell the equivalent USD Far, sell 700k EUR Near and buy the equivalent USD Near: Symbol(55)=EUR/USD, Side(54)=Buy(1), Currency(15)=EUR, Quantity=500k, Quantity2=700K

-> As a client, I want to sell 500k USD Far and buy the equivalent EUR Far, buy 700k USD Near and sell the equivalent EUR Near: Symbol(55)=EUR/USD, Side(54)=Sell(2), Currency(15)=USD, Quantity=500k, Quantity2=700K

5) Tenors (for RFS Forwards and Swaps)

The next table lists the supported tenors:

Tenor ID	Tenor Description
ONI	Overnight
TOD	Today
TOM	Tomorrow
TNX	Tomorrow Next
SP	Spot
SNX	Spot next
W1	1 week
W2	2 weeks
W3	3 weeks
M1	1 month
M2	2 months
M3	3 months
M4	4 months
M5	5 months
M6	6 months
M7	7 months
M8	8 months
M9	9 months
M10	10 months
M11	11 months
Y1	1 year
M13	13 months
M14	14 months
M15	15 months
M16	16 months
M17	17 months
M18	18 months
M19	19 months
M20	20 months
M21	21 months
M22	22 months
M23	23 months
Y2	2 years
Y3	3 years
Y4	4 years
Y5	5 years
Y6	6 years
Y7	7 years
Y8	8 years
Y9	9 years
Y10	10 years
Y15	15 years

Y20	20 years
Y25	25 years
Y30	30 years
IM1	IMM1
IM2	IMM2
IM3	IMM3
IM4	IMM4

In RFS, *Broken Dates* are also supported and the value must comply with the following pattern:

YYYYMMDD

with:

- **YYYY** representing the year
- **MM** representing the month
- **DD** representing the day

For instance, *20141225* stands for *December 25th, 2014*.

6) Currency pairs

6.1. Supported currency pairs

There are two types of currency pairs supported by FSS:

- FX – Foreign eXchange
- PM – Precious Metals

FX currency pairs are listed in section 6.2 and their precisions in section 6.4.

PM currency pairs and their precisions are listed in section 6.5.

Moreover, for Forwards in ESP mode, a predefined set of symbol combining a currency pair and a tenor is also available via SecurityListRequest as described in section 6.3.

6.2. FX Currency Pairs

Crossess

Primary						
EUR		USDEUR	JPYEUR	GBPEUR	CHFEUR	AUDEUR
USD	EURUSD		JPYUSD	GBPUSD	CHFUSD	AUDUSD
JPY	EURJPY	USDJPY		GBPJPY	CHFJPY	AUDJPY
GBP	EURGBP	USDGBP	JPYGBP		CHFGBP	AUDGBP
CHF	EURCHF	USDCHF	JPYCHF	GBPCHF		AUDCHF
AUD	EURAUD	USDAUD	JPYAUD	GBPAUD	CHFAUD	

Secondary							
AED	AUDCAD	CHFDKK	EURCHF	GBPPLN	NOKCZK	SEKDKK	USDHKD
CAD	AUDCHF	CHFEUR	EURCNH	GBPRON	NOKDKK	SEKEUR	USDHUF
CHF	AUDCZK	CHFGBP	EURCZK	GBPSAR	NOKEUR	SEKGBP	USDILS
CNH	AUDDKK	CHFHKD	EURDKK	GBPSEK	NOKGBP	SEKHKD	USDISK
CZK	AUDEUR	CHFHUF	EURGBP	GBPSGD	NOKHKD	SEKHUF	USDJPY
DKK	AUDGBP	CHFJPY	EURHKD	GBPTHB	NOKHUF	SEKJPY	USDKWD
HKD	AUDHKD	CHFMXN	EURHUF	GBPTRY	NOKJPY	SEKNOK	USDMXN
HUF	AUDHUF	CHFNOK	EURILS	GBPUSD	NOKSEK	SEKUSD	USDNGN
ILS	AUDILS	CHFNZD	EURJPY	GBPZAR	NOKUSD	SGDAUD	USDNOK
KWD	AUDJPY	CHFPLN	EURKWD	HKDCNH	NOKZAR	SGDCHF	USDNZD
MXN	AUDMXN	CHFRON	EURMXN	HKDJPY	NZDAUD	SGDCNH	USDOMR
NGN	AUDNOK	CHFSEK	EURNOK	HKDSEK	NZDCAD	SGDEUR	USDPLN
NOK	AUDNZD	CHFSGD	EURNZD	HKDTHB	NZDCHF	SGDGBP	USDRON
NZD	AUDPLN	CHFTHB	EUROMR	HKDUSD	NZDCZK	SGDHUF	USDRUB
OMR	AUDSEK	CHFTRY	EURPLN	ILSJPY	NZDDKK	SGDJPY	USDSAR
PLN	AUDSGD	CHFUSD	EURRON	JPYAUD	NZDEUR	SGDNOK	USDSEK
RON	AUDTHB	CHFZAR	EURRUB	JPYCAD	NZDGBP	SGDPLN	USDSGD
RUB	AUDTRY	CNHHKD	EURSAR	JPYCHF	NZDHKD	SGDSEK	USDTHB
SAR	AUDUSD	CNHJPY	EURSEK	JPYCNH	NZDHUF	SGDTHB	USDTRY
SEK	AUDZAR	CNHSGD	EURSGD	JPYDKK	NZDJPY	SGDUSD	USDZAR
SGD	CADAUD	CNHUSD	EURTHB	JPYEUR	NZDNOK	SGDZAR	ZARAUD
THB	CADCHF	CZKHUF	EURTRY	JPYGBP	NZDPLN	THBJPY	ZARCHF
TRY	CADCZK	CZKJPY	EURUSD	JPYHKD	NZDSEK	TRYAUD	ZAREUR
ZAR	CADDKK	DKKAUD	EURZAR	JPYHUF	NZDSGD	TRYCHF	ZARMXN
	CADEUR	DKKCHF	GBPAED	JPYMXN	NZDTHB	TRYDKK	ZARGBP
	CADGBP	DKKEUR	GBPAUD	JPYNOK	NZDTRY	TRYEUR	ZARJPY
	CADHKD	DKKGBP	GBPCAD	JPYNZD	NZDUSD	TRYGBP	ZARUSD
	CADHUF	DKKHKD	GBPCHF	JPYSEK	NZDZAR	TRYJPY	
	CADJPY	DKKHUF	GBPCZK	JPYSGD	OMRJPY	TRYMXN	
	CADMXN	DKKJPY	GBPDKK	JPYTRY	PLNCZK	TRYNZD	
	CADNOK	DKKNOK	GBPEUR	JPYUSD	PLNDKK	TRYUSD	
	CADPLN	DKKPLN	GBPHKD	JPYZAR	PLNHUF	USDAED	
	CADSEK	DKKSEK	GBPHUF	MXNAUD	PLNJPY	USDAUD	
	CADSGD	DKKSGD	GBPILS	MXNCHF	PLNNOK	USDCAD	
	CADTHB	DKKTHB	GBPJPY	MXNEUR	PLNSEK	USDCHF	
	CADUSD	DKKUSD	GBPKWD	MXNGBP	PLNZAR	USDCNH	
	CADZAR	DKKZAR	GBPMXN	MXNJPY	RONNOK	USDCZK	
	CHFAUD	EURAED	GBPNOK	MXNUSD	SEKAUD	USDDKK	
	CHFCAD	EURAUD	GBPNZD	NOKAUD	SEKCHF	USDEUR	
	CHFCZK	EURCAD	GBPOMR	NOKCHF	SEKCZK	USDGBP	

6.3. FX Currency Pairs with Tenors for ESP Forwards

FXSpotStream has no technical limitations on the CCY pairs and tenors that are available to stream in respect to the pairs listed previously and the exception of the *Broken Date*. To confirm what is currently available, please request the full security list (see section 8). If there is a CCY pair or tenor that you would like to trade that is not currently on the security list or in the above tables, please contact FXSpotStream support (support@fxspotstream.com).

The symbol of the corresponding ESP Forward securities complies with the following naming convention:

BASE/TERM*.FWD.*TENOR

The next table lists some examples of ESP Forward symbols and the corresponding interpretation (the full list is available through security list request as stated above):

Symbol	Currency Pair	Tenor
EUR/USD.FWD.M1	EURUSD	1 month
EUR/USD.FWD.M3	EURUSD	3 months
USD/JPY.FWD.M1	USDJPY	1 month
USD/JPY.FWD.M3	USDJPY	3 months
GBP/USD.FWD.M1	GBPUSD	1 month
GBP/USD.FWD.M3	GBPUSD	3 months

6.4. FX Precisions

For the Full Amount option, the calculated VWAP price will be rounded to a precision that is dependent to the currency pairs as per the table in section 6.4.1 for:

- CommerzBank
- Citi
- HSBC
- JPMorgan Chase
- Morgan Stanley
- UBS
- BNP Paribas

The following liquidity providers will use specific precisions:

- Bank of America's precisions are listed in section 6.4.2
- Goldman Sachs' precisions are listed in section 6.4.3
- Credit Suisse's precisions are listed in section 6.4.4
- Standard Chartered precisions are listed in section 6.4.5

For the pass-through option, FSS will send the price sent by the bank, without doing any rounding.

6.4.1. General FX Precisions (COBA, CITI, HSBC, JPMC, MS, UBS, BNP)

Ccy Pair	Decimal Precision	Ccy Pair	Decimal Precision	Ccy Pair	Decimal Precision
AUDCAD	5	EURMXN	4	NZDNOK	4
AUDCHF	5	EURNOK	5	NZDPLN	4
AUDCZK	3	EURNZD	5	NZDSEK	5
AUDDKK	5	EUROMR	5	NZDSGD	5
AUDEUR	5	EURPLN	5	NZDTHB	3
AUDGBP	5	EURRON	5	NZDTRY	5
AUDHKD	5	EURRUB	5	NZDUSD	5
AUDHUF	2	EURSAR	5	NZDZAR	5
AUDILS	4	EURSEK	5	OMRJPY	4
AUDJPY	3	EURSGD	5	PLNCZK	4
AUDMXN	4	EURTHB	3	PLNDKK	5
AUDNOK	5	EURTRY	5	PLNHUF	2
AUDNZD	5	EURUSD	5	PLNJPY	3
AUDPLN	4	EURZAR	4	PLNNOK	5
AUDSEK	5	GBPAED	5	PLNSEK	5
AUDSGD	5	GBPAUD	5	PLNZAR	5
AUDTHB	3	GBPCAD	5	RONNOK	2
AUDTRY	2	GBPCHF	5	SEKAUD	5
AUDUSD	5	GBPCZK	4	SEKCHF	4
AUDZAR	4	GBPDKK	5	SEKCZK	3
CADAUD	4	GBPEUR	5	SEKDKK	5
CADCHF	5	GBPHKD	5	SEKEUR	5
CADCZK	3	GBPHUF	3	SEKGBP	5
CADDKK	5	GBPILS	4	SEKHKD	5
CADEUR	5	GBPJPY	3	SEKHUF	3
CADGBP	5	GBPKWD	5	SEKJPY	5
CADHKD	5	GBPMXN	4	SEKNOK	4
CADHUF	3	GBPNOK	5	SEKUSD	5
CADJPY	3	GBPNZD	5	SGDAUD	5
CADMXN	5	GBPOMR	5	SGDCHF	5
CADNOK	5	GBPPLN	5	SGDCNH	5
CADPLN	4	GBPRON	5	SGDEUR	5
CADSEK	5	GBPSAR	5	SGDGBP	5
CADSGD	5	GBPSEK	5	SGDHUF	2
CADTHB	3	GBPSGD	4	SGDJPY	3
CADUSD	5	GBPTHB	3	SGDNOK	5
CADZAR	5	GBPTRY	5	SGDPLN	2
CHFAUD	4	GBPUSD	5	SGDSEK	5
CHFCAD	4	GBPZAR	4	SGDTHB	3
CHFCZK	2	HKDCNH	4	SGDUSD	4
CHFDKK	5	HKDJPY	4	SGDZAR	5
CHFEUR	5	HKDSEK	5	THBJPY	5
CHFGBP	5	HKDTHB	3	TRYAUD	4
CHFHKD	5	HKDUSD	4	TRYCHF	4
CHFHUF	3	ILSJPY	3	TRYDKK	5
CHFJPY	3	JPYAUD	4	TRYEUR	5
CHFMXN	4	JPYCAD	4	TRYGBP	5
CHFNOK	5	JPYCHF	4	TRYJPY	3
CHFNZD	5	JPYCNH	4	TRYMXN	5
CHFPLN	5	JPYDKK	5	TRYNZD	5
CHFRON	5	JPYEUR	5	TRYUSD	5

CHFSEK	5		JPYGBP	5		USDAED	5
CHFSGD	5		JPYHKD	4		USDAUD	5
CHFTHB	3		JPYHUF	4		USDCAD	5
CHFTRY	4		JPYMXN	4		USDCHF	5
CHFUSD	5		JPYNOK	4		USDCNH	4
CHFZAR	4		JPYNZD	4		USDCZK	4
CNHHKD	4		JPYSEK	4		USDDKK	5
CNHJPY	4		JPYSGD	4		USDEUR	5
CNHSGD	5		JPYTRY	4		USDGBP	5
CNHUSD	4		JPYUSD	4		USDHKD	5
CZKHUF	4		JPYZAR	4		USDHUF	3
CZKJPY	3		MXNAUD	2		USDILS	5
DKKAUD	4		MXNCHF	4		USDISK	3
DKKCHF	4		MXNEUR	4		USDJPY	3
DKKEUR	4		MXNGBP	4		USDKWD	5
DKKGBP	4		MXNJPY	4		USDMXN	5
DKKHKD	5		MXNUSD	4		USDNGN	2
DKKHUF	3		NOKAUD	4		USDNOK	5
DKKJPY	5		NOKCHF	5		USDNZD	5
DKKNOK	5		NOKCZK	2		USDOMR	6
DKKPLN	4		NOKDKK	5		USDPLN	5
DKKSEK	5		NOKEUR	5		USDRON	5
DKKSGD	4		NOKGBP	5		USDRUB	5
DKKTHB	3		NOKHKD	5		USDSAR	5
DKKUSD	2		NOKHUF	3		USDSEK	5
DKKZAR	5		NOKJPY	3		USDSGD	5
EURAED	5		NOKSEK	4		USDTHB	3
EURAUD	5		NOKUSD	5		USDTRY	5
EURCAD	5		NOKZAR	5		USDZAR	4
EURCHF	5		NZDAUD	5		ZARAUD	2
EURCNH	4		NZDCAD	5		ZARCHF	4
EURCZK	4		NZDCHF	5		ZAREUR	4
EURDKK	5		NZDCZK	3		ZARGBP	4
EURGBP	5		NZDDKK	5		ZARMXN	5
EURHKD	5		NZDEUR	5		ZARJPY	3
EURHUF	3		NZDGBP	5		ZARUSD	4
EURILS	4		NZDHKD	5			
EURJPY	3		NZDHUF	2			
EURKWD	5		NZDJPY	3			

6.4.2. FX Precisions for BAML

Ccy Pair	Decimal Precison	Ccy Pair	Decimal Precison	Ccy Pair	Decimal Precison
AUDCAD	5	EURCZK	4	HKDJPY	5
AUDCHF	5	EURDKK	5	JPYHUF	5
AUDHKD	5	EURGBP	5	MXNJPY	3
AUDJPY	3	EURHKD	5	NOKJPY	5
AUDMXN	5	EURHUF	3	NOKSEK	5
AUDNZD	5	EURJPY	3	NZDCAD	5
AUDSGD	5	EURMXN	5	NZDCHF	5
AUDUSD	5	EURNOK	5	NZDJPY	3
AUDZAR	5	EURNZD	5	NZDUSD	5
CADCHF	5	EURPLN	5	PLNCZK	5
CADJPY	3	EURRON	5	PLNHUF	5
CADMXN	5	EURSEK	5	PLNJPY	3
CADZAR	5	EURSGD	5	SEKJPY	5
CHFCZK	5	EURTRY	5	SGDJPY	3
CHFHKD	5	EURUSD	5	TRYJPY	3
CHFHUF	3	EURZAR	5	USDCAD	5
CHFJPY	3	GBPAUD	5	USDCHF	5
CHFMXN	5	GBPCAD	5	USDCZK	4
CHFNOK	5	GBPCHF	5	USDDKK	5
CHFPLN	5	GBPCZK	5	USDHKD	5
CHFSEK	5	GBPHKD	5	USDHUF	3
CHFTRY	5	GBPHUF	5	USDJPY	3
CHFZAR	5	GBPJPY	3	USDMXN	5
CZKHUF	5	GBPNOK	5	USDNOK	5
CZKJPY	3	GBPNZD	5	USDPLN	5
DKKJPY	5	GBPPLN	5	USDSEK	5
DKKNOK	5	GBPSEK	5	USDSGD	5
DKKSEK	5	GBPTRY	5	USDTRY	5
EURAUD	5	GBPUSD	5	USDZAR	5
EURCAD	5	GBPZAR	5	ZARJPY	3
EURCHF	5				

6.4.3. FX Precisions for GS

Ccy Pair	Decimal Precison	Ccy Pair	Decimal Precison	Ccy Pair	Decimal Precison
AUDCAD	5	EURPLN	4	SGDJPY	3
AUDCHF	5	EURRON	4	TRYJPY	3
AUDDKK	5	EURSEK	5	USDCAD	5
AUDJPY	3	EURTRY	4	USDCHF	5
AUDNZD	5	EURUSD	5	USDCZK	3
AUDUSD	5	GBPAUD	5	USDDKK	5
CADCHF	5	GBPCAD	5	USDHKD	5
CADJPY	3	GBPCHF	5	USDHUF	2
CHFJPY	3	GBPJPY	3	USDILS	4
EURAUD	5	GBPNZD	5	USDJPY	3
EURCAD	5	GBPSEK	5	USDMXN	4
EURCHF	5	GBPUSD	5	USDNOK	5
EURCZK	3	HKDJPY	4	USDPLN	4
EURDKK	5	NOKJPY	4	USDRUB	4
EURGBP	5	NOKSEK	5	USDSEK	5
EURHKD	5	NZDCAD	5	USDSGD	5
EURHUF	2	NZDCHF	5	USDTHB	3
EURJPY	3	NZDJPY	3	USDTRY	4
EURNOK	5	NZDUSD	5	USDZAR	5
EURNZD	5	SEKJPY	4	ZARJPY	3

