

## Order Entry and Modification Messages

- [SUBMIT\\_NEW\\_SINGLE\\_ORDER](#)
- [SUBMIT\\_NEW\\_SINGLE\\_ORDER\\_INT](#)
- [SUBMIT\\_NEW\\_OCO\\_ORDER](#)
- [SUBMIT\\_NEW\\_OCO\\_ORDER\\_INT](#)
- [SUBMIT\\_FLATTEN\\_POSITION\\_ORDER](#)
- [CANCEL\\_REPLACE\\_ORDER](#)
- [CANCEL\\_REPLACE\\_ORDER\\_INT](#)
- [CANCEL\\_ORDER](#)

---

### SUBMIT\_NEW\_SINGLE\_ORDER [s\_SubmitNewSingleOrder structure] Client >> Server

---

This message is used to submit a new single order into the market from the Client to the 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="#">[char]</a> <b>Symbol</b>	The symbol for the order.
<a href="#">[char]</a> <b>Exchange</b>	The optional exchange for the symbol.

<p><a href="#">[char]</a> <b>ClientOrderID</b></p>	<p>This is the Client supplied order identifier. The Server will maintain this order identifier throughout the life of the order and always provide it back through the <b>ClientOrderID</b> field in the <a href="#">ORDER_UPDATE</a> messages for the order.</p> <p>This identifier cannot be an identifier used for a currently <u>open</u> order and it cannot be an identifier previously used in the current trading session. The trading session typically will be a 24-hour period defined by the Server. The Server shall reject an order with a client order identifier that is for a currently open order or which has already been used during the current trading session.</p>
<p><a href="#">[OrderTypeEnum]</a> <b>OrderType</b></p>	<p>The order type. For list of order types, refer to <a href="#">OrderTypeEnum</a>.</p>
<p><a href="#">[BuySellEnum]</a> <b>BuySell</b></p>	<p>The side of the order. Either Buy or Sell.</p>
<p><a href="#">[double]</a> <b>Price1</b></p>	<p>This is the price of the order. This is the limit price for a <b>Limit</b> order, the stop price for a <b>Stop</b> order, or the trigger price for a Market if Touched order.</p>
<p><a href="#">[double]</a> <b>Price2</b></p>	<p>For a <b>Stop-Limit</b> order, this is the limit price. This only applies to <b>Stop-Limit</b> orders.</p>

<a href="#">[TimeInForceEnum]</a> <b>TimeInForce</b>	<p>The Time in Force for the order or orders (in the case of an OCO order).</p> <p>For more information, refer to <a href="#">TimeInForceEnum</a>.</p>
<a href="#">[t_DateTime]</a> <b>GoodTillDateTime</b>	<p>In the case of when the <b>TimeInForce</b> is <b>TIF_GOOD_TILL_DATE_TIME</b>, this specifies the expiration Date-Time of the order.</p>
<a href="#">[double]</a> <b>Quantity</b>	<p>The quantity of the order. The exact meaning of this will be specified by the Server implementation.</p>
<a href="#">[char]</a> <b>TradeAccount</b>	<p>This is the trade account as a text string that the order belongs to.</p>
<a href="#">[unsigned _____ int8]</a> <b>IsAutomatedOrder</b>	<p>This is set 1 to signify the order has been submitted by an automated trading process.</p>
<a href="#">[unsigned _____ int8]</a> <b>IsParentOrder</b>	<p>The Client will set this to 1 when the order is part of a bracket order. This indicates that this is the parent order. A bracket order will consist of a <a href="#">SUBMIT_NEW_SINGLE_ORDER</a> message followed by a <a href="#">SUBMIT_NEW_OCO_ORDER</a> message. The Server will use <b>IsParentOrder</b> as a flag to know that this message is a parent order. The Server will hold onto this order until it receives the subsequent <a href="#">SUBMIT_NEW_OCO_ORDER</a> message and then process all of the orders as one complete set.</p>

