

## Market Data Messages

- [MARKET\\_DATA\\_FEED\\_STATUS](#)
- [MARKET\\_DATA\\_FEED\\_SYMBOL\\_STATUS](#)
- [MARKET\\_DATA\\_REQUEST](#)
- [MARKET\\_DATA\\_REJECT](#)
- [MARKET\\_DATA\\_SNAPSHOT](#)
- [MARKET\\_DATA\\_UPDATE\\_TRADE](#)
- [MARKET\\_DATA\\_UPDATE\\_TRADE\\_WITH\\_UNBUNDLED\\_INDICATOR\\_2](#)
- [MARKET\\_DATA\\_UPDATE\\_TRADE\\_NO\\_TIMESTAMP](#)
- [MARKET\\_DATA\\_UPDATE\\_TRADE\\_COMPACT](#)
- [MARKET\\_DATA\\_UPDATE\\_LAST\\_TRADE\\_SNAPSHOT](#)
- [MARKET\\_DATA\\_UPDATE\\_BID\\_ASK](#)
- [MARKET\\_DATA\\_UPDATE\\_BID\\_ASK\\_NO\\_TIMESTAMP](#)
- [MARKET\\_DATA\\_UPDATE\\_BID\\_ASK\\_FLOAT\\_WITH\\_MICROSECONDS](#)
- [MARKET\\_DATA\\_UPDATE\\_SESSION\\_OPEN](#)
- [MARKET\\_DATA\\_UPDATE\\_SESSION\\_HIGH](#)
- [MARKET\\_DATA\\_UPDATE\\_SESSION\\_LOW](#)
- [MARKET\\_DATA\\_UPDATE\\_SESSION\\_SETTLEMENT](#)
- [MARKET\\_DATA\\_UPDATE\\_SESSION\\_VOLUME](#)
- [MARKET\\_DATA\\_UPDATE\\_OPEN\\_INTEREST](#)
- [MARKET\\_DATA\\_UPDATE\\_SESSION\\_NUM\\_TRADES](#)
- [MARKET\\_DATA\\_UPDATE\\_TRADING\\_SESSION\\_DATE](#)
- [MARKET\\_DEPTH\\_REQUEST](#)
- [MARKET\\_DEPTH\\_REJECT](#)
- [MARKET\\_DEPTH\\_SNAPSHOT\\_LEVEL](#)
- [MARKET\\_DEPTH\\_SNAPSHOT\\_LEVEL\\_FLOAT](#)
- [MARKET\\_DEPTH\\_UPDATE\\_LEVEL](#)
- [MARKET\\_DEPTH\\_UPDATE\\_LEVEL\\_FLOAT\\_WITH\\_MILLISECONDS](#)
- [MARKET\\_DEPTH\\_UPDATE\\_LEVEL\\_NO\\_TIMESTAMP](#)
- [TRADING\\_SYMBOL\\_STATUS](#)

---

### **MARKET\_DATA\_FEED\_STATUS [s\_MarketDataFeedStatus structure]** **Server >> Client**

---

The **MARKET\_DATA\_FEED\_STATUS** message is an optional message sent by the Server to indicate the overall status of the market data feed. This status applies to all symbols that have been subscribed to for market data.

Field Name	Field Description
------------	-------------------

<a href="#">[unsigned int16]</a> <b>Size</b>	The standard message size field. Automatically set by constructor.
<a href="#">[unsigned int16]</a> <b>Type</b>	The standard message type field. Automatically set by constructor.
<a href="#">[MarketDataFeedStatusEnum]</a> <b>Status</b>	<p>This can be set to <b>MARKET_DATA_FEED_UNAVAILABLE</b>, to indicate the market data feed is presently not available. Or it can be set to <b>MARKET_DATA_FEED_AVAILABLE</b>, to indicate the market data feed has been restored.</p> <p>Upon a connection to the server, <b>MARKET_DATA_FEED_AVAILABLE</b> is assumed to be the status. It is not until there has been expressly given <b>MARKET_DATA_FEED_UNAVAILABLE</b>, will the data feed be considered lost.</p>

### **MARKET\_DATA\_FEED\_SYMBOL\_STATUS [s\_MarketDataFeed SymbolStatus structure] Server >> Client**

The **MARKET\_DATA\_FEED\_SYMBOL\_STATUS** message is an optional message sent by the Server to indicate status of the market data feed for a particular symbol.

<b>Field Name</b>	<b>Field Description</b>
<a href="#">[unsigned int16]</a> <b>Size</b>	The standard message size field. Automatically set by constructor.
<a href="#">[unsigned int16]</a> <b>Type</b>	The standard message type field. Automatically set by constructor.

<p><a href="#">[MarketDataFeedStatusEnum]</a> <b>Status</b></p>	<p>This can be set to <b>MARKET_DATA_FEED_UNAVAILABLE</b>, to indicate the market data feed is presently not available for the symbol. Or it can be set to <b>MARKET_DATA_FEED_AVAILABLE</b>, to indicate the market data feed has been restored for the symbol.</p> <p>Upon a connection to the server, <b>MARKET_DATA_FEED_AVAILABLE</b> is assumed to be the status. It is not until there has been expressly given <b>MARKET_DATA_FEED_UNAVAILABLE</b>, will the market data feed for the symbol be considered lost.</p>
<p><a href="#">[unsigned int32]</a> <b>SymbolID</b></p>	<p>This is the same SymbolID sent by the Client in the <a href="#">MARKET_DATA_REQUEST</a> message which corresponds to the Symbol that the status in this message is for.</p>

## MARKET\_DATA\_REQUEST [s\_MarketDataRequest structure] Client >> Server

The [MARKET\\_DATA\\_REQUEST](#) message will subscribe to market data for a particular Symbol or request a market data snapshot.

The Server can also send market depth data in response to this message and not require a [MARKET\\_DEPTH\\_REQUEST](#).

