cTrader FIX Engine, Rules of Engagement

Spotware Systems Ltd

Version 2.13.4, 2017-08

Table Of Contents

1. Changes

Version	Date	Changes
2.11.1	2017-03-31	* added AbsoluteTP, RelativeTP, AbsoluteSL, RelativeSL, TrailingSL, TriggerMethodSL, GuaranteedSL to Execution Report * added AbsoluteTP, AbsoluteSL, TrailingSL, TriggerMethodSL, GuaranteedSL to Position Report
2.12.1	2017-04-04	* added TotNumReports to Execution Report * added Order Status(I) value to ExecType
2.12.2	2017-04-06	* types of RelativeTP and RelativeSL were changed from 'Integer' to 'Price' for Execution Report
2.12.3	2017-04-10	* added Order Cancel Request
2.12.4	2017-05-29	* added support fractional volumes New Order Single
2.12.5	2017-07-19	* added example for failed Logon
2.13.0	2017-07-20	* TimeInForce is optional and deprecated for New Order Single
2.13.1	2017-07-21	* Wrong Checksum and Length values were fixed in Examples
2.13.2	2017-07-24	* Updated Order Cancel Request example
2.13.3	2017-08-02	* TimeInForce description updated for New Order Single
2.13.4	2017-08-29	* BeginSeqNo is required for ResendRequest * RejSeqNum changed to RefSeqNum (typo fix) * AvgPx and TransactTime are optional for ExecutionReport * Username and Password are optional for Logon

2. Disclaimer

This document is made publicly available on https://help.spotware.com/FIX, it can be revised at any time and without any prior notice. It is recommended to regularly check the aforementioned URL for new revisions. This document remains the propriety intellectual property of Spotware Systems Ltd. It may be re-distributed on an as-is basis only. It may not be modified or reproduced without our connect. By using this document you agree to our EULA.

3. Introduction

3.1. Scope

This document is intended to serve software developers as an implementation guide for the cTrader FIX Engine Application Programming Interface (API).

3.2. FIX Version

cTrader supports FIX version 4.4. For further information about this version please refer to the specifications published by the FIX Protocol Organization under http://www.fixprotocol.org/specifications/FIX.4.4.

4. Connectivity

4.1. Connection type

Connection to cTrader's FIX engine is available over the Internet, VPN tunnel or cross-connect to our datacenter facilities in the UK. Please contact us for further details.

4.2. Sequence number reset

All sides of FIX session should have sequence numbers reset on establishing FIX session. See Logon message.

5. Messages

As defined in the FIX protocol, the cTrader FIX server is using two different data levels: System and Application. Please note that this is the minimum set of messages required to support the necessary work flows and is subject to change over time as both business needs and the FIX standard evolve.

cTrader FIX Engine

System Messages

- Heartbeat (Client ⇔ cTrader)
- Test Request (Client ⇔ cTrader)
- Logon (Client → cTrader)
- Logout (Client → cTrader)
- Resend Request (Client ⇔ cTrader)
- Reject (Client ⇔ cTrader)
- Sequence Reset (Client ⇔ cTrader)

Application messages

- Market Data Request (Client → cTrader)
- Market Data Snapshot/Full Refresh (Client ← cTrader)
- Market Data Incremental Refresh (Client ← cTrader)
- New Order Single (Client → cTrader)
- Order Status Request (Client → cTrader)
- Order Mass Status Request (Client → cTrader)
- Execution Report (Client ← cTrader)
- Business Message Reject (Client ← cTrader)
- Request For Positions (Client → cTrader)
- Position Report (Client ← cTrader)
- Order Cancel request (Client → cTrader)

5.1. Standard Header

Each administrative or application message is preceded by a standard header. The header identifies the message type, length, destination, sequence number, origination point and time.

All messages sent to cTrader should have a standard header with following fields:

Tag	Field Name	Req'd	Value	FIX Format	Comments
8	BeginString	Y	FIX.4.4		FIX.4.4 (Always unencrypted, must be first field in message).
9	BodyLength	Y	-	Integer	Message body length. Always unencrypted, must be second field in message.
35	MsgType	Y	A	String	Message type. Always unencrypted, must be third field in message.
49	SenderCompID	Y	-	String	ID of the trading party in following format: "<BrokerUID>.<Trader Login>" where BrokerUID is provided by cTrader and Trader Login is numeric identifier of the trader account.
56	TargetCompID	Y	CSERVER	String	Message target. Valid value is "CSERVER".
57	TargetSubID	Y		String	Additional session qualifier. Possible values are: "QUOTE", "TRADE".