<a href="#">[char]</a> <b>FreeFormText</b>	Optional: This is an optional text string which can be set by the Client to associate text with the order. The Server is not under any obligation to use this text and it may place a limitation on the length of this text.
<a href="#">[OpenCloseTradeEnum]</a> <b>OpenOrClose</b>	For the description for this field, refer to <a href="#">OpenCloseTradeEnum</a> .
<a href="#">[double]</a> <b>MaxShowQuantity</b>	This is an optional field. This field is provided to the exchange and represents the maximum quantity to show in the Limit order book for the order. By default this is set to 0 which means it is not used.
<a href="#">[char]</a> <b>Price1AsString</b>	<p>This is an optional field which may be used by the Server.</p> <p>This field is the order price 1 as a string.</p>
<a href="#">[char]</a> <b>Price2AsString</b>	<p>This is an optional field which may be used by the Server.</p> <p>This field is the order price 2 as a string.</p>

### **SUBMIT\_NEW\_SINGLE\_ORDER\_INT [s\_SubmitNewSingleOrderInt structure] Client >> Server**

The **SUBMIT\_NEW\_SINGLE\_ORDER\_INT** message is identical to [SUBMIT\\_NEW\\_SINGLE\\_ORDER](#) except that the order prices are as integers.

The Client will only send this message to the Server if **UseIntegerPriceOrderMessages** is set in the [LOGON\\_RESPONSE](#) message.

When setting the **Price1** and **Price2** fields, multiply the order price by the **FloatToIntPriceMultiplier** value provided in the [SECURITY\\_DEFINITION\\_RESPONSE](#) message for the Symbol being traded.

This message also contains the **Divisor** field. This is the **FloatToIntPriceMultiplier** value provided in the [SECURITY\\_DEFINITION\\_RESPONSE](#) message for the symbol being traded. The Server needs to divide the **Price1** and **Price2** fields by this Divisor to get the prices as float values.

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="#">[char]</a> <b>Symbol</b>	The symbol for the order.
<a href="#">[char]</a> <b>Exchange</b>	The optional exchange for the symbol.
<a href="#">[char]</a> <b>ClientOrderID</b>	<p>This is the Client supplied order identifier. The Server will maintain this order identifier throughout the life of the order and always provide it back through the <b>ClientOrderID</b> field in the <a href="#">ORDER_UPDATE</a> messages for the order.</p> <p>This identifier cannot be an identifier used for a currently <u>open</u> order and it cannot be an identifier previously used in the current trading session. The trading session typically will be a 24-hour period defined by the Server. The Server shall reject an order with a client order identifier that is for a currently open order or which has already been used during the current trading session.</p>

<a href="#">[OrderTypeEnum]</a> <b>OrderType</b>	The order type. For list of order types, refer to <a href="#">OrderTypeEnum</a> .
<a href="#">[BuySellEnum]</a> <b>BuySell</b>	The side of the order. Either Buy or Sell.
<a href="#">[int64]</a> <b>Price1</b>	This is the price of the order. This is the limit price for a <b>Limit</b> order, the stop price for a <b>Stop</b> order, or the trigger price for a Market if Touched order.
<a href="#">[int64]</a> <b>Price2</b>	For a <b>Stop-Limit</b> order, this is the limit price. This only applies to <b>Stop-Limit</b> orders.
<a href="#">[TimeInForceEnum]</a> <b>TimeInForce</b>	The Time in Force for the order or orders (in the case of an OCO order).  For more information, refer to <a href="#">TimeInForceEnum</a> .
<a href="#">[t_DateTime]</a> <b>GoodTillDateTime</b>	In the case of when the <b>TimeInForce</b> is <b>TIF_GOOD_TILL_DATE_TIME</b> , this specifies the expiration Date-Time of the order.
<a href="#">[float]</a> <b>Divisor</b>	This is the <b>FloatToIntPriceMultiplier</b> value provided in the <a href="#">SECURITY_DEFINITION_RESPONSE</a> message for the symbol being traded. The Server needs to divide the <b>Price1</b> and <b>Price2</b> fields by this Divisor to get the prices as float values.

<a href="#">[int64]</a> <b>Quantity</b>	The quantity of the order. The exact meaning of this will be specified by the Server implementation.
<a href="#">[char]</a> <b>TradeAccount</b>	This is the trade account as a text string that the order belongs to.
<a href="#">[unsigned int8]</a> <b>IsAutomatedOrder</b>	This is set 1 to signify the order has been submitted by an automated trading process.
<a href="#">[unsigned int8]</a> <b>IsParentOrder</b>	The Client will set this to 1 when the order is part of a bracket order. This indicates that this is the parent order. A bracket order will consist of a <a href="#">SUBMIT_NEW_SINGLE_ORDER</a> message followed by a <a href="#">SUBMIT_NEW_OCO_ORDER</a> message. The Server will use <b>IsParentOrder</b> as a flag to know that this message is a parent order. The Server will hold onto this order until it receives the subsequent <a href="#">SUBMIT_NEW_OCO_ORDER</a> message and then process all of the orders as one complete set.
<a href="#">[char]</a> <b>FreeFormText</b>	Optional: This is an optional text string which can be set by the Client to associate text with the order. The Server is not under any obligation to use this text and it may place a limitation on the length of this text.
<a href="#">[OpenCloseTradeEnum]</a> <b>OpenOrClose</b>	For the description for this field, refer to <a href="#">OpenCloseTradeEnum</a> .

