Amberdata FIX Specification

Version 1.0.4

June 23rd, 2021

Table of Contents

Connection

The Financial Information eXchange (FIX) protocol is an open electronic communications protocol designed to standardise and streamline electronic communications in the financial services industry supporting multiple formats and types of communications between financial entities including trade allocation, order submissions, order changes, execution reporting and advertisements.

To connect to Amberdata's FIX servers, please use the following information:

Parameter	**Value**
Hostname	fix.web3api.io
Port	443
Version	FIX 4.4
Specification	Amberdata_FIX_Protocol_1.0.4.pdf
Dictionary	Amberdata_FIX44.xml

The FIX implementation follows the standard and conventional request messages found in the industry:

- **SecurityListRequest** request followed by **SecurityList** response
- **MarketDataRequest** request followed by either **MarketDataRequestReject**, **MarketDataSnapshotFullRefresh** or **MarketDataIncrementalRefresh** responses

We supports 4 different types of market data:

- Order Book Snapshots
- Order Book Updates
- Trades
- OHLCV

And 3 types of contracts:

- Spot
- Futures
- Swap

Message Structure

Standard Header

Tag	Name	Req	Type	Description	Example
8	BeginString	Y	String	Identifies the beginning of the message and protocol version.	8=FIX4.4
9	BodyLength	Y	Length	Message length, in bytes, forward to the CheckSum field.	9=286
35	MsgType	Y	String	Defines the message type (A=Logon, 0=Heartbeat, etc).	35=A
49	SenderCompID	Y	String	Identifier for the sender of the message.	49=acme.com
56	TargetCompID	Y	String	Identifier for the receiver of the message.	56=warner.com
34	MsgSeqNum	Y	SeqNum	The unique sequence number for the message within the current FIX session. Resets at Logon, Logout and Disconnect.	34=13056
62	SendingTime	Y	UTCTimestamp	Time of message transmission (expressed in UTC - YYYYMMDD-HH:MM:SS.sss).	52=20210413-05:51:15.877

Note that in order to successfully connect to the Amberdata's server, you need to:

- use the string **Amberdata** as the **TargetCompID**
- use your **API key** as the **SenderCompId**

Standard Trailer

Tag	Name	Req	Type	Description	Example
10	Checksum	Y	String	Three byte, simple checksum.	10=457

Custom Fields

In order to accurately and precisely support all the different types of data, a few fields needed to be added to the data dictionary, as they are not part of the original FIX specification.

Tag	Name	Type	Description
2000	MDEntryDepth	int	Depth of the entry in the order book snapshot
2001	TradeId	String	The unique trade identifier (if available)
2002	Open	Price	The Open price in the OHLCV aggregation or candle
2003	High	Price	The High price in the OHLCV aggregation or candle
2004	Low	Price	The Low price in the OHLCV aggregation or candle
2005	Close	Price	The Close price in the OHLCV aggregation or candle
2006	Volume	Amt	The Volume of trades in the OHLCV aggregation or candle

Session-Level Messages

Logon (A)

The Logon (A) message must be the first message sent by the application requesting to initiate a FIX session.

Clients must wait for a Logon from the server before sending other messages

Tag	Name	Req	Type	Description	Example
	Header	Y	MsgType='A'		
98	EncryptMethod	Y	int	Encryption Method - always '0'.	98=0
108	HeartBtInt	Y	int	Heartbeat interval in seconds - time to wait before a Heartbeat message is sent in the absence of other activity on the connection.	108=30
141	ResetSeqNumFlag	N	Boolean	Set to Y to indicate both sides of the FIX session should reset sequence numbers	141=Y
553	Username	Y	String	Session username - email address used during registration.	553=roger@warner.com
554	Password	Y	String	Session password - API key obtained during registration.	554=UAKabc...123
	Trailer	Y			

Logout (5)

Message used to terminate a FIX connection. Can be sent by the client or the server.

Tag	Name	Req	Type	Description	Example
	Header	Y	MsgType='5'		
58	Text	N	String	Reason for the logout.	58=Invalid username or password
	Trailer	Y			

Heartbeat (0)

Message sent during quiet intervals on the connection to ensure that connection is still alive.

Tag	Name	Req	Type	Description	Example
	Header	Y	MsgType='0'		
	TestReqID	N*	String	Required when the heartbeat is the result of a Test Request message.	112=f4e9
	Trailer	Y			

Security List Messages

SecurityListRequest (x)

Message sent by the client to subscribe to market data. Many different types of data are supported.

Tag	Name	Req	Type	Description	Example
	Header	Y	MsgType='x'		
320	SecurityReqID	Y	String	Unique identifier of this request.	320=234dsj
559	SecurityListRequestType	Y	int	The type/criteria of the request. 4 = All securities 6 = Spot 7 = Futures 8 = Swaps	559=4
207	SecurityExchange	N	String	Exchange Name - if not specified, all supported exchanges will be returned.	207=huobi
	Trailer	Y			

SecurityList (y)

Message sent by the server in response to a SecurityListRequest request.