6.4.4. FX Precisions for CS

Ccy Pair	Decimal Precision	Ccy Pair	Decimal Precision	Ccy Pair	Decimal Precision
AUDCAD	5	EURHUF	2	NZDHKD	4
AUDCHF	5	EURJPY	3	NZDHUF	2
AUDCZK	3	EURMXN	4	NZDJPY	3
AUDDKK	4	EURNOK	5	NZDNOK	4
AUDEUR	4	EURNZD	5	NZDPLN	4
AUDGBP	4	EURPLN	4	NZDSEK	4
AUDHKD	4	EURSEK	5	NZDSGD	4
AUDHUF	2	EURSGD	5	NZDTRY	4
AUDILS	4	EURTHB	3	NZDUSD	5
AUDJPY	3	EURTRY	5	NZDZAR	4
AUDMXN	4	EURUSD	5	PLNCZK	3
AUDNOK	4	EURZAR	4	PLNDKK	5
AUDNZD	5	GBPAUD	5	PLNHUF	3
AUDPLN	4	GBPCAD	5	PLNJPY	4
AUDSEK	4	GBPCHF	5	PLNNOK	4
AUDTRY	4	GBPCZK	3	PLNSEK	4
AUDUSD	5	GBPDKK	4	PLNZAR	4
AUDZAR	4	GBPEUR	5	SEKAUD	6
CADAUD	5	GBPHKD	4	SEKCHF	5
CADCHF	5	GBPHUF	2	SEKCZK	3
CADCZK	3	GBPJPY	3	SEKDKK	5
CADDKK	4	GBPMXN	4	SEKEUR	6
CADEUR	5	GBPNOK	4	SEKGBP	6
CADGBP	5	GBPNZD	4	SEKHKD	4
CADHKD	4	GBPPLN	4	SEKHUF	2
CADHUF	2	GBPSEK	4	SEKJPY	4
CADJPY	3	GBPSGD	4	SEKNOK	5
CADMXN	4	GBPUSD	5	SEKUSD	6
CADNOK	4	GBPZAR	4	SGDCHF	5
CADPLN	4	HKDJPY	4	SGDEUR	5
CADSEK	4	HKDSEK	6	SGDGBP	5
CADSGD	5	HKDUSD	6	SGDHUF	2
CADUSD	5	ILSJPY	3	SGDJPY	3
CADZAR	4	JPYAUD	7	SGDNOK	4
CHFAUD	5	JPYCAD	7	SGDPLN	4
CHFCAD	5	JPYCHF	7	SGDSEK	4
CHFCZK	3	JPYDKK	6	SGDUSD	5
CHFDKK	4	JPYEUR	7	SGDZAR	4
CHFEUR	5	JPYGBP	8	TRYAUD	4
CHFGBP	5	JPYHKD	4	TRYDKK	4
CHFHKD	5	JPYHUF	4	TRYEUR	4
CHFHUF	2	JPYMXN	4	TRYJPY	5
CHFJPY	3	JPYNOK	5	TRYNZD	4
CHFMXN	4	JPYNZD	7	TRYUSD	4
CHFNOK	4	JPYSEK	5	USDAED	4
CHFNZD	5	JPYSGD	7	USDAUD	5
CHFPLN	4	JPYTRY	4	USDCAD	5
CHFSEK	4	JPYUSD	7	USDCHF	5
CHFSGD	5	JPYZAR	4	USDCZK	3
CHFUSD	5	MXNAUD	6	USDDKK	5
CHFZAR	4	MXNCHF	6	USDEUR	5

CZKHUF	2		MXNEUR	6		USDGBP	5
CZKJPY	4		MXNGBP	6		USDHKD	5
DKKAUD	5		MXNJPY	5		USDHUF	2
DKKCHF	5		MXNUSD	6		USDILS	5
DKKEUR	6		NOKAUD	6		USDJPY	3
DKKGBP	6		NOKCHF	5		USDKWD	5
DKKHKD	4		NOKCZK	3		USDMXN	4
DKKHUF	2		NOKDKK	5		USDNOK	5
DKKJPY	3		NOKEUR	6		USDNZD	5
DKKNOK	5		NOKGBP	6		USDPLN	4
DKKPLN	6		NOKHKD	4		USDSAR	4
DKKSEK	5		NOKHUF	2		USDSEK	5
DKKSGD	6		NOKJPY	4		USDSGD	5
DKKUSD	6		NOKSEK	5		USDTRY	5
DKKZAR	4		NOKUSD	6		USDZAR	4
EURAED	4		NOKZAR	4		ZARAUD	5
EURAUD	5		NZDAUD	5		ZARCHF	5
EURCAD	5		NZDCAD	5		ZAREUR	5
EURCHF	5		NZDCHF	5		ZARGBP	5
EURCZK	3		NZDCZK	3		ZARJPY	5
EURDKK	5		NZDDKK	4		ZARUSD	5
EURGBP	5		NZDEUR	5			
EURHKD	5		NZDGBP	5			

6.4.5. FX Precisions for SCB

Ccy Pair	Decimal Precision		Ccy Pair	Decimal Precision		Ccy Pair	Decimal Precision
AUDCAD	5		EURMXN	5		NZDNOK	5
AUDCHF	5		EURNOK	5		NZDPLN	5
AUDCZK	4		EURNZD	5		NZDSEK	5
AUDDKK	5		EUROMR	6		NZDSGD	5
AUDEUR	5		EURPLN	5		NZDTHB	4
AUDGBP	5		EURRON	4		NZDTRY	5
AUDHKD	5		EURRUB	4		NZDUSD	5
AUDHUF	3		EURSAR	5		NZDZAR	5
AUDILS	5		EURSEK	5		OMRJPY	5
AUDJPY	3		EURSGD	5		PLNCZK	5
AUDMXN	5		EURTHB	3		PLNDKK	5
AUDNOK	5		EURTRY	5		PLNHUF	3
AUDNZD	5		EURUSD	5		PLNJPY	4
AUDPLN	5		EURZAR	5		PLNNOK	5
AUDSEK	5		GBPAED	5		PLNSEK	5
AUDSGD	5		GBPAUD	5		PLNZAR	6
AUDTHB	3		GBPCAD	5		RONNOK	4
AUDTRY	5		GBPCHF	5		SEKAUD	5
AUDUSD	5		GBPCZK	4		SEKCHF	5
AUDZAR	5		GBPDKK	5		SEKCZK	4
CADAUD	5		GBPEUR	5		SEKDKK	5
CADCHF	5		GBPHKD	5		SEKEUR	5
CADCZK	4		GBPHUF	3		SEKGBP	5
CADDKK	5		GBPILS	5		SEKHKD	6
CADEUR	5		GBPJPY	3		SEKHUF	3
CADGBP	5		GBPKWD	6		SEKJPY	5

CADHKD	5		GBPMXN	5		SEKNOK	5
CADHUF	3		GBPNOK	5		SEKUSD	5
CADJPY	3		GBPNZD	5		SGDAUD	6
CADMXN	5		GBPOMR	6		SGDCHF	4
CADNOK	5		GBPPLN	5		SGDCNH	5
CADPLN	5		GBPRON	4		SGDEUR	6
CADSEK	5		GBPSAR	5		SGDGBP	6
CADSGD	5		GBPSEK	5		SGDHUF	3
CADTHB	4		GBPSGD	5		SGDJPY	3
CADUSD	5		GBPTHB	3		SGDNOK	4
CADZAR	5		GBPTRY	5		SGDPLN	5
CHFAUD	5		GBPUSD	5		SGDSEK	5
CHFCAD	5		GBPZAR	5		SGDTHB	3
CHFCZK	4		HKDCNH	5		SGDUSD	5
CHFDKK	5		HKDJPY	3		SGDZAR	5
CHFEUR	5		HKDSEK	4		THBJPY	4
CHFGBP	5		HKDTHB	4		TRYAUD	5
CHFHKD	5		HKDUSD	4		TRYCHF	5
CHFHUF	3		ILSJPY	4		TRYDKK	5
CHFJPY	3		JPYAUD	6		TRYEUR	5
CHFMXN	5		JPYCAD	6		TRYGBP	5
CHFNOK	5		JPYCHF	7		TRYJPY	3
CHFNZD	5		JPYCNH	7		TRYMXN	5
CHFPLN	5		JPYDKK	7		TRYNZD	5
CHFRON	4		JPYEUR	6		TRYUSD	5
CHFSEK	5		JPYGBP	6		USDAED	6
CHFSGD	5		JPYHKD	6		USDAUD	5
CHFTHB	4		JPYHUF	5		USDCAD	5
CHFTRY	5		JPYMXN	7		USDCHF	5
CHFUSD	5		JPYNOK	7		USDCNH	5
CHFZAR	5		JPYNZD	6		USDCZK	4
CNHHKD	5		JPYSEK	7		USDDKK	5
CNHJPY	3		JPYSGD	7		USDEUR	5
CNHSGD	7		JPYTRY	7		USDGBP	5
CNHUSD	7		JPYUSD	5		USDHKD	5
CZKHUF	4		JPYZAR	5		USDHUF	3
CZKJPY	5		MXNAUD	6		USDILS	5
DKKAUD	5		MXNCHF	6		USDJPY	3
DKKCHF	5		MXNEUR	5		USDKWD	5
DKKEUR	5		MXNGBP	6		USDMXN	5
DKKGBP	5		MXNJPY	5		USDNGN	2
DKKHKD	6		MXNUSD	5		USDNOK	5
DKKHUF	3		NOKAUD	5		USDNZD	5
DKKJPY	5		NOKCHF	5		USDOMR	6
DKKNOK	5		NOKCZK	4		USDPLN	5
DKKPLN	5		NOKDKK	5		USDRON	4
DKKSEK	5		NOKEUR	5		USDRUB	4
DKKSGD	5		NOKGBP	5		USDSAR	5
DKKTHB	4		NOKHKD	6		USDSEK	5
DKKUSD	5		NOKHUF	3		USDSGD	5
DKKZAR	6		NOKJPY	5		USDTHB	3
EURAED	5		NOKSEK	5		USDTRY	5
EURAUD	5		NOKUSD	5		USDZAR	5

EURCAD	5	NOKZAR	6	ZARAUD	6
EURCHF	5	NZDAUD	5	ZARCHF	6
EURCNH	5	NZDCAD	5	ZAREUR	5
EURCZK	4	NZDCHF	5	ZARGBP	6
EURDKK	5	NZDCZK	4	ZARJPY	5
EURGBP	5	NZDDKK	5	ZARMXN	5
EURHKD	5	NZDEUR	5	ZARUSD	5
EURHUF	3	NZDGBP	5		
EURILS	5	NZDHKD	5		
EURJPY	3	NZDHUF	3		
EURKWD	6	NZDJPY	3		

6.5. Precious Metals Currency Pairs and Precisions

Ccy Pair	Decimal Precison	Ccy Pair	Decimal Precison	Ccy Pair	Decimal Precison
XAGAUD	6	XAUHKD	4	XPDTHB	4
XAGCAD	6	XAUJPY	4	XPDTRY	4
XAGCHF	6	XAUNOK	4	XPDUSD	4
XAGDKK	6	XAUNZD	4	XPDZAR	4
XAGEUR	6	XAUSEK	4	XPTAUD	4
XAGGBP	6	XAUSGD	4	XPTCAD	4
XAGHKD	6	XAUTHB	4	XPTCHF	4
XAGJPY	6	XAUTRY	4	XPTDKK	4
XAGNOK	6	XAUUSD	4	XPTEUR	4
XAGNZD	6	XAUZAR	4	XPTGBP	4
XAGSEK	6	XPDAUD	4	XPTHKD	4
XAGSGD	6	XPDCAD	4	XPTJPY	4
XAGTHB	6	XPDCHF	4	XPTNOK	4
XAGTRY	6	XPDDKK	4	XPTNZD	4
XAGUSD	6	XPDEUR	4	XPTSEK	4
XAGZAR	6	XPDGBP	4	XPTSGD	4
XAUAUD	4	XPDHKD	4	XPTTHB	4
XAUCAD	4	XPDJPY	4	XPTTRY	4
XAUCHF	4	XPDNOK	4	XPTUSD	4
XAUDKK	4	XPDNZD	4	XPTZAR	4
XAUEUR	4	XPDSEK	4		
XAUGBP	4	XPDSGD	4		

7) Session

To establish each session, a “Logon” message is sent. You may request to reset the sequence numbers (tag “ResetSeqNumFlag” set to ”Y”). A successful login will result with a “Logon” message sent back.

The Logon message for the streaming session must contain the ResetOnLogon flag set to Y as there is no store. MarketDataSnapshot will not be retransmitted if a gap is detected. Instead a SequenceReset will be sent back on reception of a ResendRequest.

7.1. Logon Message Definition (Type A)

Tag name	Tag number	Required	Description
EncryptMethod	98	Y	Has to be 0 (None)
HeartBtInt	108	Y	Heartbeat interval in seconds
ResetSeqNumFlag	141	N	Set to “Y” to reset sequence numbers. This should be the default for market data sessions. Set to “N” (or not sent) to not reset sequence numbers. This should be the default for trading sessions in order to allow guaranteed delivery.
Username	553	N	UserID or username.
Password	554	N	Password or passphrase.

An example of a FIX logon message on a streaming session:

8=FIX.4.4|9=93|35=A|34=1|49=STR.NY.SIM.CLIENT|52=20150212-04:31:38.971|56=FSS|141=Y|98=0|108=35|10=087|

An example of a FIX logon message on a trading session:

8=FIX.4.4|9=90|35=A|34=1850|49=TRD.NY.SIM.CLIENT|52=20150212-04:31:38.972|56=FSS|141=N|98=0|108=35|10=169|

7.2. Logout Message Definition (Type 5)

Tag name	Tag number	Required	Description
Text	58	N	Will contain a description for failed logins or unsolicited logouts.

An example of a FIX logout message:

8=FIX.4.4|9=57|35=5|49=CLIENT|56=FSS|34=51|52=20101026-18:46:40|10=082|

8) Security

A “SecurityListRequest” message is be used to request a list of currency pairs that can be traded by the session.

A “SecurityList” message in response will contain a list of currency pairs supported by FSS.

For network optimizations, the security list might be split into multiple security lists. Each last security list will have a flag “LastFragment” specifying whether it is the last one. Only the last one will have the flag set to true.

8.1. SecurityListRequest Message Definition (Type x)

Tag name	Tag number	Required	Description
SecurityReqID	320	Y	Unique identifier for this request.
SecurityListRequestType	559	Y	Has to be set to AllSecurities (4)

An example of a SecurityListRequest message:

8=FIX.4.4|9=109|35=x|34=2|49=CLIENT|52=20130904-12:22:50.159|56=FSS|320=SLRkruDaT/hSdXiRu8UFy2zmg|559=4|10=045|

8.2. SecurityList (Type y)

Tag name	Tag number	Required	Description
SecurityReqID	320	Y	The original request id sent by the client.
SecurityRequestResult	560	Y	The result of this request. Will either be: 0 – ValidReq 1 – InvalidReq
Text	58	N	Will contain the full error description when SecurityRequestResult = InvalidReq
LastFragment	893	N	Indicates whether this message is the last in a sequence of messages.
NoRelatedSym	146	Y	Number of currency pairs returned
➢ Symbol	55	Y	The currency pair
➢ SecurityType	167	Y	Indicates type of security. Will either be: “FX” – Foreign eXchange “PM” – Precious Metals
➢ SettlDate	64	N	For Forwards only, indicates the corresponding tenor

An example of a SecurityList message:

8=FIX.4.4|9=0883|35=y|52=20130904-12:22:50.160|49=FSS|56=CLIENT|34=4|320=SLRkruDaT/hSdXiRu8UFy2zmg|560=0|893=Y|146=42|55=SEK/AUD|167=FX|55=SEK/CHF|167=FX|55=SEK/EUR|167=FX|55=SEK/GBP|167=FX|55=SEK/JPY|167=FX|55=SEK/USD|167=FX|55=SGD/AUD|167=FX|55=SGD/CHF|167=FX|55=SGD/EUR|167=FX|55=SGD/GBP|167=FX|55=SGD/JPY|167=FX|55=SGD/USD|167=FX|55=TRY/AUD|167=FX|55=TRY/CHF|167=FX|55=TRY/EUR|167=FX|55=TRY/GBP|1

67=FX|55=TRY/JPY|167=FX|55=TRY/USD|167=FX|55=USD/AUD|167=FX|55=USD/CAD|167=FX|55=USD/CHF|167=FX|55=USD/DKK|167=FX|55=USD/EUR|167=FX|55=USD/GBP|167=FX|55=USD/JPY|167=FX|55=USD/MXN|167=FX|55=USD/NOK|167=FX|55=USD/NZD|167=FX|55=USD/SEK|167=FX|55=USD/SGD|167=FX|55=USD/TRY|167=FX|55=USD/ZAR|167=FX|55=XAG/USD|167=PM|55=XAU/USD|167=PM|55=XPD/USD|167=PM|55=XPT/USD|167=PM|55=ZAR/AUD|167=FX|55=ZAR/CHF|167=FX|55=ZAR/EUR|167=FX|55=ZAR/GBP|167=FX|55=ZAR/JPY|167=FX|55=ZAR/USD|167=FX|10=103|

9) Market Data

To be able to receive market data, clients need to subscribe to a currency pair. If the subscription is successful, clients will start receiving prices. If the subscription fails, a reject message will be sent back to the client. Clients need to unsubscribe to stop receiving market data.

In the event that the FIX server is unable to send any more prices, it will send a reject message back, and the initial subscription is invalidated. Clients will need to re-subscribe to receive prices again.

9.1. Subscribe to Market Data

This message is used by the client system to request Market Data Snapshots for a single instrument (currency pair). In response, FIX Server will send out MarketDataSnapshots if it is a valid request. Otherwise, FIX Server will send out a MarketDataRequestReject whose 'Text' tag might contain the reason of rejection.

Each subscription request needs to have a unique identifier, set via tag "MDReqID", which will be used to refer back to the client. It is also used to unsubscribe from market data.

Optional sizes can be specified using the group tag "RequestedSize" to allow streaming of market data for specific sizes. Multiple requests for different sizes of the same currency can be requested. The client needs to ensure that a unique MDReqID is specified for each of these requests.

Optional PartyIDs can be specified to reduce the streaming of market data to a subset of all the Liquidity Providers. If no PartyID is specified then it will request Market Data Snapshots from all the Liquidity Providers. It is possible to reuse the same "MDReqID" in order to update the list of Liquidity Providers in the subscription.

9.2. Unsubscribe from Market Data

To unsubscribe from prices, a "MarketDataRequest" message is sent with the unique identifier used in the initial subscription request and the tag "SubscriptionRequestType" set to "Unsubscribe" (2).

If the subscription removal is successful, the client will stop receiving market data immediately.

However, if the request was unsuccessful, a "MarketDataRequestReject" message will be sent back to the client. The tag 'Text' will contain the reason for rejection.