**SUBMIT\_NEW\_OCO\_ORDER [s\_SubmitNewOCOOrder structure]**

## Client >> Server

This is a message from the Client to the Server for submitting an order cancels order (OCO) pair into the market. What this means is when one of the orders is filled or canceled, the other order will be canceled. If one order partially fills, the other order will be reduced in quantity by the fill amount of the order that partially filled.

A service provider must implement OCO orders on the server so that they can independently be modified (Cancel/Replace) and canceled independently using each order's distinct **ServerOrderID**. Although, if one of the orders is canceled by the Client, the other order will be canceled as well unless they have a parent order, as specified through the **ParentTriggerClientOrderID** field, in which case the other order should remain open.

If the OCO order pair is rejected, this must be communicated through two separate [ORDER\\_UPDATE](#) messages, 1 for each order, with the OrderUpdateReason set to **NEW\_ORDER\_REJECTED**.

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="#">[char]</a> <b>Symbol</b>	The symbol for the order.
<a href="#">[char]</a> <b>Exchange</b>	The optional exchange for the symbol.
<a href="#">[char]</a> <b>ClientOrderID_1</b>	The Client supplied order identifier for the first order. The Server will remember this for the life of the order.



<a href="#">[OrderTypeEnum]</a> <b>OrderType_1</b>	The order type for the first order. For list of order types, refer to <a href="#">OrderTypeEnum</a> .
<a href="#">[BuySellEnum]</a> <b>BuySell_1</b>	The side for the first order. Either Buy or Sell.
<a href="#">[double]</a> <b>Price1_1</b>	This is the price of the first order. This is the limit price for a <b>Limit</b> order, the stop price for a <b>Stop</b> order, or the trigger price for a Market if Touched order.
<a href="#">[double]</a> <b>Price2_1</b>	This is the second price for the first order. For a <b>Stop-Limit</b> order, this is the limit price. <b>Price2_1</b> only applies to <b>Stop-Limit</b> orders.
<a href="#">[double]</a> <b>Quantity_1</b>	The quantity for the first order.
<a href="#">[char]</a> <b>ClientOrderID_2</b>	The Client supplied order identifier for the second order. The Server will remember this for the life of the order.
<a href="#">[OrderTypeEnum]</a> <b>OrderType_2</b>	The order type for the second order. For list of order types, refer to <a href="#">OrderTypeEnum</a> .
<a href="#">[BuySellEnum]</a> <b>BuySell_2</b>	The side for the second order. Either Buy or Sell.

<a href="#">[double]</a> <b>Price1_2</b>	This is the price of the second order. This is the limit price for a <b>Limit</b> order, the stop price for a <b>Stop</b> order, or the trigger price for a Market if Touched order.
<a href="#">[double]</a> <b>Price2_2</b>	This is the second price for the second order. For a <b>Stop-Limit</b> order, this is the limit price. <b>Price2_2</b> only applies to <b>Stop-Limit</b> orders.
<a href="#">[double]</a> <b>Quantity_2</b>	The quantity for the second order.
<a href="#">[TimeInForceEnum]</a> <b>TimeInForce</b>	The Time in Force for the order or orders (in the case of an OCO order).  For more information, refer to <a href="#">TimeInForceEnum</a> .
<a href="#">[t_DateTime]</a> <b>GoodTillDateTime</b>	In the case of when the <b>TimeInForce</b> is <b>TIF_GOOD_TILL_DATE_TIME</b> , this specifies the expiration Date-Time of the order. This applies to both of the orders in the OCO pair.
<a href="#">[unsigned int8]</a> <b>IsAutomatedOrder</b>	Set to 1 for an order submitted by an automated trading system.
<a href="#">[char]</a> <b>TradeAccount</b>	This is the trade account as a text string that the orders belong to.