Tag	Name	Req	Type	Description	Example
	Header	Y	MsgType='y'		
320	SecurityReqID	Y	String	Unique identifier matching the ID from the request.	320=234dsj
322	SecurityResponseID	Y	String	Unique identifier for the response.	322=234dsj
559	SecurityListRequestType	N	int	The type/criteria of the request. 4 = All securities 6 = Spot 7 = Futures 8 = Swaps	559=4
560	SecurityRequestResult	Y	int	Result of the request: 0 = Valid request 1 = Invalid or unsupported request 2 = No instruments/pairs found matching the selection criteria	560=0
146	NoRelatedSym	N	int	Number of returned symbols	146=10
55	Symbol	N	String	Symbol Name	55=BTC/USD
207	SecurityExchange	N	String	Exchange Name	207=huobi
	Trailer	Y			

Market Data Messages

MarketDataRequest (V)

Message sent by the client to subscribe to market data. Many different types of data are supported.

Tag	Name	Req	Type	Description	Example
	Header	Y	MsgType='V'		
262	MDReqID	Y	String	Must be unique, or the ID of previous Market Data Request to disable if SubscriptionRequestType = Disable previous Snapshot + Updates Request (2).	262=234dsj
263	SubscriptionRequestType	Y	char	Indicates what type of response is expected: - a snapshot request asks for current information - an update request asks for new information as soon as it is available - an unsubscribe request cancels a previous update request Types: 0 = Current Order Book Snapshot 1 = Current Order Book Snapshot followed by Order Book Events 2 = Cancel previous request of type 1	263=1

559	SecurityListRequestType	N	int	The contract type/criteria of the request. 6 = Spot 7 = Futures 8 = Swaps Default is 6 if not provided.	559=6
264	MarketDepth	Y	int	Used to specify the number of bids/asks to receive. At this time, only 0 = full order book is supported.	264=0
265	MDUpdateType	N*	int	Required if SubscriptionRequestType = 1 0 = Full Refresh 1 = Incremental Updates	265=1
267	NoMDEntryTypes	Y	NumInGroup	Number of MDEntryType (269) fields requested.	
=> 269	MDEntryType	Y	char	Type Market Data entry: 0 = Bid 1 = Offer 2 = Trade o = OHLCV Only the following combinations are supported: 1) Both bid and ask at the same time 2) Trades 3) OHLCV If a Market Data Request contains other combinations of	269=0

				MDEntryType, the request will be rejected.	
146	NoRelatedSym	Y	NumInGroup	Number of symbols requested. At this time, only one symbol at a time is supported.	146=1
=> 55	Symbol	Y	String	Symbol Name	55=BTC_USD
=> 207	SecurityExchange	N	String	Exchange Name - if not specified, all supported exchanges will be returned.	2000=HUOBI
=> 916	StartDate	N	LOCALMKTDATE	Used in Replay mode - note that both start & end dates have to be specified at this time.	
=> 917	EndDate	N	LOCALMKTDATE	Used in Replay mode - note that both start & end dates have to be specified at this time.	
	Trailer	Y			

MarketDataRequestReject (Y)

Used to reject the Market Data Request, due to business or technical reasons.

Tag	Name	Req	Type	Description	Example
	Header	Y	MsgType='Y'		
262	MDReqID	Y	String	Must refer to the MDReqID of the request.	262=234dsj
281	MDReqRejReason	N	char	Reason for the rejection of the request. Supported values: 0 = Unknown symbol 1 = Duplicate MDReqID 3 = Insufficient Permissions 4 = Unsupported SubscriptionRequestType 5 = Unsupported MarketDepth 6 = Unsupported MDUpdateType 8 = Unsupported MDEntryType	281=0
58	Text	N	String	Free format text string	58=Unknown symbol
	Trailer	Y			

MarketDataSnapshotFullRefresh (W)

This message is sent to the client when new market data is available for one of the clients market data subscriptions.

Note that the message will contain either an order book update or trades. Any given message will not contain both.

Tag	Name	Req	Type	Description	Example
	Header	Y	MsgType='W'		
262	MDReqID	Y	String	Must refer to the MDReqID of the request.	262=234dsj
55	Symbol	Y	String	Symbol Name	55=BTC_USD
2000	Exchange	Y	String	Exchange Name	2000=HUOBI
268	NoMdEntries	Y	NumInGroup	Number of entries in the order book	
=> 269	MDEntryType	Y	char	Market Data entry type: 0 = Bid 1 = Offer 2 = Trade o = OHLCV	269=0
=> 270	MDEntryPx	Y	Price	Price level of the order book level	270=14.948534
=> 271	MDEntrySize	Y	Qty	Quantity or volume represented by this entry.	271=15.67
=> 272	MDEntryDate	Y	UTCDateOnly	Date of the order in the book - tag will only be present on the first entry in the repeating group.	272=20210412
=> 273	MDEntryTime	Y	UTCTimeOnly	Time of the order in the book - tag will only be present on the first	273=23:56:12.853647

				entry in the repeating group.	
559	SecurityListRequestType	N	int	The contract type/criteria of the request. 6 = Spot 7 = Futures 8 = Swaps Default is 6 if not provided.	559=6
	Trailer	Y			

MarketDataIncrementalRefresh (X)

This message represents new executed trade or order book updates.