Field Name	Field Description
<p><a href="#">[unsigned int16]</a> <b>Size</b></p>	<p>The standard message size field. Automatically set by constructor.</p>

<a href="#">[unsigned int16]</a> <b>Type</b>	The standard message type field. Automatically set by constructor.
<a href="#">[RequestActionEnum]</a> <b>RequestAction</b>	<p>This needs to be set to <b>SUBSCRIBE</b> to subscribe to market data for the Symbol from the Server. The Server will respond with an initial <a href="#">MARKET_DATA_SNAPSHOT</a> message and then provide <b>MARKET_DATA_UPDATE_*</b> updates as necessary.</p> <p>To unsubscribe, use <b>UNSUBSCRIBE</b>.</p> <p>To request only a <a href="#">MARKET_DATA_SNAPSHOT</a> message, set this to <b>SNAPSHOT</b>.</p>
<a href="#">[unsigned int32]</a> <b>SymbolID</b>	<p>This is the identifier which will be used in all of the market data response messages. This identifier is used so that the <b>Symbol</b> and <b>Exchange</b> do not have to be passed back in response messages from the Server. This makes the market data feed bandwidth efficient.</p> <p>If the Server receives a <a href="#">MARKET_DATA_REQUEST</a> for a Symbol and Exchange to subscribe to data for, that is currently subscribed to and this SymbolID is different, then the Server needs to reject it.</p>
<a href="#">[char]</a> <b>Symbol</b>	The Symbol that market data is requested for. Not set when unsubscribing.
<a href="#">[char]</a> <b>Exchange</b>	Optional Exchange. Not set when unsubscribing.

**MARKET\_DATA\_REJECT** [**s\_MarketDataReject** structure] **Server >> Client**

The [MARKET\\_DATA\\_REJECT](#) message is sent by the Server to the Client to reject a [MARKET\\_DATA\\_REQUEST](#) message for any reason.

Field Name	Field Description
<a href="#">[unsigned int16]</a> <b>Size</b>	The standard message size field. Automatically set by constructor.
<a href="#">[unsigned int16]</a> <b>Type</b>	The standard message type field. Automatically set by constructor.
<a href="#">[unsigned int32]</a> <b>SymbolID</b>	This is the same SymbolID sent by the Client in the <a href="#">MARKET_DATA_REQUEST</a> message which corresponds to the Symbol that the data in this message is for.
<a href="#">[char]</a> <b>RejectText</b>	Free-form text explaining the reason for the reject.

## **MARKET\_DATA\_SNAPSHOT [s\_MarketDataSnapshot structure] Server >> Client**

The Server sends the [MARKET\\_DATA\\_SNAPSHOT](#) message to the Client immediately after a successful [MARKET\\_DATA\\_REQUEST](#) message has been received from the Client and it has indicated to subscribe to the symbol or requested the snapshot of data.

Any changes to the data fields within the **MARKET\_DATA\_SNAPSHOT** message during the trading session will be sent by the Server to the Client through the corresponding **MARKET\_DATA\_UPDATE\_\*** messages.

It is recommended that the **MARKET\_DATA\_SNAPSHOT** be sent by the Server at the start of a new trading session.

This message can be sent more often, however it is not intended to be sent frequently.

This message type does not signify a trade has occurred. It should never be interpreted by the Client in that way.

There is no need to send this when there is a new High or Low during the trading session. The Server should use the [MARKET\\_DATA\\_UPDATE\\_SESSION\\_HIGH](#) or [MARKET\\_DATA\\_UPDATE\\_SESSION\\_LOW](#) messages instead.

Field Name	Field Description
<a href="#">[unsigned int16]</a> <b>Size</b>	The standard message size field. Automatically set by constructor.
<a href="#">[unsigned int16]</a> <b>Type</b>	The standard message type field. Automatically set by constructor.
<a href="#">[unsigned int32]</a> <b>SymbolID</b>	This is the same SymbolID sent by the Client in the <a href="#">MARKET_DATA_REQUEST</a> message which corresponds to the Symbol that the data in this message is for.
<a href="#">[double]</a> <b>SessionSettlementPrice</b>	<p>The previous Settlement price when this message is sent before the market closes for the trading session. After the market has closed, this is the most recent Settlement price.</p> <p>For binary encoding, if this field is not set it needs to be set to DBL_MAX. Refer to <a href="#">Unset Message Fields</a>.</p>
<a href="#">[double]</a> <b>SessionOpenPrice</b>	<p>The Opening price for the trading session.</p> <p>For binary encoding, if this field is not set it needs to be set to DBL_MAX. Refer to <a href="#">Unset Message Fields</a>.</p>

<p><a href="#">[double]</a> <b>SessionHighPrice</b></p>	<p>The the High price for the trading session.</p> <p>For binary encoding, if this field is not set it needs to be set to DBL_MAX. Refer to <a href="#">Unset Message Fields</a>.</p>
<p><a href="#">[double]</a> <b>SessionLowPrice</b></p>	<p>The Low price for the trading session.</p> <p>For binary encoding, if this field is not set it needs to be set to DBL_MAX. Refer to <a href="#">Unset Message Fields</a>.</p>
<p><a href="#">[double]</a> <b>SessionVolume</b></p>	<p>The total Volume for the trading session.</p> <p>For binary encoding, if this field is not set it needs to be set to DBL_MAX. Refer to <a href="#">Unset Message Fields</a>.</p>
<p><a href="#">[unsigned int32]</a> <b>SessionNumTrades</b></p>	<p>The number of Trades for the trading session.</p> <p>For binary encoding, if this field is not set it needs to be set to UINT_MAX. Refer to <a href="#">Unset Message Fields</a>.</p>
<p><a href="#">[unsigned int32]</a> <b>OpenInterest</b></p>	<p>Contains the Open Interest for futures and options.</p> <p>For binary encoding, if this field is not set it needs to be set to UINT_MAX. Refer to <a href="#">Unset Message Fields</a>.</p>

<a href="#">[double]</a> <b>BidPrice</b>	<p>The latest best Bid price.</p> <p>For binary encoding, if this field is not set it needs to be set to DBL_MAX. Refer to <a href="#">Unset Message Fields</a>.</p>
<a href="#">[double]</a> <b>AskPrice</b>	<p>The latest best Ask price.</p> <p>For binary encoding, if this field is not set it needs to be set to DBL_MAX. Refer to <a href="#">Unset Message Fields</a>.</p>
<a href="#">[double]</a> <b>AskQuantity</b>	<p>The quantity of the orders at the Ask price.</p> <p>For binary encoding, if this field is not set it needs to be set to DBL_MAX. Refer to <a href="#">Unset Message Fields</a>.</p>
<a href="#">[double]</a> <b>BidQuantity</b>	<p>The quantity of the orders at the Bid price.</p> <p>For binary encoding, if this field is not set it needs to be set to DBL_MAX. Refer to <a href="#">Unset Message Fields</a>.</p>
<a href="#">[double]</a> <b>LastTradePrice</b>	<p>The most recent last trade price.</p> <p>For binary encoding, if this field is not set it needs to be set to DBL_MAX. Refer to <a href="#">Unset Message Fields</a>.</p>