<p><a href="#">[char]</a> <b>ParentTriggerClientOrderID</b></p>	<p>Optional: This field supports the submission of an OCO order pair which has a parent. This is known as a Bracket order.</p> <p>For complete documentation for bracket orders, refer to <a href="#">Bracket Order Procedures</a>.</p>
<p><a href="#">[char]</a> <b>FreeFormText</b></p>	<p>Optional: This is an optional text string which can be set by the Client to associate text with each of the OCO orders. The Server is not under any obligation to use this text and it may place a limitation on the length of this text.</p>
<p><a href="#">[OpenCloseTradeEnum]</a> <b>OpenOrClose</b></p>	<p>For the description for this field, refer to <a href="#">OpenCloseTradeEnum</a>.</p>
<p><a href="#">[PartialFillHandlingEnum]</a> <b>PartialFillHandling</b></p>	<p>Specifies how partial fills should be handled when when one of the orders in the OCO order set partially fills.</p> <p>For the possible values, refer to <a href="#">PartialFillHandlingEnum</a>.</p>

## [unsigned int8] **UseOffsets**

This field is only relevant to a Bracket order which is the case when the **ParentTriggerClientOrderID** field is set.

**UseOffsets** can be set to 1 and indicates that the **OffsetFromParent1** and **OffsetFromParent2** fields specify the two OCO order prices as a price offset from the parent order **Price1** field, rather than an absolute price. In this case **Price1\_1** and **Price1\_2** are not used.

When **UseOffsets** is set to 0, the default, then the OCO order prices are specified with **Price1\_1** and **Price1\_2**.

When **UseOffsets** is set to 1 and the **OffsetFromParent1** and **OffsetFromParent2** fields are set, it is necessary that the parent order **Price1** field be set even in the case of a Market order type. In the case of a Market order type use the current order price. This is so that the server has a reference price for the offsets in case it needs to translate them to actual prices.

When the parent order fills, it is expected the Server will maintain the specified offsets to the parent order fill price for the Target and Stop orders.

A Server is not required to support this field.

<p><a href="#">[double]</a> <b>OffsetFromParent1</b></p>	<p>When <b>UseOffsets</b> is set to 1 , then this field specifies the <b>Price1_1</b> price as an offset from the parent order. In this case <b>Price1_1</b> will not be set in the message. Instead the Server calculates that price from this offset and parent order price.</p> <p>This needs to always be set to a positive price value which is an offset from the parent order price. The Server will make the correct calculation based upon the Side and Order Type.</p> <p>A Server is not required to support this field.</p>
<p><a href="#">[double]</a> <b>OffsetFromParent2</b></p>	<p>When <b>UseOffsets</b> is set to 1 , then this field specifies the <b>Price1_2</b> price as an offset from the parent order. In this case <b>Price1_2</b> will not be set in the message. Instead the Server calculates that price from this offset and parent order price.</p> <p>This needs to always be set to a positive price value which is an offset from the parent order price. The Server will make the correct calculation based upon the Side and Order Type.</p> <p>A Server is not required to support this field.</p>

## **SUBMIT\_NEW\_OCO\_ORDER\_INT [s\_SubmitNewOCOOrderInt structure] Client >> Server**

The **SUBMIT\_NEW\_OCO\_ORDER\_INT** message is identical to [SUBMIT\\_NEW\\_OCO\\_ORDER](#) except that the order prices are as integers.

The Client will only send this message to the Server if **UseIntegerPriceOrderMessages** is set in the [LOGON\\_RESPONSE](#) message.

When setting the **Price1** and **Price2** fields, multiply the order price by the **FloatToIntPriceMultiplier** value provided in the [SECURITY\\_DEFINITION\\_RESPONSE](#)

message for the Symbol being traded.

This message also contains the **Divisor** field. This is the **FloatToIntPriceMultiplier** value provided in the [SECURITY\\_DEFINITION\\_RESPONSE](#) message for the symbol being traded. The Server needs to divide the **Price1** and **Price2** fields by this Divisor to get the prices as float values.

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="#">[char]</a> <b>Symbol</b>	The symbol for the order.
<a href="#">[char]</a> <b>Exchange</b>	The optional exchange for the symbol.
<a href="#">[char]</a> <b>ClientOrderID_1</b>	The Client supplied order identifier for the first order. The Server will remember this for the life of the order.
<a href="#">[OrderTypeEnum]</a> <b>OrderType_1</b>	The order type for the first order. For list of order types, refer to <a href="#">OrderTypeEnum</a> .
<a href="#">[BuySellEnum]</a> <b>BuySell_1</b>	The side for the first order. Either Buy or Sell.