50	SenderSubID	N		String	Assigned value used to identify specific message originator.
34	MsgSeqNum	Y	1	Integer	Message sequence number.
52	SendingTime	Y	20131129-15:40:08.155	UTCTimestamp	Time of message transmission (always expressed in UTC (Universal Time Coordinated, also known as 'GMT').

5.2. Standard Trailer

Each message, administrative or application, is terminated by a standard trailer. The trailer is used to segregate messages and contains the three digit character representation of the CheckSum <10> value.

Tag	Field Name	Req'd	Value	FIX Format	Comments
10	CheckSum	Y	054	String	Three byte, simple checksum. Always last field in message; i.e. serves, with the trailing <SOH>, as the end-of-message delimiter. Always defined as three characters (and always unencrypted).

5.3. Session Messages

5.3.1. Heartbeat (MsgType(35)=0)

Heartbeat messages are sent by both cTrader and client application to confirm a live connection.

The provider's client application transmits a recurring heartbeat at the interval (defined by 'HeartBtInt (#108)' field in Logon message) or as a response to a Test Request message.

Tag	Field Name	Req'd	Value	FIX Format	Comments
	Standard Header	Y			
112	TestReqID	N	—	String	If heartbeat is result of a Test Request message, TestReqID is required.
	Standard Trailer	Y			

5.3.2. Test Request (MsgType(35)=1)

Forces heartbeat from receiver of request. A response is sent from the receiving system as a Heartbeat message containing the 'TestReqID'.

Tag	Field Name	Req’d	Value	FIX Format	Comments
	Standard Header	Y			
112	TestReqID	Y	—	String	Heartbeat message ID. TestReqID should be incremental.
	Standard Trailer	Y			

5.3.3. Logon (Bidirectional) (MsgType(35)=A)

The logon message is sent from the client side application to begin a cTrader FIX session, and a response is sent by cTrader to the client side application. Once logon is complete, quote and trade flows can proceed for the lifecycle of the session.

If an invalid logon messages received by cTrader (invalid fields), cTrader sends a logout message in response.

Tag	Field Name	Req’d	Value	FIX Format	Comments
	Standard Header	Y			
98	EncryptMethod	Y	0	int	Defines a message encryption scheme. Currently, only transport-level security is supported. Valid value is "0" (zero)= NONE_OTHER (encryption is not used).
108	HeartBtInt	Y	—	int	Heartbeat interval in seconds. Value is set in the 'config.properties' file (client side) as 'SERVER.POLLING.INTERVAL'. 30 seconds is default interval value. If HeartBtInt is set to 0, no heart beat message is required.
141	ResetSeqNumFlag	N	Y	Boolean	All sides of FIX session should have sequence numbers reset. Valid value is "Y"=Yes (reset).
553	Username	N	—	String	The numeric User ID. User is linked to SenderCompID (#49) value (the user’s organization).
554	Password	N	—	String	User password.
	Standard Trailer	Y			

Please note, that field Username (553) must contain numeric trader login value, whilst SenderCompID (49) must contain both BrokerUID and trader login delimited by dot, e.g.

"theBroker.12345".

See Logon example.

5.3.4. Logout (MsgType(35)=5)

Logout message is sent from the client application to request session end with cTrader and sent as a response by cTrader. A session logout occurs in response to a Market Participant sending a Logout message to cTrader. Before terminating the session, cTrader will cancel all prices that are still actively streaming out to the requesting party.

If an invalid logon message is received by cTrader (invalid fields), cTrader sends a logout message in response with error details in the field 'Text (#58)'.

Tag	Field Name	Req'd	Value	FIX Format	Comments
	Standard Header	Y			
58	Text	N	—	String	Logon rejection details. Used only for cTrader-to-client messages as an invalid Logon message response.
	Standard Trailer	Y			

See Logout example.

5.3.5. Resend Request (MsgType(35)=2)

Inbound/Outbound message used to request resending a message (or messages), typically when a gap is detected in the sequence numbering.

Tag	Field Name	Req'd	Value	FIX Format	Comments
	Standard Header	Y			
7	BeginSeqNo	Y		Integer	Message sequence number of first record in range to be resent.
16	EndSeqNo	Y		Integer	Message sequence number of last record in range to be resent.
	Standard Trailer	Y			

5.3.6. Reject (Bidirectional) (MsgType(35)=3)

Sent when a received message cannot be processed due to a session-level rule violation. Refused messages must be recorded and an increment must be applied to the incoming sequence number.