<a href="#">[double]</a> <b>LastTradeVolume</b>	<p>The quantity/volume of the most recent last trade.</p> <p>For binary encoding, if this field is not set it needs to be set to DBL_MAX. Refer to <a href="#">Unset Message Fields</a>.</p>
<a href="#">[t_DateTimeWithMilliseconds]</a> <b>LastTradeDateTime</b>	The Date-Time of the last trade.
<a href="#">[t_DateTimeWithMilliseconds]</a> <b>BidAskDateTime</b>	The Date-Time of the last Bid and Ask quote data update.
<a href="#">[t_DateTime4Byte]</a> <b>SessionSettlementDateTime</b>	<p>The trading date the Settlement price is for. The time component is not normally considered relevant for this field.</p> <p>This field will be 0 if this field is not available from the data feed.</p>
<a href="#">[t_DateTime4Byte]</a> <b>TradingSessionDate</b>	<p>This is the Date of the trading session that the data contained in this snapshot message is for.</p> <p>The time component is not normally considered relevant for this field.</p>

### MARKET\_DATA\_UPDATE\_TRADE [s\_MarketDataUpdateTrade structure] Server >> Client

The Server sends this market data feed message to the Client when a trade occurs.

Field Name	Field Description

<a href="#">[unsigned int16]</a> <b>Size</b>	The standard message size field. Automatically set by constructor.
<a href="#">[unsigned int16]</a> <b>Type</b>	The standard message type field. Automatically set by constructor.
<a href="#">[unsigned int32]</a> <b>SymbolID</b>	This is the same SymbolID sent by the Client in the <a href="#">MARKET_DATA_REQUEST</a> message which corresponds to the Symbol that the data in this message is for.
<a href="#">[AtBidOrAskEnum]</a> <b>AtBidOrAsk</b>	Indicator whether the trade occurred at the bid or ask.
<a href="#">[double]</a> <b>Price</b>	The price of the trade.
<a href="#">[double]</a> <b>Volume</b>	The volume of the trade.
<a href="#">[t_DateTimeWithMilliseconds]</a> <b>DateTime</b>	The Date-Time of the trade.

**MARKET\_DATA\_UPDATE\_TRADE\_WITH\_UNBUNDLED\_INDICATOR\_2**  
**[s\_MarketDataUpdateTradeWithUnbundledIndicator2 structure] Server**  
**>> Client**

Sent by the Server to the Client when a trade occurs. This message has additional fields as compared to the [MARKET\\_DATA\\_UPDATE\\_TRADE](#) message and also supports microsecond time stamping.

Field Name	Field Description
------------	-------------------

<a href="#">[unsigned int16]</a> <b>Size</b>	The standard message size field. Automati
<a href="#">[unsigned int16]</a> <b>Type</b>	The standard message type field. Automat
<a href="#">[unsigned int32]</a> <b>SymbolID</b>	This is the same SymbolID : <a href="#">MARKET_DATA_REQUEST</a> message wh the data in this message is for.
<a href="#">[float]</a> <b>Price</b>	The price of the trade.
<a href="#">[unsigned int32]</a> <b>Volume</b>	The volume of the trade.
<a href="#">[t_DateTimeWithMicrosecondsInt]</a> <b>DateTime</b>	The timestamp of the trade in UNIX micros
<a href="#">[AtBidOrAskEnum8]</a> <b>AtBidOrAsk</b>	Indicator whether the trade occurred at the
<a href="#">[UnbundledTradeIndicatorEnum]</a> <b>UnbundledTradeIndicator</b>	<p>This is an optional field.</p> <p>For symbols on exchanges which support part of a larger summary trade, this mess the trade is part of a larger summary t individual trade.</p> <p>The possible values are listed below.</p> <ul style="list-style-type: none"> <li>• UNBUNDLED_TRADE_NONE =</li> <li>• FIRST_SUB_TRADE_OF_UNBL</li> <li>• LAST_SUB_TRADE_OF_UNBU</li> </ul>

<a href="#">[TradeConditionEnum]</a> <b>TradeCondition</b>	<p>This is an optional field. It usually applies to</p> <p>This field indicates a special condition whose values are listed below.</p> <ul style="list-style-type: none"> <li>• TRADE_CONDITION_NONE = 0</li> <li>• TRADE_CONDITION_NON_LAST = 1</li> <li>• TRADE_CONDITION_ODD_LOT = 2</li> </ul>
---	--

## MARKET\_DATA\_UPDATE\_TRADE\_NO\_TIMESTAMP [s\_MarketDataUpdateTradeNoTimestamp structure] Server >> Client

This message is optional.

Sent by the Server to the Client when a trade occurs. This message is identical to the [MARKET\\_DATA\\_UPDATE\\_TRADE\\_WITH\\_UNBUNDLED\\_INDICATOR\\_2](#) message except it does not have a timestamp. It needs to be sent when there is no change with the timestamp for the trade as compared to the prior trade.

When the Server sends this message to the Client, the Client needs to use the prior received trade timestamp to know what the timestamp is for this message.

Field Name	Field Description
<a href="#">[unsigned int16]</a> <b>Size</b>	The standard message size field. Automatically set by the Server.
<a href="#">[unsigned int16]</a> <b>Type</b>	The standard message type field. Automatically set by the Server.
<a href="#">[unsigned int32]</a> <b>SymbolID</b>	This is the same SymbolID as the <a href="#">MARKET_DATA_REQUEST</a> message which the data in this message is for.
<a href="#">[float]</a> <b>Price</b>	The price of the trade.

<a href="#">[unsigned int32]</a> <b>Volume</b>	The volume of the trade.
<a href="#">[AtBidOrAskEnum8]</a> <b>AtBidOrAsk</b>	Indicator whether the trade occurred at the E
<a href="#">[UnbundledTradeIndicatorEnum]</a> <b>UnbundledTradeIndicator</b>	<p>This is an optional field.</p> <p>For symbols on exchanges which support r part of a larger summary trade, this messag the trade is part of a larger summary tra individual trade.</p> <p>The possible values are listed below.</p> <ul style="list-style-type: none"> <li>• UNBUNDLED_TRADE_NONE = 0</li> <li>• FIRST_SUB_TRADE_OF_UNBUN</li> <li>• LAST_SUB_TRADE_OF_UNBUN</li> </ul>
<a href="#">[TradeConditionEnum]</a> <b>TradeCondition</b>	<p>This is an optional field. It usually applies to :</p> <p>This field indicates a special condition which values are listed below.</p> <ul style="list-style-type: none"> <li>• TRADE_CONDITION_NONE = 0</li> <li>• TRADE_CONDITION_NON_LAST = 1</li> <li>• TRADE_CONDITION_ODD_LOT_</li> </ul>

## MARKET\_DATA\_UPDATE\_TRADE\_COMPACT

[\[s\\_MarketDataUpdateTradeCompact structure\]](#) Server >> Client

Sent by the Server to the Client when a trade occurs. This message is a more compact [MARKET\\_DATA\\_UPDATE\\_TRADE](#). For the price it uses a 4 byte float.