<a href="#">[int64]</a> <b>Price1_1</b>	This is the price of the first order. This is the limit price for a Limit order, the stop price for a Stop order, or the trigger price for a Market if Touched order.
<a href="#">[int64]</a> <b>Price2_1</b>	This is the second price for the first order. For a Stop-Limit order, this is the limit price. This only applies to Stop-Limit orders.
<a href="#">[int64]</a> <b>Quantity_1</b>	The quantity of the first order. The exact meaning of this will be specified by the Server implementation.
<a href="#">[char]</a> <b>ClientOrderID_2</b>	The Client supplied order identifier for the second order. The Server will remember this for the life of the order.
<a href="#">[OrderTypeEnum]</a> <b>OrderType_2</b>	The order type of the second order. For list of order types, refer to <a href="#">OrderTypeEnum</a> .
<a href="#">[BuySellEnum]</a> <b>BuySell_2</b>	The side of the second order. Either Buy or Sell.
<a href="#">[int64]</a> <b>Price1_2</b>	This is the price of the second order. This is the limit price for a Limit order, the stop price for a Stop order, or the trigger price for a Market if Touched order.

<a href="#">[int64]</a> <b>Price2_2</b>	This is the second price for the second order. For a Stop-Limit order, this is the limit price. This only applies to Stop-Limit orders.
<a href="#">[int64]</a> <b>Quantity_2</b>	The quantity of the second order. The exact meaning of this will be specified by the Server implementation.
<a href="#">[TimeInForceEnum]</a> <b>TimeInForce</b>	The Time in Force for the order or orders (in the case of an OCO order).  For more information, refer to <a href="#">TimeInForceEnum</a> .
<a href="#">[t_DateTime]</a> <b>GoodTillDateTime</b>	In the case of when the <b>TimeInForce</b> is <b>TIF_GOOD_TILL_DATE_TIME</b> , this specifies the expiration Date-Time of the order. This applies to both of the orders in the OCO pair.
<a href="#">[char]</a> <b>TradeAccount</b>	This is the trade account as a text string that the orders belong to.
<a href="#">[unsigned int8]</a> <b>IsAutomatedOrder</b>	Set to 1 for an order submitted by an automated trading system.



<p><a href="#">[char]</a> <b>ParentTriggerClientOrderID</b></p>	<p>Optional: This field supports the submission of an OCO order pair which has a parent. This is known as a Bracket order.</p> <p>For complete documentation for bracket orders, refer to <a href="#">Bracket Order Procedures</a>.</p>
<p><a href="#">[char]</a> <b>FreeFormText</b></p>	<p>Optional: This is an optional text string which can be set by the Client to associate text with each of the OCO orders. The Server is not under any obligation to use this text and it may place a limitation on the length of this text.</p>
<p><a href="#">[float]</a> <b>Divisor</b></p>	<p>This is the <b>FloatToIntPriceMultiplier</b> value provided in the <a href="#">SECURITY DEFINITION RESPONSE</a> message for the Symbol being traded. The Server needs to divide the <b>Price1_*</b> and <b>Price2_*</b> fields by this Divisor to get the prices as float values.</p>
<p><a href="#">[OpenCloseTradeEnum]</a> <b>OpenOrClose</b></p>	<p>For the description for this field, refer to <a href="#">OpenCloseTradeEnum</a>.</p>
<p><a href="#">[PartialFillHandlingEnum]</a> <b>PartialFillHandling</b></p>	<p>Specifies how partial fills should be handled when when one of the orders in the OCO order set partially fills.</p> <p>For the possible values, refer to <a href="#">PartialFillHandlingEnum</a>.</p>

**SUBMIT\_FLATTEN\_POSITION\_ORDER [s\_SubmitFlattenPositionOrder**

**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="#">[char]</a> <b>Symbol</b>	The symbol of the Trade Position to flatten.
<a href="#">[char]</a> <b>Exchange</b>	The optional exchange for the Symbol.
<a href="#">[char]</a> <b>TradeAccount</b>	The trade account as a text string of the Trade Position to flatten.
<a href="#">[char]</a> <b>ClientOrderID</b>	<p>The Client supplied order identifier for the order which will be created to flatten the Trade Position.</p> <p>The Server will remember this for the life of the order.</p>
<a href="#">[char]</a> <b>FreeFormText</b>	<p>Optional: This is an optional text string which can be set by the Client to associate text with the order which will be created to flatten the Trade Position.</p> <p>The Server is not under any obligation to use this text and it may place a limitation on the length of this text.</p>