Tag	Field Name	Req'd	Value	FIX Format	Comments
	Standard Header	Y			
45	RefSeqNum	Y	—	SeqNum	Referenced message sequence number.
58	Text	N	—	String	Free format text string.
354	EncodedText Len	N	—	Length	Length of EncodedText (non-ASCII characters) field in bytes.
355	EncodedText	N	—	data	Representation of the Text (#58) field, encoded using the format specified in the MessageEncoding (#347) field (contained in the standard header). If the encoded format specified via the MessageEncoding (#347) field in the standard header. If an ASCII representation is used, it should also be specified in the filed Text (#58).
371	RefTagID	N	—	Int	Tag number of the FIX field that initiated the message refusal.
372	RefMsgType	N	—	String	The MsgType (#35) of the referenced FIX message.
373	SessionReject Reason	N	—	Int	Coded causes of the rejection. Valid values: 0 = Invalid tag number; 1 = Missing required tag; 2 = No tag defined for this message type; 3 = Undefined Tag; 4 = No value for specified tag; 5 = Value for this tag is out of range; 6 = Incorrect data format for value; 7 = Decryption problem; 8 = Signature error; 9 = CompID error; 10 = SendingTime accuracy error; 11 = MsgType invalid; 12 = XML Validation error; 13 = Tag is being repeated; 14 = Specified tag is not in correct order; 15 = Repeating group fields not in correct order; 16 = Incorrect NumInGroup count for repeating group; 17 = Field delimiter (SOH character) included in non data value.

	Standard Trailer	Y			

5.3.7. Sequence Reset (MsgType(35)=4)

Inbound/Outbound message is used by the sending application to reset the incoming sequence number on the opposing side.

Tag	Field Name	Req'd	Value	FIX Format	Comments
	Standard Header	Y			
123	GapFillFlag	N	Y or N	String	Indicates that the Sequence Reset message is replacing administrative or application messages which will not be resent.
36	NewSeqNo	Y	1	Integer	New sequence number.
	Standard Trailer	Y			

5.4. Application Messages

5.4.1. Market Data Request (MsgType(35)=V)

Tag	Field Name	Req'd	Value	FIX Format	Comments
	Standard Header	Y			
262	MDReqID	Y		String	Unique quote request id. New ID for a new subscription, same one as previously used for subscription removal.
263	SubscriptionRequestType	Y	1	Char	1 = Snapshot plus updates (subscribe) 2 = Disable previous snapshot plus update request (unsubscribe)
264	MarketDepth	Y	0 or 1	Integer	Full book will be provided, 0 = Depth subscription; 1 = Spot subscription.
265	MDUpdateType	N	1	Integer	Only Incremental refresh is supported.
267	NoMDEntryTypes	Y	2	Integer	Always set to 2 (both bid and ask will be sent)

269	MDEntryType	Y	0 or 1	Char	This repeating group contains a list of all types of Market Data Entries the requester wants to receive. 0 = Bid; 1 = Offer.
146	NoRelatedSym	Y		Integer	Number of symbols requested.
55	Symbol	N		Long	Instrument identificators are provided by Spotware.
	Standard Trailer	Y			

See Market Data Request example.

5.4.2. Market Data Snapshot/Full Refresh (MsgType(35)=W)

Tag	Field Name	Req'd	Value	FIX Format	Comments
	Standard Header	Y			
262	MDReqID	N		String	ID of the market data request previously sent.
55	Symbol	Y		Long	Instrument identificators are provided by Spotware.
268	NoMDEntries	Y		Integer	Number of entries following.
269	MDEntryType	Y	0	Char	Valid values are: 0 = BID; 1 = OFFER.
270	MDEntryPx	Y	1.2345	Price	Price of Market Data Entry.
	Standard Trailer	Y			

5.4.3. Market Data Incremental Refresh (MsgType(35)=X)

Tag	Field Name	Req'd	Value	FIX Format	Comments
	Standard Header	Y			
262	MDReqID	N		String	ID of the market data request previously sent.
268	NoMDEntries	Y		Integer	Number of entries following. This repeating group contains a list of all types of Market Data Entries the requester wants to receive.