Field Name	Field Description

<a href="#">[unsigned int16]</a> <b>Size</b>	The standard message size field. Automatically set by constructor.
<a href="#">[unsigned int16]</a> <b>Type</b>	The standard message type field. Automatically set by constructor.
<a href="#">[float]</a> <b>Price</b>	The price of the trade.
<a href="#">[float]</a> <b>Volume</b>	The volume of the trade.
<a href="#">[t_DateTime4Byte]</a> <b>DateTime</b>	The timestamp of the trade in UNIX time format. This does not contain the milliseconds for compactness.
<a href="#">[unsigned int32]</a> <b>SymbolID</b>	This is the same SymbolID sent by the Client in the <a href="#">MARKET DATA REQUEST</a> message which corresponds to the Symbol that the data in this message is for.
<a href="#">[AtBidOrAskEnum]</a> <b>AtBidOrAsk</b>	Indicator whether the trade occurred at the Bid or Ask price.

## **MARKET\_DATA\_UPDATE\_LAST\_TRADE\_SNAPSHOT**

**[s\_MarketDataUpdateLastTradeSnapshot structure] Server >> Client**

Sent by the Server to the Client to update the last trade price, volume and date-time fields under conditions when there is not a trade.

This message type does not signify a trade has occurred. It should never be interpreted by the Client in that way.

Field Name	Field Description
<a href="#">[unsigned int16]</a> <b>Size</b>	The standard message size field. Automatically set by constructor.
<a href="#">[unsigned int16]</a> <b>Type</b>	The standard message type field. Automatically set by constructor.
<a href="#">[unsigned int32]</a> <b>SymbolID</b>	This is the same SymbolID sent by the Client in the <a href="#">MARKET_DATA_REQUEST</a> message which corresponds to the Symbol that the data in this message is for.
<a href="#">[double]</a> <b>LastTradePrice</b>	The most recent last trade price.
<a href="#">[double]</a> <b>LastTradeVolume</b>	The quantity/volume of the most recent last trade.
<a href="#">[t_DateTimeWithMilliseconds]</a> <b>LastTradeDateTime</b>	The Date-Time of the last trade.

### MARKET\_DATA\_UPDATE\_BID\_ASK [s\_MarketDataUpdateBidAsk structure] Server >> Client

The Server sends this market data feed message to the Client when the best bid or ask price or size changes.

Field Name	Field Description

<a href="#">[unsigned int16]</a> <b>Size</b>	The standard message size field. Automatically set by constructor.
<a href="#">[unsigned int16]</a> <b>Type</b>	The standard message type field. Automatically set by constructor.
<a href="#">[unsigned int32]</a> <b>SymbolID</b>	This is the same SymbolID sent by the Client in the <a href="#">MARKET_DATA_REQUEST</a> message which corresponds to the Symbol that the data in this message is for.
<a href="#">[double]</a> <b>BidPrice</b>	The current Bid price. Leave unset if there is no price available.
<a href="#">[float]</a> <b>BidQuantity</b>	The current number of contracts/shares at the bid price.
<a href="#">[double]</a> <b>AskPrice</b>	The current ask or offer price. Leave unset if there is no price available.
<a href="#">[float]</a> <b>AskQuantity</b>	The current number of contracts/shares at the ask price.
<a href="#">[t_DateTime4Byte]</a> <b>DateTime</b>	The Date-Time of the Bid and Ask update.

### **MARKET\_DATA\_UPDATE\_BID\_ASK\_NO\_TIMESTAMP** [s\_MarketDataUpdateBidAskNoTimeStamp structure] Server >> Client

This message is optional.

Sent by the Server to the Client when there is an update to the Bid Ask prices and/or quantities. This message is identical to the [MARKET\\_DATA\\_UPDATE\\_BID\\_ASK](#) message except it does



not have a timestamp. It needs to be sent when there is no change with the timestamp for the Bid Ask update as compared to the prior update.

When the Server sends this message to the Client, the Client needs to use the prior received Bid Ask update timestamp to know what the timestamp is for this message.

Field Name	Field Description
<a href="#">[unsigned int16]</a> <b>Size</b>	The standard message size field. Automatically set by constructor.
<a href="#">[unsigned int16]</a> <b>Type</b>	The standard message type field. Automatically set by constructor.
<a href="#">[unsigned int32]</a> <b>SymbolID</b>	This is the same SymbolID sent by the Client in the <a href="#">MARKET_DATA_REQUEST</a> message which corresponds to the Symbol that the data in this message is for.
<a href="#">[float]</a> <b>BidPrice</b>	The current Bid price. Leave unset if there is no price available.
<a href="#">[unsigned int32]</a> <b>BidQuantity</b>	The current number of contracts/shares at the Bid price.
<a href="#">[float]</a> <b>AskPrice</b>	The current Ask or offer price. Leave unset if there is no price available.
<a href="#">[unsigned int32]</a> <b>AskQuantity</b>	The current number of contracts/shares at the Ask price.

**MARKET\_DATA\_UPDATE\_BID\_ASK\_FLOAT\_WITH\_MICROSECONDS**  
[s\_MarketDataUpdateBidAskFloatWithMicroseconds structure] Server

## >> Client

---

This message is optional.

Sent by the Server to the Client when there is an update to the Bid Ask prices and/or quantities. This message is identical to the [MARKET\\_DATA\\_UPDATE\\_BID\\_ASK](#) message except it does not have a timestamp. It needs to be sent when there is no change with the timestamp for the Bid Ask update as compared to the prior update.

When the Server sends this message to the Client, the Client needs to use the prior received Bid Ask update timestamp to know what the timestamp is for this message.

