MarketData & Order **FIX Protocol 4.4**

FIX Protocol

Standard Header

Standard Trailer

FIX Session Management

Logon (A)

Heartbeat (0)

Logout (5)

FIX

Market Data

Market Data Request (V)

Market Data Request Reject (Y)

Market Data Snapshot (W)

Market Data Imcremental Snapshot (X)

Orders

New Order Single (D)

Execution Report (8)

Order Cancel Request (F)

Order Cancel Reject (9)

# FIX Protocol

## Standard Header

|  |  |  |  |
| --- | --- | --- | --- |
| **TAG** | **NAME** | **REQ** | **DESCRIPTION** |
| 8 | BeginString | Y | “FIX.4.4”. |
| 9 | BodyLength | Y | The length of the message. |
| 35 | MsgType | Y | The number indicating the message type. |
| 49 | SenderCompID | Y | The ID of the sender. |
| 56 | TargetCompID | Y | The ID of the intended recipient. |
| 34 | MsgSeqNum | Y | The message sequence number. |
| 52 | SendingTime | Y | The UTCTimestamp. For example, “19701231- 23:59:59.999”. |

## Standard Trailer

|  |  |  |  |
| --- | --- | --- | --- |
| **TAG** | **NAME** | **REQ** | **DESCRIPTION** |
| 10 | CheckSum | Y | Three byte checksum of the message. This is always the last tag in the message. |

# FIX Session Management

## Logon (A)

The first message of the session must be a Logon (35=A) message sent by the client. If the credentials are accepted, the server will respond with a Logon (35=A). If the credentials are invalid, the connection will be closed.

|  |  |  |  |
| --- | --- | --- | --- |
| **TAG** | **NAME** | **REQ** | **DESCRIPTION** |
|  | Header | Y | Standard header, with 35=A. |
| 108 | HeartBtInt | Y | The interval in seconds between heartbeats. The tag is always set to 30. |
| 141 | ResetSeqNumFlag | N | Supported values:    **N**​ = No  **Y**​ = Yes (reset sequence numbers) |
| 553 | Username | N\* | The user name on which the server and client agreed.  (\*) Not required. |
| 554 | Password | Y\* | The password assigned to the Username (553). (\*) Not required in a reply sent from the server. |
|  | Trailer | Y |  |

## Heartbeat (0)

A heartbeat should be sent before the timeout defined in the Logon (35=A) message is expired.

For example, the timeout is set to 30 seconds. A heartbeat must arrive before 29.999999 seconds are elapsed. If the last fraction of the second elapses, and the full 30-seconds period elapsed with no heartbeat messages, then the timeout occurs. In this case, the other side may close the connection, because the connection will be considered down.

|  |  |  |  |
| --- | --- | --- | --- |
| **TAG** | **NAME** | **REQUIRED** | **DESCRIPTION** |
|  | Header | Y | Standard header, with 35=0. |
|  | Trailer | Y |  |

## Logout (5)

Before closing the connection, the initiating side must send a Logout message (35=5), and then wait for a Logout message (35=5) sent in the reply before closing the connection.

|  |  |  |  |
| --- | --- | --- | --- |
| **TAG** | **NAME** | **REQUIRED** | **DESCRIPTION** |
|  | Header | Y | Standard header, with 35=5. |
| 58 | Text | N | Message |
|  | Trailer | Y |  |

FIX

## Security List Request (x)

The Security List Request (x) message is used to return a list of

Tag 559: 4 ‘All Securities’ is supported at the moment

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tag** | **Field Name** | | **FIX**  **Req’d** | **Update field set** | **Comments** |
|  | *Standard Header* | | Y |  | **MsgType = y (lowercase)** |
| 320 | | SecurityReqID | Y | Y |  |
| 559 | | SecurityListRequestType | Y | Y | Valid values:  ‘4’ All Pools |
|  | | *Standard Trailer* | Y |  |  |

## Market Data Request (V)

The client may send a Market Data Request (35=V) in order to subscribe to market data updates.