279	MDUpdateAction	Y	0	Char	Type of Market Data update action. Valid values: 0 = NEW; 2 = DELETE.
269	MDEntryType	N	0	Char	Valid values are: 0 = BID; 1 = OFFER.
278	MDEntryID	Y			ID of Market Data Entry.
55	Symbol	Y		Long	Instrument identificators are provided by Spotware.
270	MDEntryPx	N	1.2345	Price	Conditionally required when MDUpdateAction <279> = New(0).
271	MDEntrySize	N	10000.00	Duble	Conditionally required when MDUpdateAction <279> = New(0).
	Standard Trailer	Y			

5.4.4. New Order Single (MsgType(35)=D)

The new order single message has the following format:

Tag	Field Name	Req’d	Value	FIX Format	Comments
	Standard Header	Y			
11	ClOrdID	Y		String	Unique identifier for the order, allocated by the client.
55	Symbol	Y		Long	Instrument identificators are provided by Spotware.
54	Side	Y	1	Integer	1 = Buy; 2 = Sell.
60	TransactTime	Y		Timestamp	Client generated request time.
38	OrderQty	Y		Qty	Number of shares ordered. This represents the number of shares for equities or based on normal convention the number of contracts for options, futures, convertible bonds, etc. Maximum precision is 0.01 (Prior to FIX 4.2 this field was of type int)

Tag	Field Name	Req'd	Value	FIX Format	Comments
40	OrdType	Y	1, 2 or 3	Char	1 = Market, the Order will be processed by 'Immediate Or Cancel' scheme (see TimeInForce(59): IOC); 2 = Limit, the Order will be processed by 'Good Till Cancel' scheme (see TimeInForce(59): GTC); 3 = Stop, the Order will be processed by 'Good Till Cancel' scheme (see TimeInForce(59): GTC).
44	Price	N		Price	The worst client price that the client will accept. Required when OrdType = 2, in which case the order will not fill unless this price can be met.
99	StopPx	N		Price	Price that triggers a Stop order. Required when OrdType = 3, in which case the order will not fill unless this price can be met.
59	TimeInForce	N		String	Deprecated, this value will be ignored. TimeInForce will be detected automaticly depending on `OrdType(40)` and `ExpireTime(126)`: 1 = Good Till Cancel (GTC), will be used only for Limit and Stop Orders (see `OrdType(40)`) only if `ExpireTime(126)` is not defined 6 = Good Till Date (GTD), will be used only for Limit and Stop Orders (see `OrdType(40)`) only if `ExpireTime(126)` is defined; 3 = Immediate Or Cancel (IOC), will be used only for Market Orders (see OrdType(40));
126	ExpireTime	N	20140215-07:24:55	Timestamp	Expire Time in YYYYMMDD-HH:MM:SS format. If is assigned then the Order will be processed by 'Good Till Date' scheme (see TimeInForce: GTD).
721	PosMaintRptID	N		String	Position ID, where this order should be placed. If not set, new position will be created, it's id will be returned in ExecutionReport(8) message.
494	Designation	N		String	Client custom order label.

Tag	Field Name	Req'd	Value	FIX Format	Comments
	Standard Trailer	Y			

See New Order Single example.

5.4.5. Order Status Request (MsgType(35)=H)

The Order Status Request <H> message is used by the institution to generate an order status message back from the trader. For correct interaction it is very important to have unique client order identifiers (ClOrdID) for all orders.

Tag	Field Name	Req'd	Value	FIX Format	Comments
	Standard Header	Y			
11	ClOrdID	Y		String	Unique identifier for the order, allocated by the client.
54	Side	N	1	Integer	1 = Buy; 2 = Sell. There is for the FIX compatibility only, so it will be ignored.
	Standard Trailer	Y			

See Order Status Request example.

5.4.6. Order Mass Status Request (MsgType(35)=AF)

The Order Mass Status Request <H> message requests the status for orders matching criteria specified within the request. Answer will be returned as a number of Execution Report(8) messages (one for each order) or Business Message Reject(j) message if no orders found.

Tag	Field Name	Req'd	Value	FIX Format	Comments
	Standard Header	Y			
584	MassStatusReqID	Y		String	Unique ID of mass status request as assigned by the client.
585	MassStatusReqType	Y	1	Integer	7 = Status for all orders; only value 7 is currently supported;
225	IssueDate	N		String	If set, response will contain only orders created before this date.
	Standard Trailer	Y			

See Order Mass Status Request example.