<a href="#">[unsigned int8]</a> <b>IsAutomatedOrder</b>	Set to 1 for an order submitted by an automated trading system.
--	---

## **CANCEL\_REPLACE\_ORDER [s\_CancelReplaceOrder structure] Client >> Server**

This message is sent by the Client to the Server to cancel and replace an existing order. This is also known as an order modification.

When the cancel and replace operation is completed, an [OrderUpdate](#) message is sent by the Server with the **OrderUpdateReason** field set to **ORDER\_CANCEL\_REPLACE\_COMPLETE**. If the cancel and replace operation cannot be completed, an [OrderUpdate](#) message is sent by the Server with the **OrderUpdateReason** set to **ORDER\_CANCEL\_REPLACE\_REJECTED**.

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="#">[char]</a> <b>ServerOrderID</b>	<p>This is the order identifier for the order to modify. The Client needs to set this to the <b>ServerOrderID</b> field received back in the most recent <a href="#">ORDER_UPDATE</a> message for the order.</p> <p>The Server will rely upon this <b>ServerOrderID</b> and only this order identifier to identify the order to be canceled and replaced. Although the given <b>ClientOrderID</b> by the Client must not change.</p>

<p><a href="#">[char]</a> <b>ClientOrderID</b></p>	<p>This is the Client's own order identifier for the order.</p> <p>This must be the same throughout the life of the order. If the Server sees that this order identifier has changed in relation to the <b>ServerOrderID</b>, then it should reject this message with a <a href="#">ORDER_UPDATE</a> message with the <b>OrderUpdateReason</b> set to <b>ORDER_CANCEL_REPLACE_REJECTED</b></p> <p>In the case where the order modification cannot be performed because the <b>ServerOrderID</b> does not exist, the Server will send a <a href="#">ORDER_UPDATE</a> message with the <b>OrderUpdateReason</b> set to <b>ORDER_CANCEL_REPLACE_REJECTED</b> and <b>ClientOrderID</b> set to the given <b>ClientOrderID</b> in this message. <b>ServerOrderID</b> will be unset because an invalid server order identifier was given.</p>
<p><a href="#">[double]</a> <b>Price1</b></p>	<p>For orders that require a price, this is the new order price.</p> <p>This value can be left unset indicating to the Server that Price1 must not change and only the Quantity. In this case it is necessary to set <b>Price1IsSet</b> to a 0 value.</p>
<p><a href="#">[double]</a> <b>Price2</b></p>	<p>For Stop-Limit orders this is the new Limit price. For other order types it is not used.</p> <p>This value can be left unset indicating to the Server that Price2 must not change and only the Quantity. In this case it is necessary to set <b>Price2IsSet</b> to a 0 value.</p>

<p><u>[double]</u> <b>Quantity</b></p>	<p>This is the new order quantity. If this is 0, then this means the order quantity must not be changed by the Server.</p> <p>If the order has partially filled, then this is going to be the order quantity which also includes the amount which has partially filled.</p> <p>For example, if the original quantity was 10 and there has been a partial fill of 3, the Client wants a fill of 2 more making a total of 5, then the Client will set this to 5.</p>
<p><u>[unsigned int8]</u> <b>Price1IsSet</b></p>	<p>When this field is set to a nonzero value, it indicates that <b>Price1</b> is set and the server should use the value, if it applies to the order type.</p> <p>The default value is 1.</p>
<p><u>[unsigned int8]</u> <b>Price2IsSet</b></p>	<p>When this field is set to a nonzero value, it indicates that <b>Price2</b> is set and the server should use the value, if it applies to the order type.</p> <p>The default value is 1.</p>