9.3. Request for Trading: Limit Orders Case

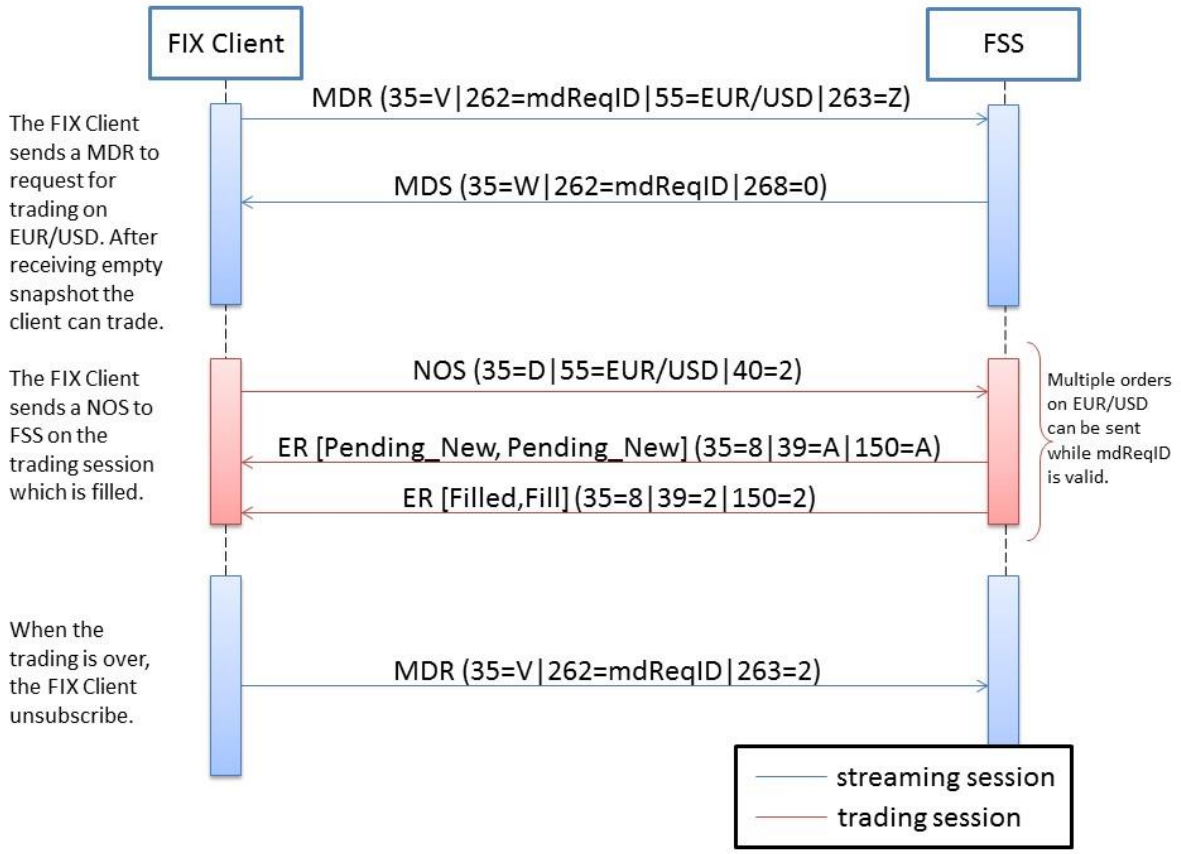
For limit orders, a MarketDataRequest with 40=2 should be sent to request for trading. It is possible to receive in return the prices that will be monitored by FSS for limit orders with a SubscriptionRequestType=1 or to subscribe without market feedback.

In order to request for trading without receiving market data, a MarketDataRequest message must be sent with tag SubscriptionRequestType set to NoMarketFeedback (Z). Trading could start after receiving an empty MarketDataSnapshot message in response to the MarketDataRequest.

This is useful for Aggressive Limit Orders in order to have liquidity available at the time of sending the orders (see section 10).

To unsubscribe, a "MarketDataRequest" message should also be sent with the unique identifier used in the initial request for trading and the tag "SubscriptionRequestType" set to "Unsubscribe" (2).

The following diagram depicts the workflow of request for trading without price feedback:



9.4. MarketDataRequest Message Definition (Type V)

Tag name	Tag number	Required	Description
MDReqID	262	Y	A unique identifier supplied by the client.
SubscriptionRequestType	263	Y	Use SnapshotAndUpdates (1) to subscribe to prices and Unsubscribe (2) to unsubscribe. Snapshot (0) requests are not supported. Use NoMarketFeedback (Z) to request for trading only. 1 = SnapshotAndUpdates 2 = Unsubscribe Z = NoMarketFeedback (Trading only)
NoPartyIDs	453	N	Repeating group below should contain unique combinations of PartyID, PartyIDSource, and PartyRole
➢ PartyID	448	Y	Used to identify source of PartyID. Valid values are listed in Liquidity Provider ID (section 8). Required if NoPartyIDs > 0.
➢ PartyIDSource	447	Y	Required if NoPartyIDs > 0. Valid value = D.
➢ PartyRole	452	Y	Required if NoPartyIDs > 0. Valid value: 35 = Liquidity Provider (codes are listed in section 2.3)
NoRelatedSym	146	Y	Specifies the number of repeating symbols Specified. Valid value: 1
➢ Symbol	55	Y	The CCY pair requested. Must be the same for all elements in the group.
MarketDepth	264	N	Depth of market for Book Snapshot. Required for “Option 2” (Pass-through). Valid values: 0 = Full Book Depth 1 = Top of Book N>1 = Report best N price tiers of data
MDUpdateType	265	Y	Specifies the type of Market Data update. Valid values: 0 = Full Refresh 1 = Incremental Refresh
NoRequestedSize	9000	N	Number of size.
➢ RequestedSize	9001	N	The size of the quote. Required for “Option 1” (Full amount).
OrdType	40	N	Price Stream type. Valid values: 2 = Limit

Here is an example of MarketDataRequest message for “Option 1” (Full Amount):

8=FIX.4.4|9=222|35=V|49=Client|56=FSS|34=46|262=MDReqID|263=1|265=0|146=1|55=EUR/USD|9000=1|9001=3000000|10=222|

Here is an example of MarketDataRequest message for “Option 2” (Passthrough):

8=FIX.4.4|9=222|35=V|49=Client|56=FSS|34=46|262=MDReqID|263=1|264=6|265=0|146=1|55=EUR/USD|10=112|

Here is an example of MarketDataRequest message with a reduced distribution containing JPMC and BAML only:

8=FIX.4.4|9=222|35=V|49=Client|56=FSS|34=46|262=MDReqID|453=2|448=JPMC|447=D|452=35|448=BAML|447=D|452=35|263=1|264=6|265=0|146=1|55=EUR/USD|10=112|

9.5. MarketDataRequestReject Message Definition (Type Y)

Tag name	Tag number	Required	Description
MDReqID	262	Y	The original unique identifier supplied by the client.
MDReqRejReason	281	N	Reject reason code.
Text	58	N	Full description for rejection.

9.6. Streaming Market Data

In the case of a subscription with MDUpdateType = 0 (Full Refresh), all market data prices are sent to clients using “MarketDataSnapshotFullRefresh” messages. Each message represents the complete liquidity levels on a currency pair, and should replace all previous messages sent on that currency pair.

In the case of a subscription with MDUpdateType = 1 (Incremental Refresh), market data prices are sent to clients using both “MarketDataSnapshotFullRefresh” and “MarketDataIncrementalRefresh” messages. The first update is always a “MarketDataSnapshotFullRefresh” message which represents the current snapshot of the market for the considered currency pair. Subsequent updates only convey the data that have changed (i.e. a delta from the previous snapshot) and are sent using “MarketDataIncrementalRefresh”. The tag MDEntryRefID (280) is used to identify the entries that have to be changed or deleted. In particular cases, a “MarketDataSnapshotFullRefresh” message can be sent after a “MarketDataIncrementalRefresh” message in order to indicate the client to clear its cache and replace its state of the market for that currency pair.

These prices will stream continuously until the client requests to unsubscribe using a Market Data Request with the SubscriptionRequestType set to Unsubscribe (2).

Each message may contain a few market data entries (MDEntriesNo repeating group) which represent different levels of liquidity available. Each entry contains a unique reference (tag MDEntryID) that should be used when submitting orders.

9.7. MarketDataSnapshotFullRefresh Message Definition (Type W)

Tag name	Tag number	Required	Description
MDReqID	262	Y	The original unique identifier supplied by the client.
Symbol	55	Y	The instrument, CCY Pair.
SettlDate	64	Y	Tenor of the instrument.
NoMDEntries	268	Y	Number of entries in the Market Data message. It is the number of liquidity bands. It can be set to 0 if there is no entry visible. The client must clear the depth in this case and ensure that no more order is sent.
➢ MDEntryType	269	Y	Side of this liquidity level 0 = Bid 1 = Offer H = Mid Rate
➢ MDEntryRefID	280	Cond	Refers to a previous MDEntryRefID. Used in the case of a subscription (type V) with Incremental Refresh Updates (MDUpdateType = 1). This id must be unique for a couple MDReqID/Side.
➢ MDEntryPositionNo	290	Cond	Display position of a bid or offer, numbered from most competitive to least competitive, per market side. Only present in the case of a subscription with MDUpdateType = 1 (Incremental Refresh)
➢ MDEntryID	278	Cond	Identifier for this market data entry. Used in previously quoted orders to reference the quote when submitting an order to hit this entry. Provided unless MDEntryType = H.
➢ MDEntryPx	270	Y	The price of this liquidity level.
➢ MDEntrySize	271	Cond	The size of this liquidity level. Provided unless MDEntryType = H.
➢ MDEntryDate	272	Optional[1] (provided by default)	The date of this liquidity level. Provided unless MDEntryType = H.
➢ MDEntryTime	273	Cond	The time of this liquidity level. Provided unless MDEntryType = H.
➢ Currency	15	Cond	The currency that MDEntryPx and MDEntrySize tags refer to. Provided unless MDEntryType = H.
➢ MinQty	110	N	Minimum quantity of an order to be executed.

[1] Can be activated/deactivated on demand.

➢ MDEntryOriginator	282	Y	The ID of the liquidity provider as defined in section 2.3
➢ SettlDate	64	Y	The value date.
➢ MDEntryOrigTime	9122	Optional[1] (not provided by default)	The original time from the LP of this liquidity level.
➢ MDEntrySpotRate	1026	N	For forwards, the spot rate when available.

9.7.1. PreTrade Mid-Market Rate

Some banks require that the client receives a pre-trade mid-market rate when they provide it. When a Liquidity Provider provides such information, it will be sent to the client.

9.8. MarketDataIncrementalRefresh Message Definition (Type X)

Tag name	Tag number	Required	Description
MDReqID	262	Y	The original unique identifier supplied by the client.
Symbol	55	Y	The instrument, CCY Pair.
SettlDate	64	Y	Tenor of the instrument.
NoMDEntries	268	Y	Number of entries in the Market Data message. It is the number of liquidity bands. It can be set to 0 if there is no entry visible. The client must clear the depth in this case and ensure that no more order is sent.
➢ MDEntryType	269	Y	Side of this liquidity level 0 = Bid 1 = Offer H = Mid Rate
➢ MDUpdateAction	279	Y	Type of Market Data update action. 0 = New 1 = Change 2 = Delete
➢ MDEntryRefID	280	Cond	Refers to a previous MDEntryRefID. This does not apply to Pre Trade Mid Rate who are not going to have MDEntryRefID.
➢ MDEntryID	278	N	Identifier for this market data entry. Used in previously quoted orders to reference the quote when submitting an order to hit this entry. Only provided if New or Change if different from previous value.
➢ MDEntryPositionNo	290	N	Display position of a bid or offer, numbered from most competitive to least competitive, per market side.

			Only provided if New or Change if different from previous value.
➢ MDEntryPx	270	N	The price of this liquidity level. Only provided if New or Change if different from previous value.
➢ MDEntrySize	271	N	The size of this liquidity level. Only provided if New or Change if different from previous value.
➢ MDEntryDate	272	Optional[1] (provided by default)	The date of this liquidity level. Only provided if New or Change if different from previous value.
➢ MDEntryTime	273	N	The time of this liquidity level. Only provided if New or Change if different from previous value.
➢ Currency	15	N	The currency that MDEntryPx and MDEntrySize tags refer to. Only provided if New or Change if different from previous value.
➢ MinQty	110	N	Minimum quantity of an order to be executed. Only provided if New or Change if different from previous value.
➢ MDEntryOriginator	282	N	The ID of the liquidity provider as defined in section 2.3. Only provided if New or Change if different from previous value.
➢ SettlDate	64	N	The value date. Only provided if New or Change if different from previous value.
➢ MDEntryOrigTime	9122	Optional[1] (not provided by default)	The original time from the LP of this liquidity level.
➢ MDEntrySpotRate	1026	N	For forwards, the spot rate when available.

9.8.1. Pre-Trade Mid-Market Rate

Some banks require that the client receives a pre-trade mid-market rate when they provide it. When a Liquidity Provider provides such information, it will be sent to the client.

For incremental subscription, this pre-trade mid-rate entry will not have a normal lifecycle (CREATE, CHANGE, DELETE) on the MDUpdateAction. Instead, the MDUpdateAction will always be CHANGE and the MDEntryRefID will not be provided.

9.9. Market Data Incremental Refresh Examples

Initial image

When subscribing to incremental refresh, the first message received is of type "MarketDataSnapshotFullRefresh" and represents the current snapshot of the market. The MDEntryRefID (280) tag is specified on all new entries for future reference in subsequent updates.

Note that the MDEntryRefID is unique for a given subscription and the side. You can potentially receive the same MDEntryRefID twice in the same MarketDataSnapshot, one for a bid entry and one for an offer entry. Two market data snapshots from different subscriptions could also have the same MDEntryRefID. The MDEntryRefID unicity is for a given MDReqID and Side.

Tag name	Tag number	Value	Description
Header		W	Snapshot Full Refresh (Initial Image)
MDReqID	262	123	My request ID
Symbol	55	EUR/USD	The instrument, CCY Pair.
SettlDate	64	SP	SPOT
NoMDEntries	268	4	Number of repeating entries
➢ **MDEntryType**	**269**	**0**	**A BID side entry**
➢ MDEntryRefID	280	A	MDEntryRefID
➢ MDEntryID	278	0.B	MDEntryID
➢ MDEntryPx	270	1.312598	Price
➢ MDEntrySize	271	2000000	Quantity
➢ MDEntryDate	272	20130103	Date
➢ MDEntryTime	273	15:40:57.775	Time
➢ Currency	15	EUR	Currency
➢ MDEntryOriginator	282	COBA	Liquidity Provider
➢ SettlDate	64	20130105	Value Date
➢ **MDEntryType**	**269**	**0**	**A BID side entry**
➢ MDEntryRefID	280	B	MDEntryRefID
➢ MDEntryID	278	1.B	MDEntryID
➢ MDEntryPx	270	1.312593	Price
➢ MDEntrySize	271	5000000	Quantity
➢ MDEntryDate	272	20130103	Date
➢ MDEntryTime	273	15:40:57.775	Time
➢ Currency	15	EUR	Currency
➢ MDEntryOriginator	282	CITI	Liquidity Provider
➢ SettlDate	64	20130105	Value Date
➢ **MDEntryType**	**269**	**1**	**An OFFER side entry**
➢ MDEntryRefID	280	A	MDEntryRefID
➢ MDEntryID	278	0.O	MDEntryID
➢ MDEntryPx	270	1.312648	Price
➢ MDEntrySize	271	2000000	Quantity
➢ MDEntryDate	272	20130103	Date
➢ MDEntryTime	273	15:40:57.775	Time
➢ Currency	15	EUR	Currency
➢ MDEntryOriginator	282	CITI	Liquidity Provider
➢ SettlDate	64	20130105	Value Date
➢ **MDEntryType**	**269**	**1**	**An OFFER side entry**
➢ MDEntryRefID	280	B	MDEntryRefID
➢ MDEntryID	278	1.O	MDEntryID
➢ MDEntryPx	270	1.312653	Price
➢ MDEntrySize	271	5000000	Quantity
➢ MDEntryDate	272	20130103	Date

➢ MDEntryTime	273	15:40:57.775	Time
➢ Currency	15	EUR	Currency
➢ MDEntryOriginator	282	COBA	Liquidity Provider
➢ SettlDate	64	20130105	Value Date
Trailer			

The above message creates the resulting order book:

BID:

MDEntryRefID	MDEntryPx	MDEntrySize
A	1.312598	2M
B	1.312593	5M

OFFER:

MDEntryRefID	MDEntryPx	MDEntrySize
A	1.312648	2M
B	1.312653	5M

Adding a market data entry

Tag name	Tag number	Value	Description
Header		X	Incremental Refresh (Update)
MDReqID	262	123	My request ID
Symbol	55	EUR/USD	The instrument, CCY Pair.
SettlDate	64	SP	SPOT
NoMDEntries	268	1	Number of repeating entries
➢ **MDEntryType**	**269**	**0**	**A BID side entry**
➢ MDUpdateAction	279	0	New
➢ MDEntryRefID	280	C	MDEntryRefID
➢ MDEntryID	278	2.B	MDEntryID
➢ MDEntryPositionNo	290	2	Depth Position
➢ MDEntryPx	270	1.312592	Price
➢ MDEntrySize	271	3000000	Quantity
➢ MDEntryDate	272	20130103	Date
➢ MDEntryTime	273	15:40:58.015	Time
➢ Currency	15	EUR	Currency
➢ MDEntryOriginator	282	COBA	Liquidity Provider
➢ SettlDate	64	20130105	Value Date
Trailer			

The above message creates the resulting order book:

BID:

MDEntryRefID	MDEntryPx	MDEntrySize
A	1.312598	2M
B	1.312593	5M
C	1.312592	3M

OFFER:

MDEntryRefID	MDEntryPx	MDEntrySize
A	1.312648	2M
B	1.312653	5M

Updating a market data entry

Tag name	Tag number	Value	Description
Header		X	Incremental Refresh (Update)
MDReqID	262	123	My request ID
Symbol	55	EUR/USD	The instrument, CCY Pair.
SettlDate	64	SP	SPOT
NoMDEntries	268	1	Number of repeating entries
➢ **MDEntryType**	**269**	**0**	**A BID side entry**
➢ MDUpdateAction	279	1	Change
➢ MDEntryRefID	280	B	MDEntryRefID
➢ MDEntrySize	271	7000000	Quantity
➢ MDEntryTime	273	15:40:58.515	Time
Trailer			

The above message creates the resulting order book:

BID:

MDEntryRefID	MDEntryPx	MDEntrySize
A	1.312598	2M
B	1.312593	7M
C	1.312592	3M

OFFER:

MDEntryRefID	MDEntryPx	MDEntrySize
A	1.312648	2M
B	1.312653	5M

Removing a market data entry

Tag name	Tag number	Value	Description
Header		X	Incremental Refresh (Update)
MDReqID	262	123	My request ID
Symbol	55	EUR/USD	The instrument, CCY Pair.
SettlDate	64	SP	SPOT
NoMDEntries	268	1	Number of repeating entries
➢ **MDEntryType**	**269**	**0**	**A BID side entry**
➢ MDUpdateAction	279	2	Delete
➢ MDEntryRefID	280	A	MDEntryRefID
Trailer			

The above message creates the resulting order book:

BID:

MDEntryRefID	MDEntryPx	MDEntrySize
B	1.312593	7M
C	1.312592	3M

OFFER:

MDEntryRefID	MDEntryPx	MDEntrySize
A	1.312648	2M
B	1.312653	5M

Clearing the depth

Tag name	Tag number	Value	Description
Header		W	Snapshot Full Refresh (Clear Cache)

MDReqID	262	123	My request ID
Symbol	55	EUR/USD	The instrument, CCY Pair.
SettlDate	64	SP	SPOT
NoMDEntries	268	0	Number of repeating entries
Trailer			

The above message creates the resulting order book:

BID:

MDEntryRefID	MDEntryPx	MDEntrySize

OFFER:

MDEntryRefID	MDEntryPx	MDEntrySize

10) Trading Session List (LP Status)

The client can request the status of the connections between FSS and the Liquidity providers by submitting a "TradingSessionListRequest". In response, the client will receive a "TradingSessionList" indicating the status of every liquidity provider he is allowed to trade with.