5.4.7. Execution Report (MsgType(35)=8)

The execution report message for an accepted order has the following format:

Tag	Field Name	Req'd	Value	FIX Format	Comments
	Standard Header	Y			
37	OrderID	Y		String	cTrader order id.
11	ClOrdID	N		String	Unique identifier for the order, allocated by the client.
911	TotNumReports	N		Integer	Total number or reports returned in response to a OrderMassStatusRequest.
150	ExecType	Y	F	Char	0 = New; 4 = Canceled; 5 = Replace; 8 = Rejected; C = Expired; F = Trade; I = Order Status.
39	OrdStatus	Y	2	Char	0 = New; 1 = Partially filled; 2 = Filled; 8 = Rejected; 4 = Cancelled (When an order is partially filled, "Cancelled" is returned signifying Tag 151: LeavesQty is cancelled and will not be subsequently filled); C = Expired.
55	Symbol	N		Long	Instrument identificators are provided by Spotware.
54	Side	N	1	Integer	1 = Buy; 2 = Sell.
60	TransactTime	N		Timestamp	Time the transaction represented by this ExecutionReport occurred message (in UTC).
6	AvgPx	N		Integer	The price at which the deal was filled. For an IOC or GTD order, this is the VWAP (Volume Weighted Average Price) of the filled order.

38	OrderQty	N		Qty	Number of shares ordered. This represents the number of shares for equities or based on normal convention the number of contracts for options, futures, convertible bonds, etc. (Prior to FIX 4.2 this field was of type int)
151	LeavesQty	N		Qty	The amount of the order still to be filled. This is a value between 0 (fully filled) and OrderQty (partially filled).
14	CumQty	N		Qty	The total amount of the order which has been filled.
40	OrdType	N	1 or 2	Char	1 = Market; 2 = Limit.
44	Price	N		Price	If supplied in the NewOrderSingle, it is echoed back in this ExecutionReport.
99	StopPx	N		Price	If supplied in the NewOrderSingle, it is echoed back in this ExecutionReport.
59	TimeInForce	N	3	String	1 = Good Till Cancel (GTC); 3 = Immediate Or Cancel (IOC); 6 = Good Till Date (GTD).
126	ExpireTime	N	20140215-07:24:55	Timestamp	If supplied in the NewOrderSingle, it is echoed back in this ExecutionReport.
58	Text	N		String	Where possible, message to explain execution report.
103	OrdRejReason	N	0	Integer	0 = OrdRejReason.BROKER_EXCHANGE_OPTION
721	PosMaintRptID	N		String	Position ID.
494	Designation	N		String	Client custom order label.
584	MassStatusReqID	N		String	Unique ID of mass status request as assigned by the client.
1000	AbsoluteTP	N		Price	The absolute price at which Take Profit will be triggered.
1001	RelativeTP	N		Price	The distance in pips from the entry price at which the Take Profit will be triggered.

1002	AbsoluteSL	N		Price	The absolute price at which Stop Loss will be triggered.
1003	RelativeSL	N		Price	The distance in pips from the entry price at which the Stop Loss will be triggered.
1004	TrailingSL	N		Boolean	Indicates if Stop Loss is trailing. N = The Stop Loss is not trailing, Y = The Stop Loss is trailing.
1005	TriggerMethodSL	N		Integer	Indicated trigger method of the Stop Loss. 1 = The Stop Loss will be triggered by the trade side. 2 = The stop loss will be triggered by the opposite side (Ask for Buy positions and by Bid for Sell positions), 3 = Stop Loss will be triggered after two consecutive ticks according to the trade side. 4 = Stop Loss will be triggered after two consecutive ticks according to the opposite side (second Ask tick for Buy positions and second Bid tick for Sell positions).
1006	GuaranteedSL	N		Boolean	Indicates if Stop Loss is guaranteed. N = The Stop Loss is not guarranteed, Y = The Stop Loss is guarranteed.
	Standard Trailer	Y			

See New Order Single example.

5.4.8. Business Message Reject (MsgType(35)=j)

This type of message is sent when the system was unable to process subscription request or an order cannot be executed.

Tag	Field Name	Req'd	Value	FIX Format	Comments
	Standard Header	Y			
45	RefSeqNum	N		Integer	MsgSeqNum<34> of rejected message.
372	RefMsgType	N		String	The MsgType<35> of the FIX message being referenced.

379	BusinessRejectRefID	N		String	The value of the business-level 'ID' field on the message being referenced. Required unless the corresponding ID field was not specified.
380	BusinessRejectReason	Y	0	Integer	Code to identify reason for a Business Message Reject<j> message. 0 = OTHER.
58	Text	N		String	Where possible, message to explain reason for rejection.
	Standard Trailer	Y			

See Business Message Reject example.

5.4.9. Request For Positions (MsgType(35)=AN)

Tag	Field Name	Req'd	Value	FIX Format	Comments
	Standard Header	Y			
710	PosReqID	Y		String	Unique request ID (set by client).
721	PosMaintRptID	N		String	Position ID to request. If not set, all open positions will be returned.
	Standard Trailer	Y			

See Request For Positions example.