Field Name	Field Description
<a href="#">[unsigned int16]</a> <b>Size</b>	The standard message size field. Automatically set by constructor.
<a href="#">[unsigned int16]</a> <b>Type</b>	The standard message type field. Automatically set by constructor.
<a href="#">[unsigned int32]</a> <b>SymbolID</b>	This is the same SymbolID sent by the Client in the <a href="#">MARKET_DATA_REQUEST</a> message which corresponds to the Symbol that the data in this message is for.
<a href="#">[float]</a> <b>BidPrice</b>	The current Bid price. Leave unset if there is no price available.
<a href="#">[float]</a> <b>BidQuantity</b>	The current number of contracts/shares at the bid price.

<a href="#">[float]</a> <b>AskPrice</b>	The current ask or offer price. Leave unset if there is no price available.
<a href="#">[float]</a> <b>AskQuantity</b>	The current number of contracts/shares at the ask price.
<a href="#">[t_DateTimeWithMicrosecondsInt]</a> <b>DateTime</b>	The timestamp of the trade in UNIX microseconds time format.

## MARKET\_DATA\_UPDATE\_SESSION\_OPEN

[s\_MarketDataUpdateSessionOpen structure] Server >> Client

Sent by the Server to the Client to update the session Open.

Field Name	Field Description
<a href="#">[unsigned int16]</a> <b>Size</b>	The standard message size field. Automatically set by constructor.
<a href="#">[unsigned int16]</a> <b>Type</b>	The standard message type field. Automatically set by constructor.
<a href="#">[unsigned int32]</a> <b>SymbolID</b>	This is the same SymbolID sent by the Client in the <a href="#">MARKET_DATA_REQUEST</a> message which corresponds to the Symbol that the data in this message is for.
<a href="#">[double]</a> <b>Price</b>	The session Open price.

## MARKET\_DATA\_UPDATE\_SESSION\_HIGH

### [s\_MarketDataUpdateSessionHigh structure] Server >> Client

---

Sent by the Server to the Client to update the session High as the High price changes throughout the session.

Field Name	Field Description
<a href="#">[unsigned int16]</a> <b>Size</b>	The standard message size field. Automatically set by constructor.
<a href="#">[unsigned int16]</a> <b>Type</b>	The standard message type field. Automatically set by constructor.
<a href="#">[unsigned int32]</a> <b>SymbolID</b>	This is the same SymbolID sent by the Client in the <a href="#">MARKET_DATA_REQUEST</a> message which corresponds to the Symbol that the data in this message is for.
<a href="#">[double]</a> <b>Price</b>	The session High price.

## MARKET\_DATA\_UPDATE\_SESSION\_LOW

### [s\_MarketDataUpdateSessionLow structure] Server >> Client

---

Sent by the Server to the Client to update the session Low as the Low price changes throughout the session.

Field Name	Field Description
<a href="#">[unsigned int16]</a> <b>Size</b>	The standard message size field. Automatically set by constructor.

<a href="#">[unsigned int16]</a> <b>Type</b>	The standard message type field. Automatically set by constructor.
<a href="#">[unsigned int32]</a> <b>SymbolID</b>	This is the same SymbolID sent by the Client in the <a href="#">MARKET_DATA_REQUEST</a> message which corresponds to the Symbol that the data in this message is for.
<a href="#">[double]</a> <b>Price</b>	The session Low price.

## MARKET\_DATA\_UPDATE\_SESSION\_SETTLEMENT

[s\_MarketDataUpdateSessionSettlement structure] Server >> Client

Sent by the Server to the Client to update the session settlement price when the session settlement price changes.

Field Name	Field Description
<a href="#">[unsigned int16]</a> <b>Size</b>	The standard message size field. Automatically set by constructor.
<a href="#">[unsigned int16]</a> <b>Type</b>	The standard message type field. Automatically set by constructor.
<a href="#">[unsigned int32]</a> <b>SymbolID</b>	This is the same SymbolID sent by the Client in the <a href="#">MARKET_DATA_REQUEST</a> message which corresponds to the Symbol that the data in this message is for.
<a href="#">[double]</a> <b>Price</b>	The settlement price.

[\[t\\_DateTime4Byte\]](#)  
**DateTime**

That trading date the settlement price is for. The time component is not normally considered relevant in this case.

## **MARKET\_DATA\_UPDATE\_SESSION\_VOLUME** **[s\_MarketDataUpdateSessionVolume structure] Server >> Client**

Sent by the Server to the Client when the session trade Volume needs to be updated.

The recommended rule for the Server to notify the Client of a change with the session trade volume to maintain bandwidth efficiency, is as follows: When a trade occurs for a symbol subscribed to, the Server will send a [MARKET\\_DATA\\_UPDATE\\_TRADE](#) message to the Client. The Client should then increment its session trade volume value for the symbol by the value in the **Volume** field in this message.

The Server will assume the Client is doing this. Therefore, when a trade occurs and the session trade volume does not equal the prior session trade volume plus the Volume for the most recent trade sent to the Client, then the Server must send out a

**MARKET\_DATA\_UPDATE\_SESSION\_VOLUME** message to the client since the client calculation of the session trade volume is no longer correct.

It is assumed that the reason for this inconsistency is due to trades included within the session trade volume which have not been sent out as normal trades.

The Server should also send this message out at the frequency that the Server determines, such as every minute if there also has been a trade at that time.

Field Name	Field Description
<a href="#">[unsigned int16]</a> <b>Size</b>	The standard message size field. Automatically set by constructor.
<a href="#">[unsigned int16]</a> <b>Type</b>	The standard message type field. Automatically set by constructor.

<a href="#">[unsigned int32]</a> <b>SymbolID</b>	This is the same SymbolID sent by the Client in the <a href="#">MARKET_DATA_REQUEST</a> message which corresponds to the Symbol that the data in this message is for.
<a href="#">[double]</a> <b>Volume</b>	The total Volume for the session.

### **MARKET\_DATA\_UPDATE\_OPEN\_INTEREST** **[s\_MarketDataUpdateOpenInterest structure] Server >> Client**

The [MARKET\\_DATA\\_UPDATE\\_OPEN\\_INTEREST](#) message is sent by the Server to the Client to update the OpenInterest field previously sent through the [MARKET\\_DATA\\_SNAPSHOT](#) message.

<b>Field Name</b>	<b>Field Description</b>
<a href="#">[unsigned int16]</a> <b>Size</b>	The standard message size field. Automatically set by constructor.
<a href="#">[unsigned int16]</a> <b>Type</b>	The standard message type field. Automatically set by constructor.
<a href="#">[unsigned int32]</a> <b>SymbolID</b>	This is the same SymbolID sent by the Client in the <a href="#">MARKET_DATA_REQUEST</a> message which corresponds to the Symbol that the data in this message is for.
<a href="#">[unsigned int32]</a> <b>OpenInterest</b>	The open interest for the symbol.