Tag	Name	Req	Type	Description	Example
	Header	Y	MsgType='X'		
262	MDReqID	Y	String	Must refer to the MDReqID of the request.	262=234dsj
55	Symbol	Y	String	Symbol Name	55=BTC_USD
2000	Exchange	Y	String	Exchange Name	2000=HUOBI
268	NoMdEntries	Y	NumInGroup	Number of entries in the order book	
=> 279	MDUpdateAction	Y	char	Type of update: 1 = Change 2 = Delete	279=1
=> 269	MDEntryType	Y	char	Market Data entry type: 0 - Bid 1 - Offer 2 - Trade o - OHLCV	269=0

=> 278	MDEntryID	N*	String	Trade Unique ID - only available for type Trade.	278=123249086
=> 270	MDEntryPx	Y	Price	Price level of the order book level	270=14.948534
=> 271	MDEntrySize	Y	Qty	Quantity or volume represented by this entry.	271=15.67
=> 272	MDEntryDate	Y	UTCDateOnly	Date of the order in the book - tag will only be present on the first entry in the repeating group.	272=20210412
=> 273	MDEntryTime	Y	UTCTimeOnly	Time of the order in the book - tag will only be present on the first entry in the repeating group.	273=23:56:12.853647
559	SecurityListRequestType	N	int	The contract type/criteria of the request. 6 = Spot 7 = Futures 8 = Swaps Default is 6 if not provided.	559=6
	Trailer	Y			

Examples

Fields

Requesting data using the FIX specification can be challenging at first, especially with all the different possible combinations.

As a reminder, here are all the fields that control the type of data one can request:

263 = SubscriptionRequestType

- 0 = Subscribe to Snapshot
- 1 = Subscribe to Snapshot + Updates
- 2 = Un-subscribe/Disable

264 = MarketDepth

- 0 = Full Book
- 1 = Quotes/Top of Book/BBO -- not supported today

265 = MDUpdateType

- 0 = Full refresh (no events, instead full order book snapshot is sent) - used for Order Book Snapshots and OHLCV
- 1 = Incremental refresh - used for Order Book updates and Trades

267/269 = MDEntryType

- 0 = Order Book Bids
- 1 = Order Book Asks/Offers
- 2 = Trades
- o = OHLCV

559 = SecurityListRequestType

- 4 = All Securities - Only supported for SecurityListRequest
- 6 = Spot contract
- 7 = Futures contract
- 8 = Swaps contract

146/207 = SecurityExchange - same exchange at this time

146/55 = Symbol - when requesting ongoing/historical data, only one symbol is supported

146/916 = StartDate - start of the replay

146/917 = EndDate - end of the replay

Replay requests have exactly the same configuration as normal or real-time requests, the only difference is the inclusion of the start and end dates in the request.

MarketDataRequest - Real-time

```
Real-time requests:
- Order Book Snapshot + Updates : 263=1, 264=0, 265=0, 267/269=0,267/269=1, 146/207=huobi, 146/55=btc_usd
- Order Book Snapshots          : 263=0, 264=0, 265=0, 267/269=0,267/269=1, 146/207=huobi, 146/55=btc_usd
- Order Book Updates            : 263=1, 264=0, 265=1, 267/269=0,267/269=1, 146/207=huobi, 146/55=btc_usd
- Trades                        : 263=1, 264=0, 265=1, 267/269=2,           146/207=huobi, 146/55=btc_usd
- OHLCV                         : 263=0, 264=0, 265=0, 267/269=3,           146/207=huobi, 146/55=btc_usd
```

MarketDataRequest - Replay

```
Replay requests:
- Order Book Snapshots : 263=0, 264=0, 265=0, 267/269=0,267/269=1, 146/207=huobi, 146/55=btc_usd, 146/916=..., 146/917=...
- Order Book Updates   : 263=1, 264=0, 265=1, 267/269=0,267/269=1, 146/207=huobi, 146/55=btc_usd, 146/916=..., 146/917=...
- Trades               : 263=1, 264=0, 265=1, 267/269=2,           146/207=huobi, 146/55=btc_usd, 146/916=..., 146/917=...
- OHLCV                : 263=0, 264=0, 265=0, 267/269=3,           146/207=huobi, 146/55=btc_usd, 146/916=..., 146/917=...
```