5.4.10. Position Report (MsgType(35)=AP)

Tag	Field Name	Req'd	Value	FIX Format	Comments
	Standard Header	Y			
710	PosReqID	Y		String	Id of RequestForPositions.
721	PosMaintRptID	N		String	Position ID (is not set if PosReqResult(728) is not VALID_REQUEST).
727	TotalNumPosReports	Y		String	Total count of PositionReport's in sequence when PosReqResult(728) is VALID_REQUEST, otherwise = 0.
728	PosReqResult	Y		String	0 = Valid Request; 2 = No open positions found that match criteria.

55	Symbol	N		String	The symbol for which the current Position Report is prepared. (is not set if PosReqResult(728) is not VALID_REQUEST).
702	NoPositions	N		String	1 when PosReqResult(728) is VALID_REQUEST, otherwise not set.
704	LongQty	N		String	Position’s open volume in case of BUY trade side, = 0 in case of SELL trade side, is not set if PosReqResult(728) is not VALID_REQUEST.
705	ShortQty	N		String	Position’s open volume in case of SELL trade side, = 0 in case of BUY trade side, is not set if PosReqResult(728) is not VALID_REQUEST.
730	SettlPrice	N		String	Average price of the opened volume in the current PositionReport.
1000	AbsoluteTP	N		Price	The absolute price at which Take Profit will be triggered.
1002	AbsoluteSL	N		Price	The absolute price at which Stop Loss will be triggered.
1004	TrailingSL	N		Boolean	Indicates if Stop Loss is trailing. N = The Stop Loss is not trailing, Y = The Stop Loss is trailing.
1005	TriggerMethodSL	N		Integer	Indicated trigger method of the Stop Loss. 1 = The Stop Loss will be triggered by the trade side. 2 = The stop loss will be triggered by the opposite side (Ask for Buy positions and by Bid for Sell positions), 3 = Stop Loss will be triggered after two consecutive ticks according to the trade side. 4 = Stop Loss will be triggered after two consecutive ticks according to the opposite side (second Ask tick for Buy positions and second Bid tick for Sell positions).
1006	GuaranteedSL	N		Boolean	Indicates if Stop Loss is guaranteed. N = The Stop Loss is not guarranteed, Y = The Stop Loss is guarranteed.
	Standard Trailer	Y			

5.4.11. Order Cancel Request (MsgType(35)=F)

Tag	Field Name	Req'd	Value	FIX Format	Comments
	Standard Header	Y			
41	OrigClOrdID	Y		String	Unique identifier for the order,which is going to be canceled, allocated by the client.
37	OrderID	N		String	Unique ID of an order, returned be cServer.
11	ClOrdID	Y		String	Unique ID of cancel request, allocated by the client.
	Standard Trailer	Y			

See Order Cancel Request example.

6. Examples

6.1. Logon

```
Request
8=FIX.4.4|9=126|35=A|49=theBroker.12345|56=CSERVER|34=1|52=20170117-
08:03:04|57=TRADE|50=any_string|98=0|108=30|141=Y|553=12345|554=passw0rd!|10=131|

Response (Success)
8=FIX.4.4|9=106|35=A|34=1|49=CSERVER|50=TRADE|52=20170117-
08:03:04.509|56=theBroker.12345|57=any_string|98=0|108=30|141=Y|10=066|

Response (Failed)
8=FIX.4.4|9=109|35=5|34=1|49=CSERVER|50=TRADE|52=20170117-
08:03:04.509|56=theBroker.12345|58=InternalError: RET_INVALID_DATA|10=033|
```

6.2. Logout

```
Request
8=FIX.4.4|9=86|35=5|49=theBroker.12345|56=CSERVER|34=161|52=20170117-
09:22:33|57=TRADE|50=any_string|10=102|

Response
8=FIX.4.4|9=90|35=5|34=160|49=CSERVER|50=TRADE|52=20170117-
09:22:33.077|56=theBroker.12345|57=any_string|10=044|
```

6.3. Market Data Request

For spots

```
Request
8=FIX.4.4|9=131|35=V|49=theBroker.12345|56=CSERVER|34=3|52=20170117-
10:26:54|50=QUOTE|262=876316403|263=1|264=1|265=1|146=1|55=1|267=2|269=0|269=1|10=094|

Response
8=FIX.4.4|9=134|35=W|34=2|49=CSERVER|50=QUOTE|52=20170117-
10:26:54.630|56=theBroker.12345|57=any_string|55=1|268=2|269=0|270=1.06625|269=1|270=1
.0663|10=118|
```