Moreover, he will receive updates every time the status of a given LP changes.

10.1. Trading Session List Request (Type BI)

Tag name	Tag number	Required	Description
TradSesReqID	335	Y	Unique identifier for this request.
SubscriptionRequestType	263	Y	Possible Values: 1 = Subscribe to updates 2 = Unsubscribe

To get the actual status of liquidity providers and subscribe to updates, a "TradingSessionListRequest" message is sent with the tag "SubscriptionRequestType" set to "Subscribe" (1).

Example:

8=FIX.4.4|9=109|35=BI|34=2|49=CLIENT|52=20150901-12:22:50.159|56=FSS|335=SLRkruDaT/hSdXiRu8UFy2zlg|263=1|10=045|

To unsubscribe from trading sessions lists updates, a "TradingSessionListRequest" message is sent with the unique identifier used in the initial subscription request and the tag "SubscriptionRequestType" set to "Unsubscribe" (2).

Example:

8=FIX.4.4|9=109|35=BI|34=2|49=CLIENT|52=20150901-12:22:50.159|56=FSS|335=SLRkruDaT/hSdXiRu8UFy2zlg|263=2|10=045|

10.2. Trading Session List (Type BJ)

Tag name	Tag number	Required	Description
TradSesReqID	335	Y	The original request id sent by the client.
NoTradingSessions	386	Y	Number of sessions returned
➢ TradingSessionID	336	Y	The Liquidity Provider ID
➢ TradSesStatus	340	Y	The status of the connection with the LP 2 = Open 7 = Suspicious

An example of a TradingSessionList message:

8=FIX.4.4|9=109|35=BJ|34=2|49=FSS|52=20150901-12:22:50.159|56=CLIENT|335=SLRkruDaT/hSdXiRu8UFy2zlg|386=4|336=BAML|340=2|336=CITI|340=2|336=JPMC|340=7|336=GS|340=2|10=045|

10.3. Trading Session list Update Report (Type BS)

If the "TradingSessionListRequest" was unsuccessful (to subscribe or to unsubscribe), a "TradingSessionListUpdateReport" message will be sent back to the client. The tag 'Text' will contain the reason of the reject.

Tag name	Tag number	Required	Description
TradSesReqID	335	Y	The original request id sent by the client.
Text	58	Y	Reason of the reject

An example of "TradingSessionListUpdateReport":

8=FIX.4.4|9=109|35=BS|34=2|49=FSS|52=20150901-12:22:50.159|56=CLIENT|335=SLRkruDaT/hSdXiRu8UFy2zlg|58=FAIL|10=045|

11) Orders

The only orders that are currently supported are Single Fill-Or-Kill / Immediate-Or-Cancel Orders with a reference to a previously sent price and Limit Orders. The different execution capabilities supported by the liquidity providers are listed in section 2.3.1.

11.1. Submitting Orders

To submit a new order, the client needs to send a "NewOrderSingle" message. The side of the orders is interpreted as being:

- submitted from the client's perspective,
- related to the currency in which the quantity is specified (tag 15).

In other words, if the quantity is expressed in the base currency, the client will buy on offer prices, and vice versa (sell on bid prices). Conversely, if the quantity is expressed in the term currency, the client will sell on offer prices and vice versa (buy on bid prices).

It is possible to indicate on the NewOrderSingle a value to identify the trader who is placing the order. This value will be passed to the bank when supported by the bank and will be returned on the Execution Report. This could be used either by the client or by the bank for statistics about the individuals who placed orders.

11.2. Previously-quoted Orders

Each previously-quoted order (40=D) will need to reference the unique reference and the price of a previously sent price.

Available liquidity differs depending on the market data subscription. Subscribing to all available liquidity returns distinct liquidity bands all of which can be traded separately[2]. Subscribing to explicit bands (by providing a value in "RequestedSize" of the MarketDataRequest subscription message) would return different prices from the underlying available liquidity; in this case only one band can be traded at a time.

11.3. Limit Orders

Different options are available for limit orders:

- orders resting on FSS and monitoring liquidity providers streams
    - aggressive orders
        - IOC
        - FOK
    - resting orders
        - GTC
        - GTD
- orders resting on the liquidity provider side with Direct Market Access Strategy (DMA)

A resting limit order can be active at most until the end of the week and an execution report canceled will eventually be sent at the end of the week to notify this fact.

[2] Note that Morgan Stanley prefers that when a client sweeps the book using the Pass Through mode, it is done from the top level down. This will help ensure (but not guarantee in *every* scenario) that the trades reach MS in the proper sequence and significantly reduce the chance of rejects.

11.3.1. FSS Limit Orders

FSS limit orders are orders resting on FSS side: FSS will monitor the liquidity available amongst the target liquidity providers and will aggress the liquidity matching the requirement of the order on the LP side.

Before placing such orders, a trading request is required with (*resp.* without) prices sent in return if the view of the market is needed (*resp.* not needed) (see section 9.3).

FSS will monitor the liquidity of all the Liquidity Providers specified on these orders as PartyID with PartyRole equals to 35 (Liquidity Provider). If no PartyID is specified on the order, all the configured LPs could be targeted by the rule. Specifying PartyIDs on the order is the way to restrict the targetable LPs, it should not be done on the MarketDataRequest used to Request for Trading.

11.3.2. Direct Market Access Strategy (DMA)

The orders sent with DMA strategy (847=2000) will be passed through to the target liquidity provider specified in PartyID with PartyRole equals to 35 (Liquidity Provider).

In this case only one LP can be specified as a PartyID because the order will be resting on the LP side.

11.4. NewOrderSingle Message Definition (Type D)

Tag name	Tag number	Required	Description
Account	1	N	Account mnemonic as agreed between client and FSS to allow a specific handling of the order by the liquidity provider. This is an optional value.
ClOrdID	11	Y	Unique identifier supplied by the client for this order.
NoPartyIDs	453	N	Number of repeating entries.
➢ PartyID	448	Y	The party identifier (possible values depend on the party role). Required if NoPartyIDs > 0.
➢ PartyIDSource	447	Y	Required if NoPartyIDs > 0. Valid values: D = Proprietary/Custom code
➢ PartyRole	452	Y	Required if NoPartyIDs > 0. Valid values: 3 = Client ID (internal id or name of the trader who is placing the trade, at most one value of this type) 35 = Liquidity Provider (codes are listed in section 2.3, only one value of this type when TargetStrategy=2000(DMA))
Symbol	55	Y	The instrument CCY pair.
Side	54	Y	Side of the order. This is from the client's perspective. Valid values: 1 = Buy

			2 = Sell
TransactTime	60	Y	The timestamp in UTC for this order.
OrderQty	38	Y	The size of the order.
OrdType	40	Y	Order type. Valid values: 2 = Limit D = Previously Quoted
Currency	15	N	The currency that this order refers to. Only base currency (ccy1) trading is supported for limit orders (40=2).
Price	44	Y	The limit price of the order.
MDEntryID	278	Cond	The id of the previously quoted data. Required when 40=D.
TimeInForce	59	Y	Specifies how long the order remains in effect. Valid values: 1 = Good Till Cancel (GTC) 3 = Immediate-Or-Cancel (IOC)[3] 4 = Fill-Or-Kill (FOK) 6 = Good Till Date (GTD)
SettlDate	64	N	Value date of the previously quoted data.
SlippageOffset[4]	639	N	Slippage amount.
MDEntrySpotRate	1026	Cond	For forwards, the spot rate of the previously quoted data.
TargetStrategy	847	N	The target strategy of the order. Valid values: 1 = VWAP, the calculated VWAP price (for one LP) must be populated in tag 44 (Price)[5] 2000 = DMA – Direct Market Access
ExpireTime	126	Cond	The expiry time of the order expressed as a UTC time. Required for GTD order type.

[3] As of today, only the following LPs support *true* IOC orders (*ie*. a partial execution can occur):

- MS
- UBS

Submitting an IOC order to another LP will have the same behavior as a FOK order, either rejected or fully filled.

[4] When adding slippage on the price originally quoted, the client is informing the bank(s) that it is authorizing the bank to fill the order at a worse price as specified by the client versus having the bank reject the order. Before sending an order with slippage, clients should speak directly to the bank to understand and agree on how orders with slippage are handled by the bank. FXSpotStream is not involved in any way in the determination of the price at which an order is filled by the bank.

[5] Calculated VWAP prices must be rounded to the closest tenth of a pip: rounded down for bid prices and rounded up for offer prices. Also these LP-specific rules must be followed:

- Citi requires that the first rung is hit when placing VWAP orders.
- UBS requires that the rung cumulating the desired quantity is hit, *eg.* if UBS streams 1M (1st rung), 2M (2nd rung), 3M (3rd rung) then the 2nd rung must be hit for a VWAP order of 3M (the 2nd rung cumulates 1M + 2M).

11.5. Submitting Order Examples

For the next examples, we consider the following book for bid and offer:

BID:

MDEntryID	MDEntryPx	MDEntrySize
quoted.bid.0	1. 31257	3M
quoted.bid.1	1. 312565	6M
quoted.bid.2	1. 31256	12M
quoted.bid.3	1. 312555	5M

OFFER:

MDEntryID	MDEntryPx	MDEntrySize
quoted.offer.0	1.312614	4M
quoted.offer.1	1.31262	6M
quoted.offer.2	1.312625	12M
quoted.offer.3	1.31263	5M

11.5.1. Buy order

In order to hit the top of the book, we would send the following order:

ClOrdID	Symbol	Currency	Side	OrderQty	OrderType	Price	MDEntryID	TIF
ClOrdID1	EUR/USD	EUR	Buy	4M	D	1.312614	quoted.offer.0	4 (FOK)

8=FIX.4.4|9=199|35=D|49=Client|56=FSS|34=69|52=20121019-15:32:34.212|11=ClOrdID1
|55=EUR/USD|54=1|60=20121019-10:32:34.000|38=4000000|40=D|44=1.312614
|15=EUR|59=4|278=quoted.offer.0|10=009|

11.5.2. Sell Order

In order to hit the top of the book, we would send the following order:

ClOrdID	Symbol	Currency	Side	OrderQty	OrderType	Price	MDEntryID	TIF
ClOrdID2	EUR/USD	EUR	Sell	3M	D	1.31257	quoted.bid.0	4 (FOK)

8=FIX.4.4|9=199|35=D|49=Client|56=FSS|34=70|52=20121019-15:42:29.216|11=ClOrdID2
|55=EUR/USD|54=2|60=20121019-10:42:29.000|38=3000000|40=D|44=1.31257
|15=EUR|59=4|278=quoted.bid.0|10=038|

11.5.3. Buy order on the term currency

In order to hit the top of the book, we would send the following order:

ClOrdID	Symbol	Currency	Side	OrderQty	OrderType	Price	MDEntryID	TIF
ClOrdID3	EUR/USD	USD	Buy	3937710	D	1.31257	quoted.bid.0	4 (FOK)

8=FIX.4.4|9=199|35=D|49=Client|56=FSS|34=70|52=20121019-15:42:29.216|11=ClOrdID3
|55=EUR/USD|54=1|60=20121019-10:42:29.000|38=3937710|40=D|44=1.31257
|15=USD|59=4|278=quoted.bid.0|10=038|

11.5.4. Sell order on the term currency

In order to hit the top of the book, we would send the following order:

ClOrdID	Symbol	Currency	Side	OrderQty	OrderType	Price	MDEntryID	TIF

ClOrdID4	EUR/USD	USD	Sell	5250456	D	1.312614	quoted.offer.0	4 (FOK)

8=FIX.4.4|9=199|35=D|49=Client|56=FSS|34=69|52=20121019-15:32:34.212|11=ClOrdID4
|55=EUR/USD|54=2|60=20121019-10:32:34.000|38=5250456|40=D|44=1.312614
|15=USD|59=4|278=quoted.offer.0|10=009|

11.6. Executions

All order responses are sent using the "ExecutionReport" message. This includes any fills and rejections. Currently each order can either be rejected or fulfilled with a complete fill.

When the client has sent an order, he will get an ExecutionReport acknowledging the reception of the order by FSS. The order status can be either PENDING_NEW if FSS could process the order or REJECTED if FSS could not process the order.

Then, the client will receive an execution report with the status of the execution reported by the bank, either FILLED or CANCELED or REJECTED.

If FSS does not receive a response from the bank within 4 seconds, both the client and the bank will be notified by email.

11.7. ExecutionReport Message Definition (Type 8)

Tag name	Tag number	Required	Description
Account	1	N	Account mnemonic as agreed between client and FSS as specified in corresponding NewOrderSingle (see 11.4). It will be set with the value provided in the NewOrderSingle.
OrderID	37	Y	Unique identifier supplied by FSS for this order.
SecondaryOrderID	198	N	Can be used to provide order id used by exchange or executing system.
ClOrdID	11	Y	The original client unique identifier (ESP case). Or identifier of the cancel request.
OrigClOrdID	41	N	For a response to a cancel request, the original client unique identifier for the order.
NoPartyIDs	453	N	Number of repeating entries. Valid values: 1

➢ PartyID	448	N	The party identifier ie the internal id or name of the trader who is placing the trade
➢ PartyIDSource	447	N	Valid values: D = Proprietary/Custom code
➢ PartyRole	452	N	Valid values: 3 = Client ID
ExecID	17	Y	Unique identifier specific to this execution message supplied by FSS.
SecondaryExecID	527	N	Assigned by the bank who received the order.
ExecType	150	Y	Describes the specific execution while OrdStatus (39) will always identify the current order status Valid values: 0 = New A = Pending_New 1 = Partial Fill 2 = Fill 4 = Canceled 6 = Pending_Canceled 8 = Rejected C = Expired
OrdStatus	39	Y	Identifies current status of order. Valid values: 0 = New A = Pending_New 1 = Partially Filled 2 = Filled 4 = Canceled 6 = Pending_Canceled 8 = Rejected C = Expired
Symbol	55	Y	The instrument CCY pair
Side	54	Y	Side of the order. This is from the client's perspective (only buys and Sells are supported): 1 = Buy 2 = Sell
OrderQty	38	Y	The size of the order. For swaps, this refers to the near leg.
OrdType	40	Y	The order type.
Price	44	N	Dealt all-in-rate price of the order. For swaps, this refers to the near leg.
Currency	15	Y	The currency that this order refers to.
TimeInForce	59	Y	Specifies how long the order remains in effect. Valid values: 1 = Good Till Cancel (GTC) 3 = Immediate-Or-Cancel (IOC)

			4 = Fill-Or-Kill (FOK) 6 = Good Till Date (GTD)
SettlDate	64	N	The settlement date YYYYMMDD. For swaps, this refers to the near leg.
LastQty	32	Y	Quantity of this fill or 0 if not fill. For swaps, this refers to the near leg.
LastPx	31	Y	Price of this fill or 0 if not fill. For swaps, this refers to the near leg.
LeavesQty	151	Y	The quantity open for execution
CumQty	14	Y	The total cumulative executed quantity of the order = OrderQty – LeavesQty
AvgPx	6	Y	The average price of all fills, or 0 if no fill. For swaps, this refers to the near leg.
TransactTime	60	Y	The timestamp in UTC for this execution.
Text	58	N	Description of this execution, or reason for rejection.
TradeDate	75	N	The trade date in the format YYYYMMDD.
OrdRejectReason	103	N	Code to identify reason for order rejection.
MDEntryID	278	N	The id of the previously quoted data. ESP case.
LastMkt	30	Cond	Not present on Pending New Execution Reports (39=A). Present otherwise. Market of execution for last fill, or an indication of the market where an order was routed. The ID of the liquidity provider as defined in section 2.3.
QuoteRespID	693	N	Unique identifier for this quote response provided by the client. RFS case.
QuoteID	117	N	The id of the previously quoted data. RFS case.
QuoteReqID	131	N	The id of the corresponding Quote-Request. RFS case.
SettlType	63	N	Standard tenor type as described in section 5) or YYYYMMDD for a broken date. For swaps, this refers to the near leg.
SettlType2	9999	N	Standard tenor type as described in section 5) or YYYYMMDD for a broken date. For swaps, this refers to the far leg.
SettlDate2	193	N	The settlement date YYYYMMDD. For swaps, this refers to the far leg.

LastSpotRate	194	N	Spot rate of the near leg.
LastSpotRate2	6161	N	Spot rate of the far leg.
LastForwardPoints	195	N	Forward points of the near leg.
OrderQty2	192	N	Size of the far leg.
Price2	640	N	Dealt all-in-rate price of the far leg.
LastForwardPoints2	641	N	Forward points of the far leg.
LastQty2	6808	N	Quantity of this fill or 0 if not fill. For swaps, this refers to the far leg.
LastPx2	6160	N	Price of this fill or 0 if not fill. For swaps, this refers to the near leg.
LeavesQty2	6164	N	Far quantity open for execution.
CumQty2	6165	N	The total cumulative executed far quantity of the order = OrderQty2 – LeavesQty2
AvgPx2	6159	N	The average price of all fills, or 0 if no fill. For swaps, this refers to the far leg.
UTIPrefix	20001	N	UTI prefix near leg
UTI	20002	N	Unique UTI ID near leg
UTIPrefix2	20003	N	UTI prefix far leg
UTI2	20004	N	Unique UTI ID far leg
TargetStrategy	847	N	The target strategy of the order. Valid values: 1 = VWAP 2000 = DMA routing rule

11.8. Order Execution Examples

Below are examples of a couple of scenarios on how orders are handled by the system, and the types of messages that are sent back to the client.