Servers may send market data in snapshots (35=W). To achieve the best network performance, the server should send a single snapshot on receiving a subscription request followed by incremental refreshes only. Servers may allow client-specific configurations to deviate from this behavior. Incremental refreshes (35=X) are not currently Supported.

Market data requests are reset when the client is disconnected. When the client is reconnected, the client must request the market data again.

|  |  |  |  |
| --- | --- | --- | --- |
| **TAG** | **NAME** | **REQ** | **DESCRIPTION** |
|  | Header | Y | Standard header, with 35=V. |
| 262 | MDReqID | Y | The ID of the Market Data Request. It must be new and unique unless 263=2 and refers to the previous market data request. |
| 263 | SubscriptionRequestType | Y | Supported values:     1. = Snapshot + Updates (e.g. “subscribe”) 2. = Stop updates (e.g. “unsubscribe”) |
| 264 | MarketDepth | Y | The book depth to report.    Supported values:     1. = Full book, unlimited depth [[1]](#footnote-1) 2. = Top of Book only |
| 265 | MDUpdateType | Y | The format of market data updates.    Supported values:     1. = Full refresh (e.g. snapshot) 2. = Incremental refresh (does not support at the moment) |
|  |  |  | *Begin Repeating Group* |
| 269 | MDEntryType | Y | Supported values:     1. = Bid 2. = Ask 3. = Trade |
|  |  |  | *End Repeating Group* |
| 146 | NoRelatedSym | Y | The number of symbols in the request. The tag must be always set to ​**1**​. |
|  |  |  | *Begin Repeating Group* |
| 55 | Symbol | Y | The pool symbol expressed as CCY1/CCY2. For example, “DOT/USDT”. |
|  |  |  | *End Repeating Group* |
|  | Trailer | Y |  |

This specification requires NoRelatedSym (146) to be set to 1. This means that the client can subscribe only to a single symbol in a FIX message.

The MDReqID defines a sort of domain for snapshots and updates of market data. Market Data Snapshots (35=W) sent for an MDReqID will effectively cancel all outstanding quotes sent for that MDReqID.

## Security List

The Security List (y) message is used to return a list of pools

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tag** | **Field Name** | | **FIX**  **Req’d** | **Update field set** | **Comments** |
|  | *Standard Header* | | Y |  | **MsgType = y** |
| 320 | | SecurityReqID | Y | Y |  |
| 322 | | SecurityResponseID | Y | Y | Identifier for the Security List (y) message |
| 560 | | SecurityRequestResult | Y | Y | The results returned to a Security Request (v) message  Valid values:  '0' Valid request  '1' Invalid or unsupported request  '2' No instruments found that match selection criteria  '3' Not authorized to retrieve instrument data  '4' Instrument data temporarily unavailable  '5' Request for instrument data not supported |
| *component block <Instrument>* | | | N | N |  |
|  | | *Standard Trailer* | Y |  |  |
|  | |  |  |  |  |

## Market Data Snapshot (W)

Market Data Snapshots (W) represent a full snapshot of the market for a given MDReqID (252). Each snapshot replaces all the market data related to that MDReqID, regardless of whether it was sent by a Market Data Snapshot or by a Market Data Incremental Refresh (X).

Market data within the snapshot is identified by MDEntryID (278). Previously published market data (such as quotes) that is not included into the snapshot was implicitly deleted. Similarly, previously publis

|  |  |  |  |
| --- | --- | --- | --- |
| **TAG** | **NAME** | **REQUIRED** | **DESCRIPTION** |
|  | Header | Y | Standard header, with 35=W. |
| 262 | MDReqID | Y | The ID of the Market Data Request that triggered sending this snapshot. |
| 55 | Symbol | Y | The pool expressed as CCY1/CCY2. For example, “DOT/USDT”. |
| 268 | NoMDEntries | Y | The number of market data entries in this message. |
|  |  |  | *Begin Repeating Group* |
| 269 | MDEntryType | Y | Supported values:     1. = Bid 2. = Ask 3. = Trade |
|  |  |  |  |
| 270 | MDEntryPx | Y | The market data entry price. |
| 271 | MDEntrySize | Y | The market data entry volume. |
| 64 | SettlDate | Y | The Settlement date expressed as YYYYMMDD. For example, “19701231” [[2]](#footnote-2) |
| 278 | MDEntryID | Y | The unique ID for this market data. |
|  |  |  | *End Repeating Group* |
|  | Trailer | Y |  |