For depths

```
Request
8=FIX.4.4|9=131|35=V|49=theBroker.12345|56=CSERVER|34=2|52=20170117-
11:13:44|50=QUOTE|262=876316411|263=1|264=0|265=1|146=1|55=1|267=2|269=0|269=1|10=087|

Responses
8=FIX.4.4|9=693|35=X|34=2|49=CSERVER|50=QUOTE|52=20170117-
11:13:44.461|56=theBroker.12345|57=any_string|268=12|279=0|269=1|278=7475|55=1|270=1.0
691|271=2000000|279=0|269=1|278=7476|55=1|270=1.06911|271=3000000|279=0|269=1|278=7484
|55=1|270=1.06931|271=34579000|279=0|269=1|278=7485|55=1|270=1.06908|271=1000000|279=0
|269=1|278=7483|55=1|270=1.06906|271=500000|279=0|269=1|278=7482|55=1|270=1.06907|271=
500000|279=0|269=1|278=7488|55=1|270=1.06909|271=3000000|279=0|269=0|278=7468|55=1|270
=1.06898|271=500000|279=0|269=0|278=7467|55=1|270=1.06874|271=32371000|279=0|269=0|278
=7457|55=1|270=1.06899|271=1000000|279=0|269=0|278=7478|55=1|270=1.06896|271=7000000|2
79=0|269=0|278=7477|55=1|270=1.06897|271=1500000|10=111|

8=FIX.4.4|9=376|35=X|34=3|49=CSERVER|50=QUOTE|52=20170117-
11:13:44.555|56=theBroker.12345|57=any_string|268=8|279=0|269=0|278=7491|55=1|270=1.06
897|271=1000000|279=0|269=0|278=7490|55=1|270=1.06898|271=1000000|279=0|269=0|278=7489
|55=1|270=1.06874|271=32373000|279=0|269=1|278=7496|55=1|270=1.06931|271=34580000|279=
2|278=7477|55=1|279=2|278=7468|55=1|279=2|278=7467|55=1|279=2|278=7484|55=1|10=192|
```

6.4. New Order Single

Market order to new position

```
Request
8=FIX.4.4|9=143|35=D|49=theBroker.12345|56=CSERVER|34=77|52=20170117-
10:02:14|50=any_string|57=TRADE|11=876316397|55=1|54=1|60=20170117-
10:02:14|40=1|38=10000|10=010|

Responses
8=FIX.4.4|9=197|35=8|34=77|49=CSERVER|50=TRADE|52=20170117-
10:02:14.720|56=theBroker.12345|57=any_string|11=876316397|14=0|37=101|38=10000|39=0|4
0=1|54=1|55=1|59=3|60=20170117-10:02:14.591|150=0|151=10000|721=101|10=149|

8=FIX.4.4|9=206|35=8|34=78|49=CSERVER|50=TRADE|52=20170117-
10:02:15.045|56=theBroker.12345|57=any_string|6=1.0674|11=876316397|14=10000|37=101|38
=10000|39=2|40=1|54=1|55=1|59=3|60=20170117-10:02:14.963|150=F|151=0|721=101|10=077|
```

Market order to existing position

```
Request
8=FIX.4.4|9=151|35=D|49=theBroker.12345|56=CSERVER|34=80|52=20170117-
10:02:55|50=any_string|57=TRADE|11=876316398|55=1|54=1|60=20170117-
10:02:55|40=1|38=10000|721=101|10=120|

Responses
8=FIX.4.4|9=197|35=8|34=80|49=CSERVER|50=TRADE|52=20170117-
10:02:56.003|56=theBroker.12345|57=any_string|11=876316398|14=0|37=102|38=10000|39=0|4
0=1|54=1|55=1|59=3|60=20170117-10:02:55.984|150=0|151=10000|721=101|10=156|

8=FIX.4.4|9=207|35=8|34=81|49=CSERVER|50=TRADE|52=20170117-
10:02:56.239|56=theBroker.12345|57=any_string|6=1.06735|11=876316398|14=10000|37=102|3
8=10000|39=2|40=1|54=1|55=1|59=3|60=20170117-10:02:56.210|150=F|151=0|721=101|10=127|
```