### **MARKET\_DATA\_UPDATE\_SESSION\_NUM\_TRADES** **[s\_MarketDataUpdateSessionNumTrades structure] Server >> Client**

Sent by the Server to the Client to update the trading session number of trades.

Field Name	Field Description
<a href="#">[unsigned int16]</a> <b>Size</b>	The standard message size field. Automatically set by constructor.
<a href="#">[unsigned int16]</a> <b>Type</b>	The standard message type field. Automatically set by constructor.
<a href="#">[unsigned int32]</a> <b>SymbolID</b>	This is the same SymbolID sent by the Client in the <a href="#">MARKET_DATA_REQUEST</a> message which corresponds to the Symbol that the data in this message is for.
<a href="#">[int32]</a> <b>NumTrades</b>	The number of trades which have occurred during the current trading session.

## MARKET\_DATA\_UPDATE\_TRADING\_SESSION\_DATE

[s\_MarketDataUpdateTradingSessionDate structure] Server >> Client

Sent by the Server to the Client to update the trading session Date.

Field Name	Field Description
<a href="#">[unsigned int16]</a> <b>Size</b>	The standard message size field. Automatically set by constructor.
<a href="#">[unsigned int16]</a> <b>Type</b>	The standard message type field. Automatically set by constructor.



<a href="#">[unsigned int32]</a> <b>SymbolID</b>	This is the same SymbolID sent by the Client in the <a href="#">MARKET_DATA_REQUEST</a> message which corresponds to the Symbol that the data in this message is for.
<a href="#">[t_DateTime4Byte]</a> <b>Date</b>	The date of the current trading session. The time component is not normally considered relevant in this case.

### MARKET\_DEPTH\_REQUEST [s\_MarketDepthRequest structure] Client >> Server

Field Name	Field Description
<a href="#">[unsigned int16]</a> <b>Size</b>	The standard message size field. Automatically set by constructor.
<a href="#">[unsigned int16]</a> <b>Type</b>	The standard message type field. Automatically set by constructor.
<a href="#">[RequestActionEnum]</a> <b>RequestAction</b>	<p>This needs to be set to <b>SUBSCRIBE</b> to subscribe to market data for the Symbol from the Server. The server will respond with an initial <a href="#">MARKET_DEPTH_SNAPSHOT_LEVEL</a> message and then provide <a href="#">MARKET_DEPTH_UPDATE_LEVEL</a> updates as necessary.</p> <p>To unsubscribe, use <b>UNSUBSCRIBE</b>.</p>

<a href="#">[unsigned int32]</a> <b>SymbolID</b>	<p>This is the identifier which will be used in all of the market depth data response messages.</p> <p>This SymbolID can be the same as the one used in the <a href="#">MARKET_DATA_REQUEST</a> message for the same Symbol and Exchange.</p> <p>This identifier is used so that the Symbol does not have to be passed back in response messages from the Server. If the Server receives a <a href="#">MARKET_DEPTH_REQUEST</a> for a Symbol and Exchange to subscribe to market depth data for, that is currently subscribed to and this SymbolID is different, then the Server should reject it.</p>
<a href="#">[char]</a> <b>Symbol</b>	The symbol for the market depth request. Not set when unsubscribing.
<a href="#">[char]</a> <b>Exchange</b>	The optional exchange for the symbol. Not set when unsubscribing.
<a href="#">[int32]</a> <b>NumLevels</b>	Number of depth levels requested. Not set when unsubscribing.

## **MARKET\_DEPTH\_REJECT [s\_MarketDepthReject structure] Server >> Client**

The **MARKET\_DEPTH\_REJECT** message is sent by the Server to the Client to reject a [MARKET\\_DEPTH\\_REQUEST](#) message for any reason.

<b>Field Name</b>	<b>Field Description</b>
<a href="#">[unsigned int16]</a> <b>Size</b>	The standard message size field. Automatically set by constructor.

<a href="#">[unsigned int16]</a> <b>Type</b>	The standard message type field. Automatically set by constructor.
<a href="#">[unsigned int32]</a> <b>SymbolID</b>	This is the same SymbolID sent by the Client in the <a href="#">MARKET_DEPTH_REQUEST</a> message which corresponds to the Symbol that the data in this message is for.
<a href="#">[char]</a> <b>RejectText</b>	Free-form text explaining the reason for the reject.

### **MARKET\_DEPTH\_SNAPSHOT\_LEVEL [s\_MarketDepthSnapshotLevel structure] Server >> Client**

This is a message sent by Server to provide the initial market depth data entries to the Client after the Client subscribes to market data or separately subscribes to market depth data. The Client will need to separately subscribe to market depth data if the Server requires it.

Each message provides a single entry of depth data. Therefore, the Server will send multiple **MARKET\_DEPTH\_SNAPSHOT\_LEVEL** messages in a series in order for the Client to build up its initial market depth book.

The first message will be identified by the **IsFirstMessageInBatch** field being set to 1. The last message will be identified by the **IsLastMessageInBatch** field being set to 1.

In the case where the market depth book is empty, the Server still needs to send through one single message with the **SymbolID** set, **IsFirstMessageInBatch** equal to 1 and **IsLastMessageInBatch** equal to 1. All other members will be at the default values. The Client will understand this as an empty book.

<b>Field Name</b>	<b>Field Description</b>
<a href="#">[unsigned int16]</a> <b>Size</b>	The standard message size field. Automatically set constructor.
<a href="#">[unsigned int16]</a> <b>Type</b>	The standard message type field. Automatically set constructor.

<a href="#">[unsigned int32]</a> <b>SymbolID</b>	This is the same SymbolID sent by the Client in <a href="#">MARKET_DATA_REQUEST/MARKET_DEPTH_REQUEST</a> message which corresponds to the Symbol that the data in this message is for.
<a href="#">[AtBidOrAskEnum]</a> <b>Side</b>	Set to AT_BID = 1 if this is a bid side market depth entry. Set to AT_ASK = 2, if this is an ask side market depth entry.
<a href="#">[double]</a> <b>Price</b>	This is the price of the market depth entry.
<a href="#">[double]</a> <b>Quantity</b>	This is the quantity of orders at the Price.
<a href="#">[unsigned int16]</a> <b>Level</b>	This indicates the level of the price within the market depth book. The minimum value is 1. There is no maximum value. A value of 1 is considered the best bid or ask data.
<a href="#">[unsigned int8]</a> <b>IsFirstMessageInBatch</b>	Set to 1 if this is the first message in the batch of messages.
<a href="#">[unsigned int8]</a> <b>IsLastMessageInBatch</b>	Set to 1 if this is the last message in a batch of messages. If there is only a single message to be sent, in case the market depth book is empty, then IsFirstMessageInBatch will equal 1 and IsLastMessageInBatch will equal 1.