## Market Data Request Reject (Y)

The server may respond to Market Data Requests (35=V) with a rejection (35=Y). For example, if an unrecognized currency pair was requested.

|  |  |  |  |
| --- | --- | --- | --- |
| **TAG** | **NAME** | **REQUIRED** | **DESCRIPTION** |
|  | Header | Y | Standard header, with 35=Y. |
| 262 | MDReqID | Y | The ID of the Market Data Request. |
| 281 | MDRecRejReason | Y | The reason for the rejection of the Market Data request. |
| 58 | Text | Y | Message. |
|  | Trailer | Y |  |

# Orders

## New Order Single (D)

|  |  |  |  |
| --- | --- | --- | --- |
| **TAG** | **NAME** | **REQUIRED** | **DESCRIPTION** |
|  | Header | Y | Standard header, with 35=D. |
| 11 | ClOrdID | Y | The client-assigned unique ID for the order. Notice that the maximum size of this tag is 58 bytes. |
| 55 | Symbol | Y | The pool name expressed as CCY1/CCY2. For example, “DOT/USDT”. |
| 15 | Currency | Y | The fixed currency of the trade, either CCY1 or CCY2. |
| 54 | Side | Y | Supported values:     1. = Buy 2. = Sell |
| 60 | TransactTime | Y | The UTCTimestamp when the order was sent by the client. |
| 38 | OrderQty | Y | The order quantity in the base currency (15). |
| 40 | OrdType | Y | The order type.    Supported values:     1. = Market 2. = Limit |
| 44 | Price | N\* | The price at which the limit order should be executed.  (\*) Required for all Limit OrdTypes (2) |
| 59 | TimeInForce | Y | Supported values:     1. = Immediate or Cancel (IOC) |
|  | Trailer | Y |  |

## Execution Report (8)

|  |  |  |  |
| --- | --- | --- | --- |
| **TAG** | **NAME** | **REQ** | **DESCRIPTION** |
|  | Header | Y | Standard header, with 35=8. |
| 37 | OrderId | Y | The unique ID for the order assigned by the server. |
| 11 | ClOrdID | Y | The unique ID of the order assigned by the client as specified in the ClOrdID (11) field in a New Order Single(35=D), Order Cancel Request (35=F) |
| 41 | OrigClOrdID | N\* | The unique client assigned ID of the original order that was subject to an Order Cancel Request (35=F)  (\*) Required in Execution Report sent in response to an Order Cancel Request (35=F) |
| 103 | OrdRejReason | N | Code to identify reason for order rejection.    For optional use with ExecType = 8 (Rejected) |
| 17 | ExecID | Y | The unique ID of the execution. |
| 150 | ExecType | Y | The type of execution that was reported.    Supported values:    0 = New   1. = Canceled 2. = Replace 3. = Pending Cancel   8 = Rejected  A = Pending New   1. = Pending Replace 2. = Trade (fill or partial fill) 3. = Order Status 4. = Trade in a Clearing Hold (reflects pending match message in EBS) |
| 39 | OrdStatus | Y | The current status of the order.    Supported values:     1. = New 2. = Partially Filled 3. = Filled   4 = Canceled  6 = Pending Cancel  8 = Rejected  A = Pending New  E = Pending Replace |
| 55 | Symbol | Y | Currency pair symbol expressed as CCY1/CCY2. For example, “DOT/USDT”. |
| 54 | Side | Y | Supported values:     1. = Buy 2. = Sell |
| 59 | TimeInForce | N | Supported values:   1. = Immediate or Cancel (IOC) |
| 38 | OrderQty | Y | The Quantity of the order specified in the units of the Currency (15). |
| 44 | Price | N\* | The price of the order.  (\*) Required if Price (44) was specified on the order. |
| 15 | Currency | N | The fixed currency of the trade.  This is the currency in which the Price (44) is specified. |
| 32 | LastQty | N\* | The quantity bought/sold on this (last) fill. (\*) Required if ExecType (150) = Trade (‘F’). |
| 31 | LastPx | N\* | Price of this (last) fill.  (\*) Required if ExecType (150) = Trade (‘F’). |
| 151 | LeavesQty | Y | The quantity remaining to be executed in the order specified in the units of the Currency (15). |
| 14 | CumQty | Y | The executed amount of the order specified in the units of the Currency (15). |
| 6 | AvgPx | N\* | The average price at which the order was executed.  (\*) Required if ExecType (150) = Trade (‘F’). |
| 60 | TransactTime | Y | The UTCTimestamp of the execution time on the server. |
| 58 | Text | N | Message |
|  | Trailer | Y |  |