11.8.1. Order Rejected

Example: Client submits an order to Buy 1000000 GBP/USD for 1.4773 and was rejected.

OrdStatus	ExecType	OrderQty	CumQty	LeavesQty	LastQty	Description
Rejected	Rejected	1000000	0	0	0	Order rejected immediately

11.8.2. Order Filled

Example: Client submits an order to Buy 1000000 GBP/USD for 1.4773 and was filled.

OrdStatus	ExecType	OrderQty	CumQty	LeavesQty	LastQty	Description
Pending_New	Pending_New	1000000	0	1000000	0	Accepted
Filled	Fill	1000000	1000000	0	1000000	Trade execution and order filled

11.8.3. Order Partially Filled (IOC orders)

Example: Client submits an order to Buy 1000000 GBP/USD for 1.4773 and was filled up to 800000.

OrdStatus	ExecType	OrderQty	CumQty	LeavesQty	LastQty	Description
Pending_New	Pending_New	1000000	0	1000000	0	Accepted
Partially Filled	Partial Fill	1000000	800000	200000	800000	Trade execution and order partially filled
Cancelled	Cancel	1000000	800000	200000	0	Remaining quantity is canceled

11.9. Canceling Orders

Only GTD and GTC orders can be canceled. IOC and FOK order cannot be canceled as their time in force is considered atomic.

A successful order cancelation request will result in an ExecutionReport with OrdStatus set to 4 (Canceled). Unsuccessful cancelation requests will result in an OrderCancelReject.

11.9.1. OrderCancelRequest (Type F)

Tag name	Tag number	Required	Description
ClOrdID	11	Y	A unique identifier for the cancel request supplied by the client.
OrigClOrdID	41	Y	The original client unique identifier for the order.
Symbol	55	Y	The instrument CCY pair
TransactTime	60	Y	Time this cancel request was initiated by the client.

11.9.2. OrderCancelReject (Type 9)

Tag name	Tag number	Required	Description
OrderID	37	N	Unique identifier for the order as provided by FSS.
ClOrdID	11	Y	A unique identifier for the cancel request supplied by the client.
OrigClOrdID	41	Y	The original client unique identifier for the order.
Symbol	55	Y	The instrument CCY pair
OrdStatus	39	N	Status of the order after this cancel reject is applied.
CxlRejResponseTo	434	Y	Always 1 = Order Cancel Request
CxlRejReason	102	N	
Text	58	N	Textual Description of the reject reason.

11.10. RFS

11.10.1. QuoteRequest (Type R)

Tag name	Tag number	Required	Description
QuoteReqID	131	Y	A unique identifier supplied by the client.
Account	1	N	Account mnemonic as agreed between client and FSS.
ExpireTime	126	N	The quote request will expire after a timeout equals to: TimeOut = ExpireTime – SendingTime. Default to 120s if not set or if the resulting TimeOut value does not respect the following: 10s < TimeOut < 120s.
NoRelatedSym	146	Y	Number of repeating symbols. Valid value: 1.
➢ Symbol	55	Y	The instrument, ccy pair.
➢ Side	54	N	Side of the quote for a One-Way request: 1 = Buy 2 = Sell If the value is not set, this is a Two-Way request. For a swap, refers to the far leg.
➢ OrderQty	38	Y	The requested size expressed in the ccy specified in tag 15. For swaps, this refers to the near leg.
➢ SettlType	63	Y	Standard tenor type as described in table 5) or YYYYMMDD for a broken date. For swaps, this refers to the near leg.
➢ SettlType2	9999	N	Standard tenor type as described in table 5) or YYYYMMDD for a broken date. For swaps, this refers to the far leg.
➢ OrderQty2	192	N	The requested size expressed in the ccy specified in tag 15. For swaps, this refers to the far leg.
➢ Currency	15	N	The currency that this request refers to.
NoPartyIDs	453	N	Repeating group below should contain unique combinations of PartyID, PartyIDSource, and PartyRole
➢ PartyID	448	Y	Used to identify source of PartyID. When 452=3, trader ID value. When 452=35, valid values are listed in Liquidity Provider ID (section 8). Required if NoPartyIDs > 0.
➢ PartyIDSource	447	Y	Required if NoPartyIDs > 0. Valid value = D.
➢ PartyRole	452	Y	Required if NoPartyIDs > 0. Valid values:

			Valid values: 3 = Client ID (internal id or name of the trader who is placing the trade, at most one value of this type) 35 = Liquidity Provider.

11.10.2. MassQuote (Type i)

Tag name	Tag number	Required	Description
QuoteReqID	131	Y	The original unique identifier supplied by the client.
Symbol	55	Y	The instrument, ccy pair.
SettlType	63	Y	Standard tenor type as described in table 5) or YYYYMMDD for a broken date. For swaps, this refers to the near leg.
SettlType2	9999	N	Standard tenor type as described in table 5) or YYYYMMDD for a broken date. For swaps, this refers to the far leg.
NoQuoteEntries	295	Y	Number of entries in the Mass Quote message. Depends on the number of LP involved.
➢ QuoteEntryID	299	Y	Unique identifier for this single entry quote and side. Updates to this entry quote bid/offer will use this ID to reference the entry quote.
➢ Side	54	Y	Side of this quote 1 = Bid 2 = Offer
➢ Currency	15	N	The currency that this response refers to.
➢ SettlDate	64	N	Value date. For swaps, this refers to the near leg.
➢ SettlDate2	193	N	Value date. For swaps, this refers to the far leg.
➢ BidSpotRate	188	N	Bid spot rate of the near leg.
➢ OfferSpotRate	190	N	Offer spot rate of the near leg.
➢ BidSpotRate2	6162	N	Bid spot rate of the far leg.
➢ OfferSpotRate2	6163	N	Offer spot rate of the far leg.
➢ BidPx	132	N	Bid all-in-rate of the near leg.
➢ BidSize	134	N	Bid size of the near leg.
➢ BidForwardPoints	189	N	Bid forward points of the near leg.
➢ OfferPx	133	N	Offer all-in-rate of the near leg.
➢ OfferSize	135	N	Offer size of the near leg.
➢ OfferForwardPoints	191	N	Offer forward points of the near leg.

➢ BidPx2	7576	N	Bid all-in-rate of the far leg.
➢ BidSize2	6052	N	Bid size of the far leg.
➢ BidForwardPoints2	642	N	Bid forward points of the far leg.
➢ OfferPx2	7577	N	Offer all-in-rate of the far leg.
➢ OfferSize2	6053	N	Offer size of the far leg.
➢ OfferForwardPoints2	643	N	Offer forward points of the far leg.
➢ BidSwapPoints	4539	N	Bid swap points.
➢ OfferSwapPoints	4540	N	Offer swap points.
➢ ValidUntilTime	62	N	Indicates expiration time in UTC.
➢ MDEntryDate	272	N	The date of this quote entry.
➢ MDEntryTime	273	N	The time of this quote entry.
➢ MDEntryOriginator	282	N	The ID of the liquidity provider as defined in 2.3.
NoMDEntries	268	Y	Number of entries in the Mass Quote message. It is the number of mid rate entries.
➢ MDEntryType	269	Y	H = Mid Rate Price
➢ MDEntryOriginator	282	Y	The ID of the liquidity provider as defined in 2.3.
➢ MidPx	631	N	Mid price of the near leg as provided by the LP.
➢ FarMidPx	9998	N	Mid price of the far leg as provided by the LP.

11.10.3. MassQuoteAcknowledgement (Type b)

Tag name	Tag number	Required	Description
QuoteReqID	131	Y	Identifier of the quote request that this ack refers to.
QuoteStatus	297	Y	Identifies the status of the quote acknowledgement. Valid values: 0 = Accepted 4 = Canceled All 5 = Rejected 7 = Expired
QuoteRejectReason	300	N	Reason of the rejection.
Text	58	N	Full description for rejection.
Symbol	55	N	The instrument ccy pair.

11.10.4. QuoteResponse (Type AJ)

Tag name	Tag number	Required	Description
QuoteRespID	693	Y	Unique identifier for this quote response provided by the client.
QuoteID	117	Y	Identifier of the quote that this quote response refers to (QuoteEntryID from the MassQuote).

QuoteRespType	694	Y	Valid values: 1 = Hit/Lift
Symbol	55	Y	The instrument ccy pair.
QuoteReqID	131	Y	Identifier of the quote request that this quote response refers to.
Side	54	Y	Side of the hit. This is from the client's perspective. Valid values: 1 = Buy 2 = Sell
SettlType	63	Y	Standard tenor type as described in table 5) or YYYYMMDD for a broken date as provided in the quoteRequest. For swaps, this refers to the near leg.
SettlDate	64	Y	Value date as provided in the targeted quote. For swaps, this refers to the near leg.
SettlType2	9999	Cond	Standard tenor type as described in table 5) or YYYYMMDD for a broken date as provided in the quoteRequest. For swaps, this refers to the far leg.
SettlDate2	193	Cond	Value date as provided in the targeted quote. For swaps, this refers to the far leg.
Currency	15	N	The currency that this response refers to.
BidSpotRate	188	Cond	Bid spot rate of the near leg.
OfferSpotRate	190	Cond	Offer spot rate of the near leg.
BidSpotRate2	6162	Cond	Bid spot rate of the far leg.
OfferSpotRate2	6163	Cond	Offer spot rate of the far leg.
BidSize	134	Cond	Bid size of the near leg.
BidPx	132	Cond	Bid all-in-rate of the near leg.
BidForwardPoints	189	N	Bid forward points of the near leg (for information only).
OfferSize	135	Cond	Offer size of the near leg.
OfferPx	133	Cond	Offer all-in-rate of the near leg.
OfferForwardPoints	191	N	Offer forward points of the near leg (for information only).
BidSize2	6052	Cond	Bid size of the far leg.
BidPx2	7576	Cond	Bid all-in-rate of the far leg.
BidForwardPoints2	642	N	Bid forward points of the far leg (for information only).
OfferSize2	6053	Cond	Offer size of the far leg.
OfferPx2	7577	Cond	Offer all-in-rate of the far leg.
OfferForwardPoints2	643	N	Offer forward points of the far leg (for information only).
TransactTime	60	Y	The timestamp in UTC for this response.
UTIPrefix	20001	N	UTI prefix near leg
UTI	20002	N	Unique UTI ID near leg
UTIPrefix2	20003	N	UTI prefix far leg
UTI2	20004	N	Unique UTI ID far leg

11.10.5. QuoteCancel (Type Z)

Tag name	Tag number	Required	Description
QuoteReqID	131	Y	The original unique identifier of the quote to be canceled.
QuoteRespID	693	Y	Unique identifier for this quote cancel provided by the client.
Symbol	55	N	The instrument ccy pair.

11.10.6. RFS Message Examples

Forward trading and broken dates

QuoteRequest

8=FIX.4.4|9=154|35=R|34=837|49=STR.RFS.NY.UAT.CLIENT|52=20150127-14:35:05.846|56=FSS|131=QR_EUR/USD_RFS_1422369305835|146=1|55=EUR/USD|15=EUR|38=10000|63=20150205|10=242|

MassQuote

8=FIX.4.4|9=0697|35=i|49=FSS|56=STR.RFS.NY.UAT.CLIENT|52=20150127-14:35:06.481|34=951|131=QR_EUR/USD_RFS_1422369305835|55=EUR/USD|63=20150205|295=4|299=6JJ00a00000000+|54=1|132=1.137067|134=100000|188=1.137|189=0.000067|15=EUR|64=20150205|272=20150127|273=14:35:06.480|282=HSBC|299=61a28.60249752.0.Q+|54=1|132=1.136717|134=100000|188=1.13666|189=0.000057|15=EUR|64=20150205|272=20150127|273=14:35:05.897|282=COBA|299=6JJ00a00000000+|54=2|133=1.137171|135=100000|190=1.1371|191=0.000071|15=EUR|64=20150205|272=20150127|273=14:35:06.480|282=HSBC|299=61a28.60249752.0.Q+|54=2|133=1.137552|135=100000|190=1.13747|191=0.000082|15=EUR|64=20150205|272=20150127|273=14:35:05.897|282=COBA|268=1|269=H|282=COBA|631=1.13714|10=000|

QuoteResponse

8=FIX.4.4|9=261|35=AJ|34=832|49=TRD.RFS.NY.UAT.CLIENT|52=20150127-14:35:06.598|56=FSS|15=EUR|54=2|55=EUR/USD|60=20150127-14:35:06|63=20150205|64=20150205|117=6JJ00a00000000+|131=QR_EUR/USD_RFS_1422369305835|132=1.137067|134=100000|188=1.137|693=NOSvg6O4zraEeO/s5iTSjTl2g|694=1|10=007|

Here both 63 and 64 tags must be provided:
- 63 must be set as provided in the QuoteRequest
- 64 must be set with the value date of the targeted entry as returned by the LPs in the MassQuote message.

Be aware that values of tags 63 and 64 may be different as FSS passes through the value of tag 64 from the LPs.

Execution Reports

8=FIX.4.4|9=0368|35=8|34=838|49=FSS|56=TRD.RFS.NY.UAT.CLIENT|52=20150127-14:35:06.606|37=O20150127L1000031313|17=E20150127L1000018135|150=A|39=A|55=EUR/USD|54=2|693=NOSvg6O4zraEeO/s5iTSjTl2g|131=QR_EUR/USD_RFS_1422369305835|117=6JJ00a00000000+|63=20150205|64=20150205|38=100000|44=1.137067|151=100000|14=0|32=0|31=0|6=0|194=1.137|195=0.000067|40=D|15=EUR|59=4|60=20150127-14:35:06.000|10=243|

8=FIX.4.4|9=0495|35=8|34=839|49=FSS|56=TRD.RFS.NY.UAT.CLIENT|52=20150127-14:35:07.218|37=O20150127L1000031313|198=LNQ020019CNX2F8K|17=E20150127L1000018138|527=LNQ020019CNX2F8K|150=2|39=2|55=EUR/USD|54=2|693=NOSvg6O4zraEeO/s5iTSjTl2g|131=QR_EUR/USD_RFS_1422369305835|117=6JJ00a00000000+|63=20150205|64=20150205|38=100000|44=1.137067|151=0|14=100000|32=100000|31=1.137067|6=1.137067|194=1.137|195=0.000067|20001=1N6DM40ZPS|20002=NYO20150127L1000031313|40=D|15=EUR|59=4|60=20150127-14:35:07.216|75=20150127|30=HSBC|10=028|

Swap trading and broken dates

QuoteRequest

8=FIX.4.4|9=178|35=R|34=19|49=STR.RFS.NY.UAT.CLIENT|52=20150127-12:19:29.586|56=FSS|131=QR_EUR/USD_RFS_1422361169552|146=1|55=EUR/USD|15=EUR|38=1000000|63=20150129|192=100000|9999=20150429|10=124|

MassQuote

8=FIX.4.4|9=0556|35=i|49=FSS|56=STR.RFS.NY.UAT.CLIENT|52=20150127-12:19:30.427|34=43|131=QR_EUR/USD_RFS_1422361169552|55=EUR/USD|63=20150129|9999=20150429|295=2|299=6JJ00300000001+|54=1|133=1.12883|7576=1.129704|135=100000|6052=100000|190=1.12883|6162=1.12883|191=0|642=0.000874|4539=0.000874|15=EUR|64=20150129|193=20150429|272=20150127|273=12:19:30.425|282=HSBC|299=6JJ00300000001+|54=2|132=1.12873|7577=1.129644|134=100000|6053=100000|188=1.12873|6163=1.12873|189=0|643=0.000914|4540=0.000914|15=EUR|64=20150129|193=20150429|272=20150127|273=12:19:30.425|282=HSBC|268=0|10=100|

QuoteResponse

8=FIX.4.4|9=327|35=AJ|34=17|49=TRD.RFS.NY.UAT.CLIENT|52=20150127-12:19:31.715|56=FSS|15=EUR|54=1|55=EUR/USD|60=20150127-12:19:31|63=20150129|64=20150129|117=6JJ00300000001+|131=QR_EUR/USD_RFS_1422361169552|132=1.12873|134=100000|188=1.12873|193=20150429|693=NOSajLtdAslja+nAiX5oYB2jQ|694=1|6053=100000|6163=1.12873|7577=1.129644|9999=20150429|10=109|

Here both 63 and 64 tags (near leg) and both 9999 and 193 tags (far leg) must be provided:
- 63 and 9999 must be set as provided in the QuoteRequest
- 64 and 193 must be set with the value date of the targeted entry as returned by the LPs in the MassQuote message.

Be aware that values of tags 63 and 64 on one hand, and the values of tags 9999 and 193 on the other hand, may be different as FSS passes through the value of tag 64 and 193 from the LPs.

Execution Reports

8=FIX.4.4|9=0504|35=8|34=18|49=FSS|56=TRD.RFS.NY.UAT.CLIENT|52=20150127-12:19:31.273|37=O20150127L1000031213|17=E20150127L1000018075|150=A|39=A|55=EUR/USD|54=1|693=NOSajLtdAslja+nAiX5oYB2jQ|131=QR_EUR/USD_RFS_1422361169552|117=6JJ003000000001+|63=20150129|64=20150129|9999=20150429|193=20150429|44=1.12873|640=1.129644|38=100000|192=100000|194=1.12873|6161=1.12873|195=0|641=0.000914|151=100000|14=0|32=0|6164=100000|6165=0|6808=0|31=1.12873|6=1.12873|6160=1.129644|6159=1.129644|40=D|15=EUR|59=4|60=20150127-12:19:31.000|10=086|

8=FIX.4.4|9=0670|35=8|34=19|49=FSS|56=TRD.RFS.NY.UAT.CLIENT|52=20150127-12:19:31.876|37=O20150127L1000031213|198=LNQ020019CNNA5L6|17=E20150127L1000018078|527=LNQ020019CNNA5L6|150=2|39=2|55=EUR/USD|54=1|693=NOSajLtdAslja+nAiX5oYB2jQ|131=QR_EUR/USD_RFS_1422361169552|117=6JJ003000000001+|63=20150129|64=20150129|9999=20150429|193=20150429|44=1.12873|640=1.129644|38=100000|192=100000|194=1.12873|6161=1.12873|195=0|641=0.000914|151=0|14=100000|32=100000|6164=0|6165=100000|6808=100000|31=1.12873|6=1.12873|6160=1.129644|6159=1.129644|20001=1N6DM40ZPS|20002=NYO20150127L1000031213|20003=1N6DM40ZPS|20004=NYO20150127L1000031213-2|40=D|15=EUR|59=4|60=20150127-12:19:31.874|75=20150127|30=HSBC|10=200|

12) Standard FIX Message Definitions

12.1. Standard Message Header

Tag name	Tag number	Required	Description
BeginString	8	Y	Set to “FIX.4.4”
BodyLength	9	Y	Length of the message body.
MessageType	35	Y	The message type.
SenderCompID	49	Y	Message sender ID, a pre-defined ID agreed by both parties. Value supplied separately outside this document. Will be assigned during on-boarding.
TargetCompID	56	Y	Message receiver ID, a pre-defined ID agreed by both parties. Value supplied separately outside this document. Will be assigned during on-boarding.
MsgSeqNum	34	Y	The sequence number for this message.
SendingTime	52	Y	Time of message transmission (UTC timestamp at the source).
PossDupFlag	43	N	Indicates possible retransmission of message with this sequence number
PossResend	97	N	Indicates that message may contain information that has been sent under another sequence number.
OrigSendingTime	122	N	Original time of message transmission (always expressed in UTC (Universal Time Coordinated, also known as "GMT") when transmitting orders as the result of a resend request.