Limit order to existing position

```
Request
8=FIX.4.4|9=162|35=D|49=theBroker.12345|56=CSERVER|34=89|52=20170117-
10:06:22|50=any_string|57=TRADE|11=876316400|55=1|54=2|60=20170117-
10:06:22|40=2|44=1.07162|38=50000|721=101|10=122|

Response
8=FIX.4.4|9=208|35=8|34=90|49=CSERVER|50=TRADE|52=20170117-
10:06:22.466|56=theBroker.12345|57=any_string|11=876316400|14=0|37=104|38=50000|39=0|4
0=2|44=1.07162|54=2|55=1|59=1|60=20170117-10:06:22.436|150=0|151=50000|721=101|10=149|
```

Stop order to new position

```
Request
8=FIX.4.4|9=153|35=D|49=theBroker.12345|56=CSERVER|34=9|52=20170117-
12:10:48|57=TRADE|50=any_string|11=876316418|55=1|54=1|60=20170117-
12:10:48|40=3|38=50000|99=1.07148|10=249|

Response
8=FIX.4.4|9=207|35=8|34=8|49=CSERVER|50=TRADE|52=20170117-
12:10:48.400|56=theBroker.12345|57=any_string|11=876316418|14=0|37=205|38=50000|39=0|4
0=3|54=1|55=1|59=1|60=20170117-12:10:48.362|99=1.07148|150=0|151=50000|721=202|10=122|
```

6.5. Order Status Request

```
Request
8=FIX.4.4|9=98|35=H|49=theBroker.12345|56=CSERVER|34=95|52=20170117-
10:08:31|50=any_string|57=TRADE|11=876316400|10=191|

Response
8=FIX.4.4|9=208|35=8|34=95|49=CSERVER|50=TRADE|52=20170117-
10:08:31.819|56=theBroker.12345|57=any_string|11=876316400|14=0|37=104|38=50000|39=0|4
0=2|44=1.07162|54=2|55=1|59=1|60=20170117-10:06:22.436|150=0|151=50000|721=101|10=158|
```

6.6. Order Mass Status Request

```
Request
8=FIX.4.4|9=117|35=AF|34=3|49=theBroker.12345|52=20170404-
07:20:55.325|56=CSERVER|57=TRADE|225=20170404-07:20:44.582|584=mZzEY|585=7|10=065|

Response
8=FIX.4.4|9=199|35=8|34=13|49=CSERVER|50=TRADE|52=20170404-
07:20:55.333|56=theBroker.12345|14=0|37=635|38=100000|39=0|40=2|44=1.35265|54=2|55=1|5
9=1|60=20170404-07:20:44.582|150=I|151=100000|584=mZzEY|721=617|911=1|10=152|
```

6.7. Business Message Reject

```
8=FIX.4.4|9=149|35=j|34=2|49=CSERVER|52=20170105-
06:36:00.912|56=theBroker.12345|57=any_string|58=Message to explain reason for
rejection|379=u4Jr7Rr5t2VS7HSP|380=0|10=123|
```

6.8. Request For Positions

```
Request
8=FIX.4.4|9=100|35=AN|49=theBroker.12345|56=CSERVER|34=99|52=20170117-
10:09:54|50=any_string|57=TRADE|710=876316401|10=103|

Response
8=FIX.4.4|9=163|35=AP|34=98|49=CSERVER|50=TRADE|52=20170117-
10:09:54.076|56=theBroker.12345|57=any_string|55=1|710=876316401|721=101|727=1|728=0|7
30=1.06671|702=1|704=0|705=30000|10=182|
```

6.9. Request For Cancel Order

```
Request
8=FIX.4.4|9=115|35=F|34=2|49=theBroker.12345|50=Trade|52=20170721-
13:41:21.694|56=CSERVER|57=TRADE|11=jR8dBPcZEQa9|41=n9Tm8x1AavO5|10=182|

Response (Success)
8=FIX.4.4|9=221|35=8|34=3|49=CSERVER|50=TRADE|52=20170721-
13:41:21.784|56=theBroker.12345|57=Trade|11=jR8dBPcZEQa9|14=0|37=641|38=100000|39=4|40
=2|41=n9Tm8x1AavO5|44=1.499|54=1|55=1|59=1|60=20170721-
13:41:21.760|150=4|151=100000|721=624|10=180|

Response (Failed)
8=FIX.4.4|9=174|35=j|34=3|49=CSERVER|50=TRADE|52=20170721-
13:41:21.856|56=theBroker.12345|57=Trade|58=ORDER_NOT_FOUND:Order with
clientOrderId=n9Tm8x1AavO5 not found.|379=jR8dBPcZEQa9|380=0|10=075|
```