<p><a href="#">[TimeInForceEnum]</a> <b>TimeInForce</b></p>	<p>The Time in Force for the order. For a list of Time in Force values, refer to <a href="#">TimeInForceEnum</a>.</p> <p>The default value is <b>TIF_UNSET</b>.</p> <p>When this field is set to a value other than <b>TIF_UNSET</b>, it indicates that the <b>TimeInForce</b> is being changed.</p> <p>If the server does not support changing the Time in Force of the order, it needs to reject this <b>CANCEL_REPLACE_ORDER</b> message and send an <a href="#">ORDER_UPDATE</a> message with the OrderUpdateReason set to <b>ORDER_CANCEL_REPLACE_REJECTED</b>.</p> <p>The server is under no obligation to support changing the Time in Force.</p>
<p><a href="#">[t_DateTime]</a> <b>GoodTillDateTime</b></p>	<p>In the case of when the <b>TimeInForce</b> field is <b>TIF_GOOD_TILL_DATE_TIME</b>, this specifies the expiration Date-Time of the order.</p>
<p><a href="#">[uint8_t]</a> <b>UpdatePrice1OffsetToParent</b></p>	<p>This is an optional field. If modifying a child order which is part of a Server managed bracket order, then when this variable is set to 1 it provides an indication to the Server to update the internal server managed price offset to the parent order that this child order has to the parent.</p> <p>This will ensure the Server will maintain the proper offset of the child order to the fill price of the parent order when the parent order fills.</p>

**CANCEL\_REPLACE\_ORDER\_INT [s\_CancelReplaceOrderInt structure] Client >> Server**

The **CANCEL\_REPLACE\_ORDER\_INT** is a message from the Client to Server to cancel and replace (modify) an order.

It is identical to [CANCEL\\_REPLACE\\_ORDER](#) except the prices are as integers.

When the cancel and replace operation is completed, an [OrderUpdate](#) message is sent by the Server with the **OrderUpdateReason** field set to **ORDER\_CANCEL\_REPLACE\_COMPLETE**. If the cancel and replace operation cannot be completed, an [OrderUpdate](#) message is sent by the Server with the **OrderUpdateReason** set to **ORDER\_CANCEL\_REPLACE\_REJECTED**.

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="#">[char]</a> <b>ServerOrderID</b>	<p>This is the order identifier for the order to modify. The Client needs to set this to the <b>ServerOrderID</b> field received back in the most recent <a href="#">ORDER_UPDATE</a> message for the order.</p> <p>The Server will rely upon this <b>ServerOrderID</b> and only this order identifier to identify the order to be canceled and replaced. Although the given <b>ClientOrderID</b> by the Client must not change.</p>

<p><a href="#">[char]</a> <b>ClientOrderID</b></p>	<p>This is the Client's own order identifier for the order.</p> <p>This must be the same throughout the life of the order. If the Server sees that this order identifier has changed in relation to the <b>ServerOrderID</b>, then it should reject this message with a <a href="#">ORDER_UPDATE</a> message with the <b>OrderUpdateReason</b> set to <b>ORDER_CANCEL_REPLACE_REJECTED</b></p> <p>In the case where the order modification cannot be performed because the <b>ServerOrderID</b> does not exist, the Server will send a <a href="#">ORDER_UPDATE</a> message with the <b>OrderUpdateReason</b> set to <b>ORDER_CANCEL_REPLACE_REJECTED</b> and <b>ClientOrderID</b> set to the given <b>ClientOrderID</b> in this message. <b>ServerOrderID</b> will be unset because an invalid server order identifier was given.</p>
<p><a href="#">[int64]</a> <b>Price1</b></p>	<p>For orders that require a price, this is the new order price.</p> <p>This is an integer value. This is calculated by multiplying the actual order price by the <b>FloatToIntPriceMultiplier</b> field from the <a href="#">SECURITY_DEFINITION_RESPONSE</a> message for the Symbol of the order.</p> <p>This value can be left unset indicating to the Server that Price1 must not change and only the Quantity. In this case it is necessary to set <b>Price1IsSet</b> to a 0 value.</p>