12.2. Standard Message Trailer

Tag name	Tag number	Required	Description
CheckSum	10	Y	3 bytes FIX checksum. Always the last tag of a message.

12.3. ResendRequest (Type 2)

Tag name	Tag number	Required	Description
BeginSeqNo	7	Y	Sequence number of the first message to be resent.
EndSeqNo	16	Y	Sequence number of last message to be resent, or 0 (zero) to send all messages subsequent to the first message.

12.4. SequenceReset (Type 4)

Tag name	Tag number	Required	Description
GapFillFlag	123	N	Set “Y” for Gap Fill mode or “N” for reset mode.
NewSeqNo	36	Y	New sequence number.

12.5. Reject (Type 3)

Tag name	Tag number	Required	Description
RefSeqNum	45	Y	The rejected message sequence number.
RefTagId	371	N	The tag number of the FIX field being referenced.
RefMsgType	372	N	The rejected message type.
SessionRejectReason	373	N	Reject code.
Text	58	N	Text description of the rejection.

12.6. TestRequest Message Definition (Type 1)

Tag name	Tag number	Required	Description
TestReqID	112	Y	Unique ID for this message.

12.7. Heartbeat Message Definition (Type 0)

Tag name	Tag number	Required	Description
TestReqID	112	N	Unique ID for the original TestRequest message.

13) Status and Error Message

Below is the list of status messages and descriptions that the FSS Trading Platform will send to the Client system in various FIX messages.

13.1. Streaming Error

Status Message	Description
Symbol not assigned to client	The client is not allowed to trade with the requested currency pair.
No profile defined	The client has no account profile yet.
No account activated for this currency	The client has no account activated for the requested currency pair.

13.2. Trading Error

Status Message	Description
Invalid QuoteID	The requested QuoteID is invalid.
Price is mandatory	The price is missing. It must be mentioned in the order.
Quantity is mandatory	The quantity is missing. It must be mentioned in the order.

Quote reference is mandatory	The quote reference is missing. It must be mentioned in the order.
Order Quantity is lower than the minimum quantity	The requested quantity is lower than the minimum tradable quantity for the requested entry.
Order Quantity is higher than the maximum quantity	The requested quantity is higher than the maximum tradable quantity for the requested entry.
Not the good price	The requested price does not correspond to the price for the requested entry.

Appendix I. QuickFIX/J code sample

The following is an example program using QuickFIX/J API (http://www.quickfixj.org/), a widely used Java open source implementation of the FIX protocol. It is composed of three files:

1) **FSSFIXSample.java**, the Java class implementing a FIX client that follows this scenario below:
   a. Logon to both streaming and trading sessions
   b. On logon response, send a SecurityListRequest and check for EUR/USD
   c. If EUR/USD is found, subscribe to this currency pair in FullAmount (by default, PassThrough code is also shown)
   d. Send a NewOrderSingle hitting the first MarketDataEntry of each received MarkerDataSnapshot.
2) **fixsample.cfg**, the QuickFIX/J configuration file defining the connectivity and the parameters of the FIX sessions (streaming and trading)
3) **FSS_FIX44.xml**, the FIX dictionary that specifies the message types and fields supported by the current *Rules of Engagement* and that is used by the QuickFIX/J engine. *The latest version of this document is available on demand*, please contact support@fxspotstream.com .

FSSFIXSample.java

```
/*============================================================================
 *
 * Copyright (c) FXSpotStream. All Rights Reserved.
 *
 *============================================================================*/

import java.io.InputStream;
import java.util.*;
import java.util.concurrent.CountDownLatch;

import org.quickfixj.jmx.JmxExporter;
import org.slf4j.*;

import quickfix.*;
import quickfix.field.*;

/**
 * FSSFixSample
 *
 */
public class FSSFixSample extends MessageCracker implements Application {

    private final Logger log = LoggerFactory.getLogger(FSSFixSample.class);
    private static final TimeZone UTC_TIMEZONE = TimeZone.getTimeZone("UTC");
    private static final CountDownLatch shutdownLatch = new CountDownLatch(1);
    private final Initiator initiator;
    private final MessageStoreFactory messageStoreFactory;
    private final LogFactory logFactory;
    private final MessageFactory messageFactory;
    private boolean initiatorStarted;

    private SessionID trd;
    private SessionID str;

    private final String ccyPair;
    private int nosCounter = 0;
    private boolean symbolFound = false;
    private final FSSOption opt;
```

```
private enum FSSOption {
    PASSTHROUGH, FULLAMOUNT
}

public FSSFixSample(SessionSettings settings, String ccyPair, FSSOption opt)
                                                                    throws Exception {
    this.ccyPair = ccyPair;
    this.opt = opt;
    messageStoreFactory = new FileStoreFactory(settings);
    logFactory = new SLF4JLogFactory(settings);
    messageFactory = new DefaultMessageFactory();
    initiator = new SocketInitiator(this, messageStoreFactory, settings,
                                    logFactory, messageFactory);
    final JmxExporter exporter = new JmxExporter();
    exporter.register(initiator);
}

public synchronized void logon() {
    if (!initiatorStarted) {
        try {
            initiator.start();
            initiatorStarted = true;
        } catch (final Exception e) {
            log.error("Logon failed", e);
        }
    } else {
        final Iterator<SessionID> sessionIds = initiator.getSessions().iterator();
        while (sessionIds.hasNext()) {
            final SessionID sessionId = sessionIds.next();
            Session.lookupSession(sessionId).logon();
        }
    }
}

public void sendSecurityListRequest() throws SessionNotFound {
    final SecurityListRequest slr = new SecurityListRequest();
    slr.set(new SecurityReqID("SAMPLE_SLR"));
    slr.set(new SecurityListRequestType(SecurityListRequestType.ALL_SECURITIES));
    Session.sendToTarget(slr, str);
}

public void subscribeToMarketDataInPassThroughMode() throws SessionNotFound {
    final MarketDataRequest mdr = new MarketDataRequest();
    mdr.set(new MDReqID(ccyPair + "_PASSTHROUGH"));
    mdr.set(new SubscriptionRequestType(
                                        SubscriptionRequestType.SNAPSHOT_PLUS_UPDATES));
    final MarketDataRequest.NoRelatedSym symbol = new MarketDataRequest.NoRelatedSym();
    symbol.set(new Symbol(ccyPair));
    mdr.addGroup(symbol);
    mdr.set(new MarketDepth(0)); // Full depth book
    mdr.set(new MDUpdateType(MDUpdateType.FULL_REFRESH));
    Session.sendToTarget(mdr, str);
}

public void subscribeToMarketDataInFullAmountMode() throws SessionNotFound {
    final MarketDataRequest mdr = new MarketDataRequest();
    mdr.set(new MDReqID(ccyPair + "_FULLAMOUNT"));
    mdr.set(new SubscriptionRequestType(
                                        SubscriptionRequestType.SNAPSHOT_PLUS_UPDATES));
    final MarketDataRequest.NoRelatedSym symbol = new MarketDataRequest.NoRelatedSym();
    symbol.set(new Symbol(ccyPair));
    mdr.addGroup(symbol);
    mdr.set(new MDUpdateType(MDUpdateType.FULL_REFRESH));
    // FSS custom fids for FullAmount
    final int[] quantities = { 1000000, 3000000, 5000000, 10000000 };
```

```
        mdr.setInt(9000, quantities.length); // NoRequestedSize (tag 9000)
        for (final int quantity : quantities) {
            final Group group = new Group(9000, 9001);
            group.setInt(9001, quantity); // RequestedSize (tag 9001)
            mdr.addGroup(group);
        }
        Session.sendToTarget(mdr, str);
    }

    public void sendNewOrderSingle(Symbol symbol, Side side, OrderQty orderQty,
                                   Price price, String mdEntryID) throws SessionNotFound {
        final NewOrderSingle nos = new NewOrderSingle();
        final Date time = Calendar.getInstance(UTC_TIMEZONE).getTime();
        nos.set(new ClOrdID(ccyPair + "_" + nosCounter++));
        nos.set(symbol);
        nos.set(side);
        nos.set(new TransactTime(time));
        nos.set(orderQty);
        nos.set(new OrdType(OrdType.PREVIOUSLY_QUOTED));
        nos.set(price);
        nos.setString(MDEntryID.FIELD, mdEntryID); // MDEntryID (tag 278)
        nos.set(new TimeInForce(TimeInForce.FILL_OR_KILL));
        Session.sendToTarget(nos, trd);
    }

    /*
     * Application methods
     */
    @Override
    public void fromAdmin(Message arg0, SessionID arg1)
            throws FieldNotFound, IncorrectDataFormat, IncorrectTagValue, RejectLogon {
    }

    @Override
    public void fromApp(Message message, SessionID sessionID)
            throws FieldNotFound, IncorrectDataFormat, IncorrectTagValue,
                   UnsupportedMessageType {
        try {
            crack(message, sessionID);
        } catch (final Exception e) {
            log.error("Problem on session " + sessionID + " with message " + message, e);
        }
    }

    @Override
    public void onCreate(SessionID arg0) {
    }

    @Override
    public void onLogon(SessionID sessionId) {
        if (sessionId.getSenderCompID().startsWith("trd")) {
            trd = sessionId;
        } else if (sessionId.getSenderCompID().startsWith("str")) {
            str = sessionId;
            try {
                sendSecurityListRequest();
            } catch (final SessionNotFound e) {
                log.error("Could not subscribe to market data.", e);
            }
        }
    }

    @Override
    public void onLogout(SessionID arg0) {
    }
```

```
@Override
public void toAdmin(Message arg0, SessionID arg1) {
}

@Override
public void toApp(Message message, SessionID sessionID) throws DoNotSend {
}

/*
 * MessageCracker methods
 */
@Override
public void onMessage(SecurityList message, SessionID sessionID) {
    try {
        final List<Group> relatedSymbols = message.getGroups(NoRelatedSym.FIELD);
        for (final Group symbolGrp : relatedSymbols) {
            if (ccyPair.equals(symbolGrp.getString(Symbol.FIELD))) {
                log.info("Symbol " + ccyPair + " has been found.");
                symbolFound = true;
                if (FSSOption.PASSTHROUGH == opt) {
                    subscribeToMarketDataInPassThroughMode();
                } else if (FSSOption.FULLAMOUNT == opt) {
                    subscribeToMarketDataInFullAmountMode();
                }
                return;
            }
        }
        final LastFragment lf = new LastFragment();
        if (message.isSet(lf)) {
            try {
                message.get(lf);
            } catch (final FieldNotFound e) {
                log.error("Could not found LastFragment", e);
            }
            if (lf.valueEquals(true) && !symbolFound) {
                log.error("Symbol " + ccyPair + " has not been found.");
            }
        }
    } catch (final Exception e) {
        log.error("Problem with SecurityList.", e);
    }
}

@Override
public void onMessage(MarketDataSnapshotFullRefresh message, SessionID sessionID) {
    try {
        final Symbol symbol = new Symbol();
        message.get(symbol);
        final NoMDEntries noMDEntries = new NoMDEntries();
        message.get(noMDEntries);
        if (noMDEntries.getValue() > 0) {
            final MarketDataSnapshotFullRefresh.NoMDEntries mdEntry
                                = new MarketDataSnapshotFullRefresh.NoMDEntries();
            message.getGroup(1, mdEntry);
            final MDEntryType mdEntryType = new MDEntryType();
            mdEntry.get(mdEntryType);
            Side side = null;
            if (MDEntryType.BID == mdEntryType.getValue()) {
                side = new Side(Side.SELL);
            } else if (MDEntryType.OFFER == mdEntryType.getValue()) {
                side = new Side(Side.BUY);
            }
            // MDEntryID (tag 278)
            final String mdEntryID = mdEntry.getString(MDEntryID.FIELD);
            final MDEntryPx mdEntryPx = new MDEntryPx();
            final MDEntrySize mdEntrySize = new MDEntrySize();
```

```
            final OrderQty orderQty = new OrderQty(mdEntrySize.getValue());
            final Price price = new Price(mdEntryPx.getValue());
            sendNewOrderSingle(symbol, side, orderQty, price, mdEntryID);
        }
    } catch (final Exception e) {
        log.error("Problem with MarketDataSnapshotFullRefresh.", e);
    }
}

private String sideToString(Side side) {
    switch (side.getValue()) {
        case Side.BUY:
            return "BUY";

        case Side.SELL:
            return "SELL";

        default:
            return "";
    }
}

@Override
public void onMessage(ExecutionReport message, SessionID sessionID) {
    // process the ExecutionReport
    final ExecType execType = new ExecType();
    try {
        message.get(execType);
        if (ExecType.FILL == execType.getValue()) {
            final ClOrdID clOrdID = new ClOrdID();
            message.get(clOrdID);
            final OrderID fssOrdID = new OrderID();
            message.get(fssOrdID);
            final SecondaryOrderID bankOrdID = new SecondaryOrderID();
            message.get(bankOrdID);
            final ExecID fssExecID = new ExecID();
            message.get(fssExecID);
            final SecondaryExecID bankExecID = new SecondaryExecID();
            message.get(bankExecID);
            final Side side = new Side();
            message.get(side);
            log.info(sideToString(side) + " order " + clOrdID.getValue()
                 + " has been filled [bankOrdID=" + bankOrdID.getValue()
                 + ", bankExecID=" + bankExecID.getValue() + ", FSSOrdID="
                 + fssOrdID.getValue() + ", FSSExecID="
                 + fssExecID.getValue() + "]");
        } else if (ExecType.REJECTED == execType.getValue()) {
            final ClOrdID clOrdID = new ClOrdID();
            message.get(clOrdID);
            final Side side = new Side();
            message.get(side);
            final Text reason = new Text();
            message.get(reason);
            log.warn(sideToString(side) + " order " + clOrdID.getValue()
                 + " has been rejected (" + reason.getValue() + ").");
        } else if (ExecType.CANCELED == execType.getValue()) {
            final ClOrdID clOrdID = new ClOrdID();
            message.get(clOrdID);
            final Side side = new Side();
            message.get(side);
            final Text reason = new Text();
            message.get(reason);
            log.warn(sideToString(side) + " order " + clOrdID.getValue()
                 + " has been canceled (" + reason.getValue() + ").");
        }
    } catch (final FieldNotFound e) {
```

```
            log.error("Could not process executionReport " + message, e);
        }
    }

    /*
     * MAIN
     */
    public static void main(String[] args) throws Exception {
        ClassLoader classLoader = FSSFixSample.class.getClassLoader();
        final InputStream inputStream = classLoader.getResourceAsStream("fixsample.cfg");
        final String ccyPair = "EUR/USD";
        final FSSOption opt = FSSOption.FULLAMOUNT;
        final SessionSettings settings = new SessionSettings(inputStream);
        inputStream.close();

        final FSSFixSample app = new FSSFixSample(settings, ccyPair, opt);
        app.logon();
        shutdownLatch.await();

    }
}
```

fixsample.cfg

```
[DEFAULT]
BeginString=FIX.4.4
StartTime=00:00:00
EndTime=00:00:00
CheckLatency=N
HeartBtInt=6000
DataDictionary=FSS_FIX44.xml
UseDataDictionary=Y
ReconnectInterval=3
FileStorePath=../var/store/fixsample/
FileLogPath=../var/log/
FileStoreSync=N

[SESSION]
ConnectionType=initiator
SocketConnectPort=9110
SocketConnectHost=127.0.0.1
SenderCompID=trd.ny.sim.fsstest.1
TargetCompID=FSS
ResetOnLogon=N

[SESSION]
ConnectionType=initiator
SocketConnectPort=9100
SocketConnectHost=127.0.0.1
SenderCompID=str.ny.sim.fsstest.1
TargetCompID=FSS
ResetOnLogon=Y
```