## MARKET\_DEPTH\_SNAPSHOT\_LEVEL\_FLOAT

[\[s\\_MarketDepthSnapshotLevelFloat structure\]](#) **Server >> Client**

This is a message sent by Server to provide the initial market depth data entries to the Client after the Client subscribes to market data or separately subscribes to market depth data. The Client will need to separately subscribe to market depth data if the Server requires it.

Each message provides a single entry of depth data. Therefore, the Server will send multiple **MARKET\_DEPTH\_SNAPSHOT\_LEVEL\_FLOAT** messages in a series in order for the Client to build up its initial market depth book.

The first message will be identified by the **IsFirstMessageInBatch** field being set to 1. The last message will be identified by the **IsLastMessageInBatch** field being set to 1.

In the case where the market depth book is empty, the Server still needs to send through one single message with the **SymbolID** set, **IsFirstMessageInBatch** equal to 1 and **IsLastMessageInBatch** equal to 1. All other members will be at the default values. The Client will understand this as an empty book.

Field Name	Field Description
<a href="#">[unsigned int16]</a> <b>Size</b>	The standard message size field. Automatically constructor.
<a href="#">[unsigned int16]</a> <b>Type</b>	The standard message type field. Automatically constructor.
<a href="#">[unsigned int32]</a> <b>SymbolID</b>	This is the same SymbolID sent by the Client <a href="#">MARKET_DATA_REQUEST/MARKET_DEPTH_RE</a> message which corresponds to the Symbol that the this message is for.
<a href="#">[AtBidOrAskEnum8]</a> <b>Side</b>	Set to AT_BID = 1 if this is a bid side market depth Set to AT_ASK = 2, if this is an ask side market entry.
<a href="#">[float]</a> <b>Price</b>	This is the price of the market depth entry.
<a href="#">[float]</a> <b>Quantity</b>	This is the quantity of orders at the Price.

<a href="#">[FinalUpdateInBatchEnum]</a> <b>FinalUpdateInBatch</b>	An indicator whether this is the final message or r batch of updates.
<a href="#">[uint32_t]</a> <b>NumOrders</b>	The number of orders at the <b>Price</b> .
<a href="#">[unsigned int16]</a> <b>Level</b>	This indicates the level of the price within the marke book. The minimum value is 1. There is no maximum A value of 1 is considered the best bid or ask data.

### **MARKET\_DEPTH\_UPDATE\_LEVEL [s\_MarketDepthUpdateLevel structure] Server >> Client**

Sent by the Server to the Client to **Update/Insert** or **Delete** a particular market depth price level in the market depth book maintained by the Client.

Each **MARKET\_DEPTH\_UPDATE\_LEVEL** message updates one level of market depth on one side. An insert/update/delete model is used for market depth.

The Client will need to determine the based upon the price, what particular market depth level is being updated, inserted or deleted.

It is for this reason, that an insert/update is considered as one update type since it is possible to determine whether it is an insert or update based upon the existence of the price level in the existing market depth book on the Client side.

What this means is that when the **UpdateType** field is **MARKET\_DEPTH\_INSERT\_UPDATE\_LEVEL**, it is considered an insert if the price level is not found on the particular side of the market depth being updated. It is considered an update, if the price level is found on the particular side of market depth being updated.

This message uses a double datatype for the **Price** field. There is no level index. It is the responsibility of the Client to determine where in its market depth array it is maintaining where the insert/update/delete operation needs to occur.

Since floating-point comparisons are not always precise, there should be a comparison made only to the number of decimal places the symbol specifies in its security definition. This can be determined through the [SECURITY\\_DEFINITION\\_RESPONSE::PriceDisplayFormat](#) field.

Field Name	Field Description
------------	-------------------

<a href="#">[unsigned int16]</a> <b>Size</b>	The standard message size field. Automatically set by constructor.
<a href="#">[unsigned int16]</a> <b>Type</b>	The standard message type field. Automatically set by constructor.
<a href="#">[unsigned int32]</a> <b>SymbolID</b>	This is the same SymbolID sent by the Client in the <a href="#">MARKET_DEPTH_REQUEST</a> message which corresponds to the Symbol that the depth in this message is for.
<a href="#">[AtBidOrAskEnum]</a> <b>Side</b>	Specifies whether the side being updated is Bid (AT_BID) or Ask (AT_ASK).
<a href="#">[double]</a> <b>Price</b>	The price level to insert, update or delete.
<a href="#">[double]</a> <b>Quantity</b>	The number of shares/contracts at the Price level. This will be 0 in the case where <b>UpdateType</b> is set to <b>MARKET_DEPTH_DELETE_LEVEL</b> .

<a href="#">[MarketDepthUpdateTypeEnum]</a> <b>UpdateType</b>	<p>Specifies whether this is <b>MARKET_DEPTH_INSERT_UPDATE_LEVEL</b> operation or <b>MARKET_DEPTH_DELETE_LEVEL</b> operation.</p> <p><b>MARKET_DEPTH_INSERT_UPDATE_LEVEL</b>  Insert or update in the market depth book on the specified side, the particular Price and Volume specified. It is an insert operation if the price level does not exist. It is an update operation if the price level already exists. In case of insert, the other levels in the market depth book need to be shifted to make room for the new level.</p> <p><b>MARKET_DEPTH_DELETE_LEVEL</b>: Remove from the market depth book on the specified side, the specified Price level. The other levels need to be shifted to fill in the missing level. In this particular case the Quantity is ignored and will be 0.</p>
<a href="#">[t_DateTimeWithMilliseconds]</a> <b>DateTime</b>	The Date-Time of the market depth update.
<a href="#">[uint32_t]</a> <b>NumOrders</b>	The number of orders at the <b>Price</b> .

### MARKET\_DEPTH\_UPDATE\_LEVEL\_FLOAT\_WITH\_MILLISECONDS [s\_MarketDepthUpdateLevelFloatWithMilliseconds structure] Server >> Client

Sent by the Server to the Client to Update/Insert or Delete a particular market depth price level in the market depth book maintained by the Client.

This message is a more compact version of the [MARKET\\_DEPTH\\_UPDATE\\_LEVEL](#) message. For the **Price** and **Quantity** fields, it uses a [4 byte float](#) for compactness. It also supports millisecond precision for the timestamp.