<p><a href="#">[int64]</a> <b>Price2</b></p>	<p>For Stop-Limit orders this is the new Limit price. For other order types it is not used.</p> <p>This is an integer value. This is calculated by multiplying the actual order price by the <b>FloatToIntPriceMultiplier</b> field from the <a href="#">SECURITY_DEFINITION_RESPONSE</a> message for the Symbol of the order.</p> <p>This value can be left unset indicating to the Server that Price2 must not change and only the Quantity. In this case it is necessary to set <b>Price2IsSet</b> to a 0 value.</p>
<p><a href="#">[float]</a> <b>Divisor</b></p>	<p>This is the <b>FloatToIntPriceMultiplier</b> field received from the <a href="#">SECURITY_DEFINITION_RESPONSE</a> message for the Symbol of the order.</p> <p>The Server needs to divide the <b>Price1</b> or <b>Price2</b> fields by this Divisor to arrive at the original floating-point value.</p>
<p><a href="#">[int64]</a> <b>Quantity</b></p>	<p>This is the new order quantity. If this is 0, then this means the order quantity must not be changed by the Server.</p> <p>If the order has partially filled, then this is going to be the order quantity which also includes the amount which has partially filled.</p> <p>For example, if the original quantity was 10 and there has been a partial fill of 3, the Client wants a fill of 2 more making a total of 5, then the Client will set this to 5.</p>

<p><a href="#">[unsigned int8]</a> <b>Price1IsSet</b></p>	<p>When this field is set to a nonzero value, it indicates that <b>Price1</b> is set and the server should use the value, if it applies to the order type.</p> <p>The default value is 1.</p>
<p><a href="#">[unsigned int8]</a> <b>Price2IsSet</b></p>	<p>When this field is set to a nonzero value, it indicates that <b>Price2</b> is set and the server should use the value, if it applies to the order type.</p> <p>The default value is 1.</p>
<p><a href="#">[TimeInForceEnum]</a> <b>TimeInForce</b></p>	<p>The Time in Force for the order. For a list of Time in Force values, refer to <a href="#">TimeInForceEnum</a>.</p> <p>The default value is <b>TIF_UNSET</b>.</p> <p>When this field is set to a value other than <b>TIF_UNSET</b>, it indicates that the <b>TimeInForce</b> is being changed.</p> <p>If the server does not support changing the Time in Force of the order, it needs to reject this <b>CANCEL_REPLACE_ORDER_INT</b> message and send an <a href="#">ORDER_UPDATE</a> message with the OrderUpdateReason set to <b>ORDER_CANCEL_REPLACE_REJECTED</b>.</p> <p>The server is under no obligation to support changing the Time in Force.</p>

<a href="#">[t_DateTime]</a> <b>GoodTillDateTime</b>	<p>In the case of when the <b>TimeInForce</b> field is <b>TIF_GOOD_TILL_DATE_TIME</b>, this specifies the expiration Date-Time of the order.</p>
<a href="#">[uint8_t]</a> <b>UpdatePrice1OffsetToParent</b>	<p>This is an optional field. If modifying a child order which is part of a Server managed bracket order, then when this variable is set to 1 it provides an indication to the Server to update the internal server managed price offset to the parent order that this child order has to the parent.</p> <p>This will ensure the Server will maintain the proper offset of the child order to the fill price of the parent order when the parent order fills.</p>

## **CANCEL\_ORDER [s\_CancelOrder structure] Client >> Server**

This is a message from the Client to the Server requesting a previously sent order to be canceled.

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>[char]</u> <b>ServerOrderID</b></p>	<p>This is the order identifier for the order to cancel. The Client needs to set this to the <b>ServerOrderID</b> field received back in the most recent <a href="#">ORDER_UPDATE</a> message for the order. The only case in which a <b>ServerOrderID</b> would change is in the case of a successful order Cancel and Replace operation.</p> <p>The Server will rely upon this <b>ServerOrderID</b> and only this order identifier to identify the order to be canceled. Although the given <b>ClientOrderID</b> from the Client must not change.</p>
<p><u>[char]</u> <b>ClientOrderID</b></p>	<p>This is the Client's own order identifier for the order.</p> <p>This must be the same throughout the life of the order. If the Server sees that this order identifier has changed in relation to the <b>ServerOrderID</b> , then it should reject this message with a <a href="#">ORDER_UPDATE</a> message with the <b>OrderUpdateReason</b> set to <b>ORDER_CANCEL_REJECTED</b>.</p> <p>In the case where the order cancellation cannot be performed because the <b>ServerOrderID</b> does not exist, the Server will send a <a href="#">ORDER_UPDATE</a> message with the <b>OrderUpdateReason</b> set to <b>ORDER_CANCEL_REJECTED</b> and <b>ClientOrderID</b> set to the given ClientOrderID in this message. <b>ServerOrderID</b> will be unset because an invalid server order identifier was given.</p>

---

\*Last modified Wednesday, 02nd August, 2023.