FSS.xml

```
<fix major="4" minor="4">
	<header>
		<field name="BeginString" required="Y" />
		<field name="BodyLength" required="Y" />
		<field name="MsgType" required="Y" />
		<field name="SenderCompID" required="Y" />
		<field name="TargetCompID" required="Y" />
		<field name="MsgSeqNum" required="Y" />
		<field name="PossDupFlag" required="N" />
		<field name="PossResend" required="N" />
		<field name="SendingTime" required="Y" />
		<field name="OrigSendingTime" required="N" />
	</header>
	<trailer>
		<field name="CheckSum" required="Y" />
	</trailer>
	<messages>
		<message name="Heartbeat" msgtype="0" msgcat="admin">
			<field name="TestReqID" required="N" />
		</message>
		<message name="Logon" msgtype="A" msgcat="admin">
			<field name="EncryptMethod" required="Y" />
			<field name="HeartBtInt" required="Y" />
			<field name="ResetSeqNumFlag" required="N" />
			<field name="Password" required="N" />
		</message>
		<message name="TestRequest" msgtype="1" msgcat="admin">
			<field name="TestReqID" required="Y" />
		</message>
		<message name="ResendRequest" msgtype="2" msgcat="admin">
			<field name="BeginSeqNo" required="Y" />
			<field name="EndSeqNo" required="Y" />
		</message>
		<message name="Reject" msgtype="3" msgcat="admin">
			<field name="RefSeqNum" required="Y" />
			<field name="RefTagID" required="N" />
			<field name="RefMsgType" required="N" />
			<field name="SessionRejectReason" required="N" />
			<field name="Text" required="N" />
		</message>
		<message name="SequenceReset" msgtype="4" msgcat="admin">
			<field name="GapFillFlag" required="N" />
			<field name="NewSeqNo" required="Y" />
		</message>
		<message name="Logout" msgtype="5" msgcat="admin">
			<field name="Text" required="N" />
		</message>
		<message name="BusinessMessageReject" msgtype="j" msgcat="app">
			<field name="RefSeqNum" required="N" />
			<field name="RefMsgType" required="Y" />
			<field name="BusinessRejectRefID" required="N" />
			<field name="BusinessRejectReason" required="Y" />
			<field name="Text" required="N" />
			<field name="EncodedTextLen" required="N" />
			<field name="EncodedText" required="N" />
		</message>
		<message name="MarketDataRequest" msgtype="V" msgcat="app">
			<field name="MDReqID" required="Y" />
			<group name="NoPartyIDs" required="N">
				<field name="PartyID" required="Y" />
				<field name="PartyIDSource" required="Y" />
				<field name="PartyRole" required="Y" />
			</group>
			<field name="Symbol" required="Y" />
			<field name="SubscriptionRequestType" required="Y" />
			<field name="MarketDepth" required="N" />
			<field name="MDUpdateType" required="N" />
			<group name="NoRequestedSize" required="N">
				<field name="RequestedSize" required="N" />
			</group>
			<field name="OrdType" required="N"/>
		</message>
```

```
<message name="MarketDataSnapshotFullRefresh" msgtype="W" msgcat="app">
        <field name="MDReqID" required="N" />
        <field name="Symbol" required="Y" />
        <field name="SettlDate" required="N" />
        <group name="NoMDEntries" required="Y">
                <field name="MDEntryType" required="Y" />
                <field name="MDEntryRefID" required="N" />
                <field name="MDEntryPositionNo" required="N" />
                <field name="MDEntryID" required="N" />
                <field name="MDEntryPx" required="Y" />
                <field name="MDEntrySize" required="N" />
                <field name="MDEntryDate" required="N" />
                <field name="MDEntryTime" required="N" />
                <field name="Currency" required="N" />
                <field name="MinQty" required="N" />
                <field name="MDEntryOriginator" required="Y" />
                <field name="SettlDate" required="N" />
                <field name="MDEntryOrigTime" required="N" />
                <field name="MDEntrySpotRate" required="N" />
        </group>
</message>
<message name="MarketDataIncrementalRefresh" msgtype="X" msgcat="app">
        <field name="MDReqID" required="Y" />
        <field name="Symbol" required="Y" />
        <field name="SettlDate" required="N" />
        <group name="NoMDEntries" required="Y">
                <field name="MDEntryType" required="Y" />
                <field name="MDUpdateAction" required="Y" />
                <field name="MDEntryRefID" required="N" />
                <field name="MDEntryPositionNo" required="N" />
                <field name="MDEntryID" required="N" />
                <field name="MDEntryPx" required="N" />
                <field name="MDEntrySize" required="N" />
                <field name="MDEntryDate" required="N" />
                <field name="MDEntryTime" required="N" />
                <field name="Currency" required="N" />
                <field name="MinQty" required="N" />
                <field name="MDEntryOriginator" required="N" />
                <field name="SettlDate" required="N" />
                <field name="MDEntryOrigTime" required="N" />
                <field name="MDEntrySpotRate" required="N" />
        </group>
</message>
<message name="MarketDataRequestReject" msgtype="Y" msgcat="app">
        <field name="MDReqID" required="Y" />
        <field name="MDReqRejReason" required="N" />
        <field name="Text" required="N" />
</message>
<message name="SecurityListRequest" msgtype="x" msgcat="app">
        <field name="SecurityReqID" required="Y" />
        <field name="SecurityListRequestType" required="Y" />
        <field name="MaturityDate" required="N" />
</message>
<message name="SecurityList" msgtype="y" msgcat="app">
        <field name="SecurityReqID" required="Y" />
        <field name="SecurityRequestResult" required="Y" />
        <field name="Text" required="N" />
        <field name="LastFragment" required="N" />
        <group name="NoRelatedSym" required="Y">
                <field name="Symbol" required="Y" />
                <field name="SecurityType" required="Y" />
                <field name="SettlDate" required="N" />
        </group>
</message>
<message name="NewOrderSingle" msgtype="D" msgcat="app">
        <field name="Account" required="N" />
        <field name="ClOrdID" required="Y" />
        <group name="NoPartyIDs" required="N">
                <field name="PartyID" required="N" />
                <field name="PartyIDSource" required="N" />
                <field name="PartyRole" required="N" />
        </group>
        <field name="Symbol" required="Y" />
        <field name="Side" required="Y" />
```

```
        <field name="TransactTime" required="Y" />
        <field name="OrderQty" required="Y" />
        <field name="OrdType" required="Y" />
        <field name="Price" required="Y" />
        <field name="PriceImprovement" required="N" />
        <field name="Currency" required="Y" />
        <field name="MDEntryID" required="N" />
        <field name="TimeInForce" required="Y" />
        <field name="SettlDate" required="N" />
        <field name="MDEntrySpotRate" required="N" />
        <field name="TargetStrategy" required="N" />
        <field name="ExpireTime" required="N" />
</message>

<message name="ExecutionReport" msgtype="8" msgcat="app">
        <field name="Account" required="N" />
        <field name="OrderID" required="N" />
        <field name="SecondaryOrderID" required="N" />
        <field name="ClOrdID" required="N" />
        <field name="OrigClOrdID" required="N" />
        <group name="NoPartyIDs" required="N">
                <field name="PartyID" required="N" />
                <field name="PartyIDSource" required="N" />
                <field name="PartyRole" required="N" />
        </group>
        <field name="ExecID" required="N" />
        <field name="SecondaryExecID" required="N" />
        <field name="ExecType" required="Y" />
        <field name="OrdStatus" required="Y" />
        <field name="Symbol" required="Y" />
        <field name="Side" required="Y" />
        <field name="OrderQty" required="Y" />
        <field name="OrdType" required="N" />
        <field name="Price" required="N" />
        <field name="Currency" required="Y" />
        <field name="TimeInForce" required="Y" />
        <field name="LastQty" required="Y" />
        <field name="LastPx" required="Y" />
        <field name="LeavesQty" required="Y" />
        <field name="CumQty" required="Y" />
        <field name="AvgPx" required="Y" />
        <field name="TransactTime" required="N" />
        <field name="Text" required="N" />
        <field name="TradeDate" required="N" />
        <field name="OrdRejReason" required="N" />
        <field name="MDEntryID" required="N" />
        <field name="LastMkt" required="N" />

        <field name="TargetStrategy" required="N" />

        <field name="ReferenceEquivalentQty" required="N" />
        <field name="ReferenceEquivalentQty2" required="N" />

        <field name="QuoteRespID" required="N" />
        <field name="QuoteID" required="N" />
        <field name="QuoteReqID" required="N" />
        <field name="SettlType" required="N" />
        <field name="SettlType2" required="N" />
        <field name="SettlDate" required="N" />
        <field name="SettlDate2" required="N" />
        <field name="LastSpotRate" required="N" />
        <field name="LastSpotRate2" required="N" />
        <field name="OrderQty2" required="N" />
        <field name="Price2" required="N" />
        <field name="LastForwardPoints" required="N" />
        <field name="LastForwardPoints2" required="N" />
        <field name="LastQty2" required="N" />
        <field name="LastPx2" required="N" />
        <field name="LeavesQty2" required="N" />
        <field name="CumQty2" required="N" />
        <field name="AvgPx2" required="N" />

        <!-- UTI -->
        <field name="UTIPrefix" required="N" />
```

```
        <field name="UTI" required="N" />
        <field name="UTIPrefix2" required="N" />
        <field name="UTI2" required="N" />
</message>


<message name="QuoteRequest" msgtype="R" msgcat="app">
        <field name="QuoteReqID" required="Y" />
        <field name="Account" required="N" />
        <field name="ExpireTime" required="N" />
        <group name="NoPartyIDs" required="N">
                <field name="PartyID" required="N" />
                <field name="PartyIDSource" required="N" />
                <field name="PartyRole" required="N" />
        </group>
        <group name="NoRelatedSym" required="Y">
                <field name="Symbol" required="Y" />
                <field name="Side" required="N" />
                <field name="OrderQty" required="Y" />
                <field name="SettlType" required="Y" />
                <field name="SettlType2" required="N" />
                <field name="OrderQty2" required="N" />
                <field name="Currency" required="N" />
        </group>
</message>


<message name="MassQuote" msgtype="i" msgcat="app">

        <field name="QuoteReqID" required="Y" />
        <field name="Symbol" required="Y" />
        <field name="SettlType" required="Y" />
        <field name="SettlType2" required="N" />

        <group name="NoQuoteEntries" required="Y">
                <field name="QuoteEntryID" required="Y" />
                <field name="Side" required="N" />
                <field name="Currency" required="N" />

                <field name="BidSpotRate" required="N" />
                <field name="OfferSpotRate" required="N" />

                <field name="BidSpotRate2" required="N" />
                <field name="OfferSpotRate2" required="N" />

                <field name="BidSwapPoints" required="N" />
                <field name="OfferSwapPoints" required="N" />

                <field name="BidPx" required="N" />
                <field name="BidSize" required="N" />
                <field name="BidForwardPoints" required="N" />

                <field name="OfferPx" required="N" />
                <field name="OfferSize" required="N" />
                <field name="OfferForwardPoints" required="N" />

                <!-- following fields for swap -->
                <field name="BidPx2" required="N" />
                <field name="BidSize2" required="N" />
                <field name="BidForwardPoints2" required="N" />

                <field name="OfferPx2" required="N" />
                <field name="OfferSize2" required="N" />
                <field name="OfferForwardPoints2" required="N" />
                <!-- end of fields for swap -->

                <!-- LP specific expiration time -->
                <field name="ValidUntilTime" required="N" />
                <field name="MDEntryDate" required="N" />
                <field name="MDEntryTime" required="N" />
                <!-- LP id -->
                <field name="MDEntryOriginator" required="N" />

                <field name="SettlDate" required="N" />
```

```
                <field name="SettlDate2" required="N" />
        </group>
        <group name="NoMDEntries" required="Y">
                <field name="MDEntryType" required="Y" />
                <field name="MDEntryOriginator" required="Y" />
                <field name="MidPx" required="N" />
                <field name="FarMidPx" required="N" />
        </group>

</message>

<message name="QuoteResponse" msgtype="AJ" msgcat="app">
        <field name="QuoteRespID" required="Y" />
        <field name="QuoteID" required="Y" />

        <field name="QuoteRespType" required="Y" />
        <field name="Symbol" required="Y" />
        <field name="QuoteReqID" required="Y" />
        <field name="Side" required="Y" />

        <field name="SettlType" required="N" />
        <field name="SettlDate" required="N" />
        <field name="SettlType2" required="N" />
        <field name="SettlDate2" required="N" />
        <field name="Currency" required="N" />

        <field name="BidSpotRate" required="N" />
        <field name="OfferSpotRate" required="N" />

        <field name="BidSpotRate2" required="N" />
        <field name="OfferSpotRate2" required="N" />

        <field name="BidSize" required="N" />
        <field name="BidPx" required="N" />
        <field name="BidForwardPoints" required="N" />

        <field name="OfferSize" required="N" />
        <field name="OfferPx" required="N" />
        <field name="OfferForwardPoints" required="N" />

        <!-- following fields for swap -->
        <field name="BidSize2" required="N" />
        <field name="BidPx2" required="N" />
        <field name="BidForwardPoints2" required="N" />

        <field name="OfferSize2" required="N" />
        <field name="OfferPx2" required="N" />
        <field name="OfferForwardPoints2" required="N" />
        <!-- end of fields for swap -->

        <field name="TransactTime" required="Y" />

        <!-- UTI -->
        <field name="UTIPrefix" required="N" />
        <field name="UTI" required="N" />
        <field name="UTIPrefix2" required="N" />
        <field name="UTI2" required="N" />
</message>

<message name="MassQuoteAcknowledgement" msgtype="b" msgcat="app">
        <field name="QuoteReqID" required="Y" />
        <field name="QuoteStatus" required="N" />
        <field name="QuoteRejectReason" required="N" />
        <field name="Text" required="N" />
        <field name="Symbol" required="N" />
</message>


<message name="QuoteCancel" msgtype="Z" msgcat="app">
        <field name="QuoteReqID" required="Y" />
        <field name="QuoteRespID" required="Y" />
        <field name="Symbol" required="N" />
</message>
```

```
        <message name="OrderCancelRequest" msgtype="F" msgcat="app">
                <field name="ClOrdID" required="Y" />
                <field name="OrigClOrdID" required="Y" />
                <field name="Symbol" required="Y" />
                <field name="TransactTime" required="Y" />
        </message>

        <message name="OrderCancelReject" msgtype="9" msgcat="app">
                <field name="OrderID" required="N" />
                <field name="ClOrdID" required="Y" />
                <field name="OrigClOrdID" required="Y" />
                <field name="Symbol" required="Y" />
                <field name="OrdStatus" required="N" />
                <field name="CxlRejResponseTo" required="Y" />
                <field name="CxlRejReason" required="N" />
                <field name="Text" required="N" />
        </message>

<message name="TradingSessionListRequest" msgtype="BI" msgcat="app">
    <field name="TradSesReqID" required="Y" />
    <field name="SubscriptionRequestType" required="Y" />
</message>

<message name="TradingSessionList" msgtype="BJ" msgcat="app">
    <field name="TradSesReqID" required="Y" />
    <group name="NoTradingSessions" required="Y">
        <field name="TradingSessionID" required="Y" />
        <field name="TradSesStatus" required="Y" />
    </group>
</message>

<message name="TradingSessionListUpdateReport" msgtype="BS" msgcat="app">
    <field name="TradSesReqID" required="Y" />
                <field name="Text" required="Y" />
</message>
</messages>
<fields>
        <field number="1" name="Account" type="STRING" />
        <field number="6" name="AvgPx" type="PRICE" />
        <field number="7" name="BeginSeqNo" type="SEQNUM" />
        <field number="8" name="BeginString" type="STRING" />
        <field number="9" name="BodyLength" type="LENGTH" />
        <field number="10" name="CheckSum" type="STRING" />
        <field number="11" name="ClOrdID" type="STRING" />
        <field number="14" name="CumQty" type="QTY" />
        <field number="15" name="Currency" type="CURRENCY" />
        <field number="16" name="EndSeqNo" type="SEQNUM" />
        <field number="17" name="ExecID" type="STRING" />
        <field number="30" name="LastMkt" type="EXCHANGE" />
        <field number="31" name="LastPx" type="PRICE" />
        <field number="32" name="LastQty" type="QTY" />
        <field number="34" name="MsgSeqNum" type="SEQNUM" />
        <field number="35" name="MsgType" type="STRING">
                <value enum="0" description="HEARTBEAT" />
                <value enum="1" description="TEST_REQUEST" />
                <value enum="2" description="RESEND_REQUEST" />
                <value enum="3" description="REJECT" />
                <value enum="4" description="SEQUENCE_RESET" />
                <value enum="5" description="LOGOUT" />
                <value enum="8" description="EXECUTION_REPORT" />
                <value enum="9" description="ORDER_CANCEL_REJECT" />
                <value enum="A" description="LOGON" />
                <value enum="D" description="ORDER_SINGLE" />
                <value enum="F" description="ORDER_CANCEL_REQUEST" />
                <value enum="V" description="MARKET_DATA_REQUEST" />
                <value enum="W" description="MARKET_DATA_SNAPSHOT_FULL_REFRESH" />
                <value enum="X" description="MARKET_DATA_SNAPSHOT_INCREMENTAL_REFRESH" />
                <value enum="Y" description="MARKET_DATA_REQUEST_REJECT" />
                <value enum="j" description="BUSINESS_MESSAGE_REJECT" />
                <value enum="x" description="SECURITY_LIST_REQUEST" />
                <value enum="y" description="SECURITY_LIST" />
                <value enum="AJ" description="QUOTE_RESPONSE" />
                <value enum="i" description="MASS_QUOTE" />
                <value enum="R" description="QUOTE_REQUEST" />
```

```
            <value enum="b" description="MASS_QUOTE_ACKNOWLEDGEMENT" />
            <value enum="Z" description="CANCEL_QUOTE" />
            <value enum="BI" description="TRADING_SESSION_LIST_REQUEST" />
            <value enum="BJ" description="TRADING_SESSION_LIST" />
            <value enum="BS" description="TRADING_SESSION_LIST_UPDATE_REPORT" />
    </field>
    <field number="36" name="NewSeqNo" type="SEQNUM" />
    <field number="37" name="OrderID" type="STRING" />
    <field number="38" name="OrderQty" type="QTY" />
    <field number="39" name="OrdStatus" type="CHAR">
            <value enum="1" description="PARTIALLY_FILLED" />
            <value enum="2" description="FILLED" />
            <value enum="4" description="CANCELED" />
            <value enum="8" description="REJECTED" />
            <value enum="A" description="PENDING_NEW" />
            <value enum="C" description="EXPIRED" />
    </field>
    <field number="40" name="OrdType" type="CHAR">
            <value enum="2" description="LIMIT" />
            <value enum="D" description="PREVIOUSLY_QUOTED" />
    </field>
    <field number="41" name="OrigClOrdID" type="STRING" />
    <field number="43" name="PossDupFlag" type="BOOLEAN" />
    <field number="44" name="Price" type="PRICE" />
    <field number="45" name="RefSeqNum" type="SEQNUM" />
    <field number="49" name="SenderCompID" type="STRING" />
    <field number="52" name="SendingTime" type="UTCTIMESTAMP" />
    <field number="54" name="Side" type="CHAR">
            <value enum="1" description="BUY" />
            <value enum="2" description="SELL" />
    </field>
    <field number="55" name="Symbol" type="STRING" />
    <field number="56" name="TargetCompID" type="STRING" />
    <field number="58" name="Text" type="STRING" />
    <field number="59" name="TimeInForce" type="CHAR">
            <value enum="1" description="GOOD_TILL_CANCEL" />
            <value enum="3" description="IMMEDIATE_OR_CANCEL" />