Field Name	Field Description
<a href="#">[unsigned int16]</a> <b>Size</b>	The standard message size field. Automatically set by constructor.
<a href="#">[unsigned int16]</a> <b>Type</b>	The standard message type field. Automatically set by constructor.
<a href="#">[unsigned int32]</a> <b>SymbolID</b>	This is the same SymbolID sent by the Client in the <a href="#">MARKET_DEPTH_REQUEST</a> message which corresponds to the Symbol that the depth update in this message is for.
<a href="#">[t_DateTimeWithMillisecondsInt]</a> <b>DateTime</b>	<p>The Date-Time of the market depth update with millisecond precision.</p> <p>This is an integer representing the number of milliseconds since the UNIX Epoch.</p>
<a href="#">[AtBidOrAskEnum]</a> <b>Side</b>	Specifies whether the side being updated is Bid (AT_BID) or Ask (AT_ASK).
<a href="#">[float]</a> <b>Price</b>	The price level to insert, update or delete.
<a href="#">[float]</a> <b>Quantity</b>	<p>The number of shares/contracts at the Price level. This will be 0 in the case where <b>UpdateType</b> is set to <b>MARKET_DEPTH_DELETE_LEVEL</b>.</p>

<p><a href="#">[MarketDepthUpdateTypeEnum]</a> <b>UpdateType</b></p>	<p>Specifies whether this is <b>MARKET_DEPTH_INSERT_UPDATE_LEVEL</b> operation or <b>MARKET_DEPTH_DELETE_LEVEL</b> operation.</p> <p><b>MARKET_DEPTH_INSERT_UPDATE_LEVEL</b> Insert or update in the market depth book on the specified side, the particular Price and Volume specified. It is an insert operation if the price level does not exist. It is an update operation if the price level already exists. In case of insert, the other levels in the market depth book need to be shifted to make room for the new level.</p> <p><b>MARKET_DEPTH_DELETE_LEVEL</b>: Remove from the market depth book on the specified side, the specified Price level. The other levels need to be shifted to fill in the missing level. In this particular case the Quantity is ignored and will be 0.</p>
<p><a href="#">[uint32_t]</a> <b>NumOrders</b></p>	<p>The number of orders at the <b>Price</b>.</p>
<p><a href="#">[FinalUpdateInBatchEnum]</a> <b>FinalUpdateInBatch</b></p>	<p>An indicator whether this is the final message or not in a batch of updates.</p>

## MARKET\_DEPTH\_UPDATE\_LEVEL\_NO\_TIMESTAMP

[\[s\\_MarketDepthUpdateLevelNoTimestamp structure\]](#) Server >> Client

Sent by the Server to the Client to **Update/Insert** or **Delete** a particular market depth price level in the market depth book maintained by the Client.

This message is identical to the [MARKET\\_DEPTH\\_UPDATE\\_LEVEL](#) message except it has no timestamp field. It needs to be sent when there is no change with the timestamp for the market depth update as compared to the prior update.

When the Server sends this message to the Client, the Client needs to use the prior received market depth update timestamp to know what the timestamp is for this message.

Field Name	Field Description
<a href="#">[unsigned int16]</a> <b>Size</b>	The standard message size field. Automatically set by constructor.
<a href="#">[unsigned int16]</a> <b>Type</b>	The standard message type field. Automatically set by constructor.
<a href="#">[unsigned int32]</a> <b>SymbolID</b>	This is the same SymbolID sent by the Client in the <a href="#">MARKET_DEPTH_REQUEST</a> message which corresponds to the Symbol that the order in this message is for.
<a href="#">[uint8_t]</a> <b>Side</b>	Specifies whether the side being updated is Bid (AT_BID) or Ask (AT_ASK).
<a href="#">[float]</a> <b>Price</b>	The price level to insert, update or delete.
<a href="#">[float]</a> <b>Quantity</b>	The number of shares/contracts at the Price level. This will be 0 in the case where <b>UpdateType</b> is set to <b>MARKET_DEPTH_DELETE_LEVEL</b> .
<a href="#">[FinalUpdateInBatchEnum]</a> <b>FinalUpdateInBatch</b>	An indicator whether this is the beginning of a final update in a batch of updates.
<a href="#">[unsigned int16]</a> <b>NumOrders</b>	The number of orders at the <b>Price</b> .

<p><a href="#">[MarketDepthUpdateTypeEnum]</a> <b>UpdateType</b></p>	<p>Specifies whether this is <b>MARKET_DEPTH_INSERT_UPDATE_LEVEL</b> operation or <b>MARKET_DEPTH_DELETE_LEVEL</b> operation.</p> <p><b>MARKET_DEPTH_INSERT_UPDATE_LEVEL</b> Insert or update in the market depth book on the specified side, the particular Price and Volume specified. It is an insert operation if the price level does not exist. It is an update operation if the price level already exists. In case of insert, the other levels in the market depth book need to be shifted to make room for the new level.</p> <p><b>MARKET_DEPTH_DELETE_LEVEL</b>: Remove the price level from the market depth book on the specified side, the specified Price level. The other levels need to be shifted to fill in the missing level. In this particular case the Quantity is ignored and will be 0.</p>
--	--

## TRADING\_SYMBOL\_STATUS [s\_TradingSymbolStatus structure] Server >> Client

Sent by the Server to the Client to indicate the status of the symbol in regards to whether trading is open or closed or some other intermediate state.

Field Name	Field Description
<a href="#">[unsigned int16]</a> <b>Size</b>	The standard message size field. Automatically set by constructor.
<a href="#">[unsigned int16]</a> <b>Type</b>	The standard message type field. Automatically set by constructor.

<p><u>[unsigned int32]</u> <b>SymbolID</b></p>	<p>This is the same SymbolID sent by the Client in the <u>MARKET_DATA_REQUEST</u> message which corresponds to the Symbol that the status in this message is for.</p>
<p><u>[TradingStatusEnum]</u> <b>Status</b></p>	<p>The current trading status for the symbol. Can be one of the following.</p> <ul style="list-style-type: none"> <li>• TRADING_STATUS_UNKNOWN = 0</li> <li>• TRADING_STATUS_PRE_OPEN = 1</li> <li>• TRADING_STATUS_OPEN = 2</li> <li>• TRADING_STATUS_CLOSE = 3</li> <li>• TRADING_STATUS_TRADING_HALT = 4</li> </ul>

---

\*Last modified Friday, 21st January, 2022.