## Order Cancel Request (F)

Cancel an existing order.

|  |  |  |  |
| --- | --- | --- | --- |
| **TAG** | **NAME** | **REQUIRED** | **DESCRIPTION** |
|  | Header | Y | Standard header, with 35=F. |
| 41 | OrigClOrdID | Y | The unique ID of the last non-cancelled order assigned by the client.  Notice that the maximum size of this tag is 58 bytes. |
| 37 | OrderId | N | The unique ID of the order, as returned to the client in the Execution Report in the OrderId field (37). |
| 11 | ClOrdID | Y | The unique ID of the cancel request as assigned by the Client.    Note: This is not the ClOrdID of the order being cancelled, but rather an identifier of the cancel request. |
| 55 | Symbol | Y | The same as in the original order. |
| 54 | Side | Y | The same as in the original order. |
| 60 | TransactTime | Y | UTCTimestamp of the request from the client. |
| 38 | OrderQty | N | The quantity being cancelled.  This must be the total remaining quantity in the order. |
|  | Trailer | Y |  |

## Order Cancel Reject (9)

|  |  |  |  |
| --- | --- | --- | --- |
| **TAG** | **NAME** | **REQ** | **DESCRIPTION** |
|  | Header | Y | Standard header, with 35=9. |
| 37 | OrderId | Y | The unique ID of the order, as returned to the client in the Execution Report in the OrderId field (37).    If the rejection occurred because of unknown order, this tag is set to NONE. |
| 11 | ClOrdID | Y | The unique ID of the order assigned by the client. This ID would have been the new ClOrdID.    When CxlRejResponseTo is “Order Cancel  Request”, this is identical to the ClOrdID (11) in the Order Cancel Request (35=F).    When CxlRejResponseTo is “Order Cancel/Replace Request”, this is identical to the ClOrdID (11) in the Order Cancel/Replace Request (35=F). |
| 41 | OrigClOrdID | Y | The ID of the order assigned by the client that could not be modified or canceled. |
| 39 | OrdStatus | Y | The order status after the Cancel Reject.    Note: If CxlRejReason is “Unknown order”, the  OrdStatus should be Rejected (8). |
| 434 | CxlRejResponseTo | Y | The type of request that has been rejected.    Supported values:     1. = Order Cancel Request 2. = Order Cancel/Replace Request |
| 102 | CxlRejReason | N | The reason for the rejection. |
| 58 | Text | N | Message |
|  | Trailer | Y |  |

1. The server may artificially limit the size of the book to limit resources spent by the priceCLOUD machine and improve performance. [↑](#footnote-ref-1)
2. Sent out with each Market Data Snapshot message. [↑](#footnote-ref-2)