            <value enum="4" description="FILL_OR_KILL" />
            <value enum="6" description="GOOD_TILL_DATE" />
    </field>
    <field number="60" name="TransactTime" type="UTCTIMESTAMP" />

    <field number="62" name="ValidUntilTime" type="UTCTIMESTAMP" />
    <field number="63" name="SettlType" type="STRING">
            <value enum="ONI" description="ONI"></value>
            <value enum="TOD" description="TOD"></value>
            <value enum="TOM" description="TOM"></value>
            <value enum="TNX" description="TNX"></value>
            <value enum="SP" description="SP"></value>
            <value enum="SNX" description="SNX"></value>
            <value enum="D2" description="D2"></value>
            <value enum="D3" description="D3"></value>
            <value enum="D4" description="D4"></value>
            <value enum="W1" description="W1"></value>
            <value enum="W2" description="W2"></value>
            <value enum="W3" description="W3"></value>
            <value enum="M1" description="M1"></value>
            <value enum="M2" description="M2"></value>
            <value enum="M3" description="M3"></value>
            <value enum="M4" description="M4"></value>
            <value enum="M5" description="M5"></value>
            <value enum="M6" description="M6"></value>
            <value enum="M7" description="M7"></value>
            <value enum="M8" description="M8"></value>
            <value enum="M9" description="M9"></value>
            <value enum="M10" description="M10"></value>
            <value enum="M11" description="M11"></value>
            <value enum="Y1" description="Y1"></value>
            <value enum="M15" description="M15"></value>
            <value enum="M18" description="M18"></value>
            <value enum="M21" description="M21"></value>
            <value enum="Y2" description="Y2"></value>
            <value enum="Y3" description="Y3"></value>
            <value enum="Y4" description="Y4"></value>
```

```
                        <value enum="Y5" description="Y5"></value>
                        <value enum="Y6" description="Y6"></value>
                        <value enum="Y7" description="Y7"></value>
                        <value enum="Y8" description="Y8"></value>
                        <value enum="Y9" description="Y9"></value>
                        <value enum="Y10" description="Y10"></value>
                        <value enum="Y15" description="Y15"></value>
                        <value enum="Y20" description="Y20"></value>
                        <value enum="Y25" description="Y25"></value>
                        <value enum="Y30" description="Y30"></value>
                        <value enum="IM1" description="IMM1"></value>
                        <value enum="IM2" description="IMM2"></value>
                        <value enum="IM3" description="IMM3"></value>
                        <value enum="IM4" description="IMM4"></value>
                        <value enum="B" description="B"></value>
                </field>

                <field number="64" name="SettlDate" type="LOCALMKTDATE" />
                <field number="75" name="TradeDate" type="LOCALMKTDATE" />
                <field number="97" name="PossResend" type="BOOLEAN" />
                <field number="98" name="EncryptMethod" type="INT">
                        <value enum="0" description="NONE_OTHER" />
                </field>
                <field number="102" name="CxlRejReason" type="INT">
                        <value enum="0" description="TOO_LATE_TO_CANCEL" />
                        <value enum="1" description="UNKNOWN_ORDER" />
                        <value enum="2" description="BROKER_EXCHANGE_OPTION" />
                        <value enum="3"
description="ORDER_ALREADY_IN_PENDING_CANCEL_OR_PENDING_REPLACE_STATUS" />
                        <value enum="4" description="UNABLE_TO_PROCESS_ORDER_MASS_CANCEL_REQUEST" />
                        <value enum="5"
description="ORIGORDMODTIME_DID_NOT_MATCH_LAST_TRANSACTTIME_OF_ORDER" />
                        <value enum="6" description="DUPLICATE_CLORDID_RECEIVED" />
                        <value enum="99" description="OTHER" />
                </field>
                <field number="103" name="OrdRejReason" type="INT">
                        <value enum="0" description="BROKER_EXCHANGE_OPTION" />
                        <value enum="1" description="UNKNOWN_SYMBOL" />
                        <value enum="2" description="EXCHANGE_CLOSED" />
                        <value enum="3" description="ORDER_EXCEEDS_LIMIT" />
                        <value enum="4" description="TOO_LATE_TO_ENTER" />
                        <value enum="5" description="UNKNOWN_ORDER" />
                        <value enum="6" description="DUPLICATE_ORDER" />
                        <value enum="7" description="DUPLICATE_OF_A_VERBALLY_COMMUNICATED_ORDER" />
                        <value enum="8" description="STALE_ORDER" />
                        <value enum="9" description="TRADE_ALONG_REQUIRED" />
                        <value enum="10" description="INVALID_INVESTOR_ID" />
                        <value enum="11" description="UNSUPPORTED_ORDER_CHARACTERISTIC" />
                        <value enum="12" description="SURVEILLENCE_OPTION" />
                        <value enum="13" description="INCORRECT_QUANTITY" />
                        <value enum="14" description="INCORRECT_ALLOCATED_QUANTITY" />
                        <value enum="15" description="UNKNOWN_ACCOUNT" />
                        <value enum="99" description="OTHER" />
                </field>
                <field number="108" name="HeartBtInt" type="INT" />
                <field number="110" name="MinQty" type="QTY" />
                <field number="112" name="TestReqID" type="STRING" />
                <field number="117" name="QuoteID" type="STRING" />
                <field number="122" name="OrigSendingTime" type="UTCTIMESTAMP" />
                <field number="123" name="GapFillFlag" type="BOOLEAN" />
                <field number="126" name="ExpireTime" type="UTCTIMESTAMP" />
                <field number="131" name="QuoteReqID" type="STRING" />
                <field number="132" name="BidPx" type="PRICE" />
                <field number="133" name="OfferPx" type="PRICE" />
                <field number="134" name="BidSize" type="QTY" />
                <field number="135" name="OfferSize" type="QTY" />
                <field number="141" name="ResetSeqNumFlag" type="BOOLEAN" />
                <field number="146" name="NoRelatedSym" type="NUMINGROUP" />
                <field number="150" name="ExecType" type="CHAR">
                        <value enum="0" description="NEW" />
                        <value enum="1" description="PARTIAL_FILL" />
                        <value enum="2" description="FILL" />
                        <value enum="4" description="CANCELED" />
                        <value enum="8" description="REJECTED" />
```

```
        <value enum="A" description="PENDING_NEW" />
        <value enum="C" description="EXPIRED" />
</field>
<field number="151" name="LeavesQty" type="QTY" />
<field number="167" name="SecurityType" type="STRING">
        <value enum="FX" description="FOREIGN_EXCHANGE" />
        <value enum="PM" description="PRECIOUS_METALS" />
</field>

<field number="188" name="BidSpotRate" type="PRICE" />
<field number="189" name="BidForwardPoints" type="PRICEOFFSET" />
<field number="190" name="OfferSpotRate" type="PRICE" />
<field number="191" name="OfferForwardPoints" type="PRICEOFFSET" />
<field number="192" name="OrderQty2" type="QTY" />
<field number="193" name="SettlDate2" type="LOCALMKTDATE" />
<field number="194" name="LastSpotRate" type="PRICE" />
<field number="195" name="LastForwardPoints" type="PRICEOFFSET" />

<field number="198" name="SecondaryOrderID" type="STRING" />
<field number="262" name="MDReqID" type="STRING" />
<field number="263" name="SubscriptionRequestType" type="CHAR">
        <value enum="1" description="SNAPSHOT_AND_UPDATES" />
        <value enum="2" description="UNSUBSCRIBE" />
        <value enum="Z" description="NO_MARKET_FEEDBACK" />
</field>
<field number="264" name="MarketDepth" type="INT" />
<field number="265" name="MDUpdateType" type="INT">
        <value enum="0" description="FULL_REFRESH" />
        <value enum="1" description="INCREMENTAL_REFRESH" />
</field>
<field number="268" name="NoMDEntries" type="NUMINGROUP" />
<field number="269" name="MDEntryType" type="CHAR">
        <value enum="0" description="BID" />
        <value enum="1" description="OFFER" />
        <value enum="H" description="MID_PRICE" />
</field>
<field number="270" name="MDEntryPx" type="PRICE" />
<field number="271" name="MDEntrySize" type="QTY" />
<field number="272" name="MDEntryDate" type="UTCDATEONLY" />
<field number="273" name="MDEntryTime" type="UTCTIMEONLY" />
<field number="278" name="MDEntryID" type="STRING" />
<field number="279" name="MDUpdateAction" type="CHAR">
        <value enum="0" description="NEW" />
        <value enum="1" description="CHANGE" />
        <value enum="2" description="DELETE" />
</field>
<field number="280" name="MDEntryRefID" type="STRING" />
<field number="281" name="MDReqRejReason" type="CHAR">
        <value enum="0" description="UNKNOWN_SYMBOL" />
        <value enum="1" description="DUPLICATE_MDREQID" />
        <value enum="2" description="INSUFFICIENT_BANDWIDTH" />
        <value enum="3" description="INSUFFICIENT_PERMISSIONS" />
        <value enum="4" description="UNSUPPORTED_SUBSCRIPTIONREQUESTTYPE" />
        <value enum="5" description="UNSUPPORTED_MARKETDEPTH" />
        <value enum="6" description="UNSUPPORTED_MDUPDATETYPE" />
        <value enum="7" description="UNSUPPORTED_AGGREGATEDBOOK" />
        <value enum="8" description="UNSUPPORTED_MDENTRYTYPE" />
        <value enum="9" description="UNSUPPORTED_TRADINGSESSIONID" />
        <value enum="A" description="UNSUPPORTED_SCOPE" />
        <value enum="B" description="UNSUPPORTED_OPENCLOSESETTLEFLAG" />
        <value enum="C" description="UNSUPPORTED_MDIMPLICITDELETE" />
        <value enum="Z" description="TECHNICAL_INCIDENT" />
</field>
<field number="282" name="MDEntryOriginator" type="STRING" />
<field number="290" name="MDEntryPositionNo" type="INT" />
<field number="295" name="NoQuoteEntries" type="NUMINGROUP" />
<field number="297" name="QuoteStatus" type="INT">
        <value enum="0" description="ACCEPTED" />
        <value enum="4" description="CANCELED" />
        <value enum="7" description="EXPIRED" />
        <value enum="5" description="REJECTED" />
        <value enum="16" description="EXECUTED" />
</field>
<field number="299" name="QuoteEntryID" type="STRING" />
```

```
                <field number="300" name="QuoteRejectReason" type="INT">
                        <value enum="1" description="UNKNOWN_SYMBOL" />
                        <value enum="2" description="EXCHANGE_CLOSED" />
                        <value enum="3" description="QUOTE_REQUEST_EXCEEDS_LIMIT" />
                        <value enum="4" description="TOO_LATE_TO_ENTER" />
                        <value enum="5" description="UNKNOWN_QUOTE" />
                        <value enum="6" description="DUPLICATE_QUOTE" />
                        <value enum="7" description="INVALID_BID_ASK_SPREAD" />
                        <value enum="8" description="INVALID_PRICE" />
                        <value enum="9" description="NOT_AUTHORIZED_TO_QUOTE_SECURITY" />
                        <value enum="99" description="OTHER" />
                </field>
                <field number="320" name="SecurityReqID" type="STRING" />
                <field number="335" name="TradSesReqID" type="STRING" />
                <field number="336" name="TradingSessionID" type="STRING" />
        <field number="340" name="TradSesStatus" type="INT">
                        <value enum="2" description="OPEN" />
                        <value enum="7" description="SUSPICIOUS" />
                </field>
        <field number="354" name="EncodedTextLen" type="LENGTH" />
                <field number="355" name="EncodedText" type="DATA" />
                <field number="371" name="RefTagID" type="INT" />
                <field number="372" name="RefMsgType" type="STRING" />
                <field number="373" name="SessionRejectReason" type="INT">
                        <value enum="0" description="INVALID_TAG_NUMBER" />
                        <value enum="1" description="REQUIRED_TAG_MISSING" />
                        <value enum="2" description="TAG_NOT_DEFINED_FOR_THIS_MESSAGE_TYPE" />
                        <value enum="3" description="UNDEFINED_TAG" />
                        <value enum="4" description="TAG_SPECIFIED_WITHOUT_A_VALUE" />
                        <value enum="5" description="VALUE_IS_INCORRECT" />
                        <value enum="6" description="INCORRECT_DATA_FORMAT_FOR_VALUE" />
                        <value enum="7" description="DECRYPTION_PROBLEM" />
                        <value enum="8" description="SIGNATURE_PROBLEM" />
                        <value enum="9" description="COMPID_PROBLEM" />
                        <value enum="10" description="SENDINGTIME_ACCURACY_PROBLEM" />
                        <value enum="11" description="INVALID_MSGTYPE" />
                        <value enum="12" description="XML_VALIDATION_ERROR" />
                        <value enum="13" description="TAG_APPEARS_MORE_THAN_ONCE" />
                        <value enum="14" description="TAG_SPECIFIED_OUT_OF_REQUIRED_ORDER" />
                        <value enum="15" description="REPEATING_GROUP_FIELDS_OUT_OF_ORDER" />
                        <value enum="16" description="INCORRECT_NUMINGROUP_COUNT_FOR_REPEATING_GROUP"
/>
                        <value enum="17" description="NON_DATA_VALUE_INCLUDES_FIELD_DELIMITER" />
                        <value enum="99" description="OTHER" />
                </field>
                <field number="379" name="BusinessRejectRefID" type="STRING" />
                <field number="380" name="BusinessRejectReason" type="INT">
                        <value enum="0" description="OTHER" />
                        <value enum="1" description="UNKOWN_ID" />
                        <value enum="2" description="UNKNOWN_SECURITY" />
                        <value enum="3" description="UNSUPPORTED_MESSAGE_TYPE" />
                        <value enum="4" description="APPLICATION_NOT_AVAILABLE" />
                        <value enum="5" description="CONDITIONALLY_REQUIRED_FIELD_MISSING" />
                        <value enum="6" description="NOT_AUTHORIZED" />
                        <value enum="7" description="DELIVERTO_FIRM_NOT_AVAILABLE_AT_THIS_TIME" />
                </field>
                <field number="434" name="CxlRejResponseTo" type="CHAR">
                        <value enum="1" description="ORDER_CANCEL_REQUEST" />
                        <value enum="2" description="ORDER_CANCEL_REPLACE_REQUEST" />
                </field>
                <field number="447" name="PartyIDSource" type="CHAR">
                        <value enum="D" description="PROPRIETARY_CUSTOM_CODE" />
                </field>
                <field number="448" name="PartyID" type="STRING" />
                <field number="452" name="PartyRole" type="INT">
                        <value enum="1" description="EXECUTING_FIRM" />
                        <value enum="3" description="CLIENT_ID" />
                        <value enum="35" description="LIQUIDITY_PROVIDER" />
                </field>
                <field number="453" name="NoPartyIDs" type="NUMINGROUP" />
                <field number="541" name="MaturityDate" type="LOCALMKTDATE" />
                <field number="554" name="Password" type="STRING" />
                <field number="559" name="SecurityListRequestType" type="INT">
                        <value enum="4" description="ALL_SECURITIES" />
```

```
</field>
<field number="527" name="SecondaryExecID" type="STRING" />
<field number="560" name="SecurityRequestResult" type="INT">
        <value enum="0" description="VALID_REQUEST" />
        <value enum="1" description="INVALID_OR_UNSUPPORTED_REQUEST" />
</field>
<field number="631" name="MidPx" type="PRICE" />
<field number="640" name="Price2" type="PRICE" />
<field number="641" name="LastForwardPoints2" type="PRICEOFFSET" />
<field number="642" name="BidForwardPoints2" type="PRICEOFFSET" />
<field number="643" name="OfferForwardPoints2" type="PRICEOFFSET" />
<field number="693" name="QuoteRespID" type="STRING" />
<field number="694" name="QuoteRespType" type="INT">
        <value enum="1" description="HIT_LIFT" />
        <value enum="6" description="PASS" />
</field>
<field number="893" name="LastFragment" type="BOOLEAN" />
<field number="1026" name="MDEntrySpotRate" type="PRICE" />
<field number="386" name="NoTradingSessions" type="NUMINGROUP" />
<field number="4539" name="BidSwapPoints" type="PRICE" />
<field number="4540" name="OfferSwapPoints" type="PRICE" />

<field number="6052" name="BidSize2" type="QTY" />
<field number="6053" name="OfferSize2" type="QTY" />
<field number="6159" name="AvgPx2" type="PRICE" />
<field number="6160" name="LastPx2" type="PRICE" />
<field number="6161" name="LastSpotRate2" type="PRICE" />
<field number="6162" name="BidSpotRate2" type="PRICE" />
<field number="6163" name="OfferSpotRate2" type="PRICE" />
<field number="6808" name="LastQty2" type="QTY" />
<field number="6164" name="LeavesQty2" type="QTY" />
<field number="6165" name="CumQty2" type="QTY" />
<field number="7012" name="ReferenceEquivalentQty" type="QTY" />
<field number="7013" name="ReferenceEquivalentQty2" type="QTY" />
<field number="7576" name="BidPx2" type="PRICE" />
<field number="7577" name="OfferPx2" type="PRICE" />

<field number="9000" name="NoRequestedSize" type="INT" />
<field number="9001" name="RequestedSize" type="INT" />
<field number="9122" name="MDEntryOrigTime" type="UTCTIMEONLY" />
<field number="9998" name="FarMidPx" type="PRICE" />

<field number="9999" name="SettlType2" type="STRING">
        <value enum="ONI" description="ONI"></value>
        <value enum="TOD" description="TOD"></value>
        <value enum="TOM" description="TOM"></value>
        <value enum="TNX" description="TNX"></value>
        <value enum="SP" description="SP"></value>
        <value enum="SNX" description="SNX"></value>
        <value enum="D2" description="D2"></value>
        <value enum="D3" description="D3"></value>
        <value enum="D4" description="D4"></value>
        <value enum="W1" description="W1"></value>
        <value enum="W2" description="W2"></value>
        <value enum="W3" description="W3"></value>
        <value enum="M1" description="M1"></value>
        <value enum="M2" description="M2"></value>
        <value enum="M3" description="M3"></value>
        <value enum="M4" description="M4"></value>
        <value enum="M5" description="M5"></value>
        <value enum="M6" description="M6"></value>
        <value enum="M7" description="M7"></value>
        <value enum="M8" description="M8"></value>
        <value enum="M9" description="M9"></value>
        <value enum="M10" description="M10"></value>
        <value enum="M11" description="M11"></value>
        <value enum="Y1" description="Y1"></value>
        <value enum="M15" description="M15"></value>
        <value enum="M18" description="M18"></value>
        <value enum="M21" description="M21"></value>
        <value enum="Y2" description="Y2"></value>
        <value enum="Y3" description="Y3"></value>
        <value enum="Y4" description="Y4"></value>
        <value enum="Y5" description="Y5"></value>
```

```
                        <value enum="Y6" description="Y6"></value>
                        <value enum="Y7" description="Y7"></value>
                        <value enum="Y8" description="Y8"></value>
                        <value enum="Y9" description="Y9"></value>
                        <value enum="Y10" description="Y10"></value>
                        <value enum="Y15" description="Y15"></value>
                        <value enum="Y20" description="Y20"></value>
                        <value enum="Y25" description="Y25"></value>
                        <value enum="Y30" description="Y30"></value>
                        <value enum="IM1" description="IMM1"></value>
                        <value enum="IM2" description="IMM2"></value>
                        <value enum="IM3" description="IMM3"></value>
                        <value enum="IM4" description="IMM4"></value>
                        <value enum="B" description="B"></value>
                </field>
                <field number="20001" name="UTIPrefix" type="STRING" />
                <field number="20002" name="UTI" type="STRING" />
                <field number="20003" name="UTIPrefix2" type="STRING" />
                <field number="20004" name="UTI2" type="STRING" />

                <field number="639" name="PriceImprovement" type="PRICE" />

                <field number="847" name="TargetStrategy" type="INT">
                        <value enum="1" description="VWAP" />
                        <value enum="2000" description="DMA" />
                </field>

        </fields>
</fix>
```