A2X
MARKETS

A2X Binary Protocol (ATP) Specification

2nd February 2024

Version 2.9.1

A2X MARKETS

6th Floor Katherine Towers, 1 Park Lane, Wierda Valley, Sandton, 2196. PO Box 781763,
Sandton, Johannesburg 2196

A2X Markets is an exchange licensed in terms of the Financial Markets Act, 2012

Table Of Contents

Version History

Version	Date	Comments
1.0	01 Aug 2017	Initial version.
1.1	09 Feb 2023	We have made some minor changes to the reject codes in the introduction, explicitly stated the datatypes of all fields and accepted protocol versions. We have also corrected the ordering of the standard header and trade bust fields.
2.9	24 May 2023	Updated protocol spec to allow members to take advantage of all functionality using the binary protocol
2.9.1	2 February 2024	Removed reserved field from Order Add Response. Added additional reserved fields to Order Modify message.

1 Introduction

A2X makes use of software provided by Aquis Exchange (Aquis), they have developed a proprietary binary protocol for efficient, streamlined and low-latency order entry and trading activity – Aquis Trading Protocol (ATP).

This document describes the protocol and message formats for this trading interface. It is intended for those firms which are planning to develop against the protocol.

Please contact the A2X support team (support@a2x.co.za) for any questions related to this document.

The latest Exchange version of ATP is v2.9

1.1 Connectivity

A2X will provide trading Members that wish to use ATP to access the platform, with the necessary IP address and port information to establish a TCP/IP connection for their trading session.

In addition, a *senderID* and *password* for the session will be agreed.

One or more connections will be provided to the Member test environment and, on successful certification, connection details to the primary and backup production trading environments will be supplied. Connectivity options should be discussed with the A2X connectivity team.

1.2 Enquiries / Support

Please contact the A2X support team (email support@a2x.co.za) for any questions related to this document.

2 Overview

2.1 Message Structure and Sequence Number Logic

The message structure principles for ATP are to provide efficient, fixed-length messages with binary field data directly aligned to the internal message structure used by the trading system.

To avoid unnecessary traffic on the internal trading system, session messages (login, logout, heartbeat) are not sequenced - only business messages relating to orders and trades are sequenced and recoverable. Sequenced messages belong to one of two streams - trading Member to A2X or A2X to trading Member - each with their own numbering starting at 1 each trading day.

The sequence number of business messages to A2X must always increase. A message with a sequence number that is lower than, or the same as, one already seen will lead to the session being terminated with an appropriate error code. The sequence number of business messages sent from A2X to the trading Member will also increase, incremented by 1 on each message during the day.

2.2 Order Reference

In the ATP protocol it is not necessary for a separate client order identifier to be sent on an Order Add message. The sequence number of the Order message is used by both sides as an ’order reference’ to identify to that orderin subsequent messages.

For example, if an Order Add message is sent in with *msgSeqNo* = 121, then any subsequent Order Modify messages for that order should carry *orderRef* = 121. Similarly an Order Cancel message should also carry the *orderRef* value. Responses and Trade messages from A2X will also carry *orderRef* = 121 to refer to the relevant order.

Note that if an Order Modify or Order Cancel message had been sent in with, for example, *msgSeqNo* = 143 then the Order Modify Response or Order Cancel Response message would carry *orderRef* = 121 and *msgRef* = 143 to identify both the order and the request.

For the convenience of trading Members a *userTag* field has also been made available on Order Add, Order Cancel and Order Modify messages. This value is echoed back by A2X on the related Response messages and also on Trade messages. This allows the trading Member to link business messages back to their own order tracking system using their own internal identifiers. Note that the *userTag* is a free-format field and is not validated by A2X; it may be left blank if not used by the trading Member.

2.3 Clearing Configuration

Before trading is permitted in the cash equities market, the Trading Member must supply their CSDP details to the A2X Settlement Committee. This is verified and set up by A2X as part of the Member’s trading configuration and settlement.

2.4 Cancel on Disconnect

Trading Members using ATP should note that ”cancel on disconnect” behaviour is in place on ATP sessions by default.

In other words, any open orders are cancelled as soon as a session disconnect is detected or if the user requests a session logout.

Note that after MaC lock time, matched order quantity is not cancelled by a disconnect or session logout.

2.5 Self-Trade Prevention

When a Member has multiple trading connections, A2X offers the option for self-trade prevention to be configured across these trading sessions. This prevents a Member from trading with themselves by cancelling the resting order that would otherwise match.

- Cancel Resting Order – If an incoming (or price modified) order would trade with another order from that Member that is already on the A2X order book, the existing resting order is automatically cancelled. The incoming order may trade with other orders, or is posted to the order book, as normal.

Trading Members who wish to use self-trade prevention should contact A2X support.

2.6 Post-Only and Post-Only Cancel Replace Order Types

Please note that Members must have self-trade prevention enabled in order for the POCR order type to function correctly.

If the order is cancelled back to prevent an aggressive trade the Order Add Response message will carry the status of Cancelled with cancel reason code 9 (post-only).

A PO/POCR order can be cancelled like any other order.

If a PO/POCR order is modified (for example to change price) it will only be updated on the order book if it will not trade on entry, otherwise the order will be cancelled back to the trading Member.

For further details on message formats see Section 3.4 and Section 4.

3 ATP Message Formats

This section provides details of the message formats used within the A2X Trading Protocol. This includes data types, message headers, message fields and descriptions.

3.1 Data Types

In all messages, 1-byte packing is used and all integers are represented in little-endian format.

Data type	Size	Value
char(n)	n	Left justified ascii string, padded with zero (0x00) to length n
u8	1	unsigned integer 0-255
u16	2	unsigned integer 0-65,535
u32	4	unsigned integer 0-4,294,967,295
u64	8	unsigned integer 0-2^64-1
Price	8	unsigned integer representing price with 5 decimal places implied i.e. value 1462500 represents a price of 14.625

Data type	Size	Value
Time	8	unsigned integer representing elapsed time in nanoseconds* since Unix epoch 00:00 UTC on 1st January 1970

3.2 Message Structure

All ATP messages carry a standard message header followed (for most message types) by a message body.

3.2.1 Message Header

All ATP messages carry a standard message header followed (for most message types) by a message body.

The header identifies the length and type of the message and, where appropriate, the sequence number of the message. The message body for a particular type of message is always a fixed length with all fields in a fixed order to support efficient creation and parsing of the binary ATP messages.

Field name	Type	Offset	Width	Comments
length	u16	0	2	Length of message including this header
msgType	u8	2	1	Message type
msgSeqNo	u32	3	4	Message sequence number from the user or A2X

3.3 Session Level Messages

All ATP session level messages are unsequenced. They carry the next sequence number that will be sent on the next business message in that stream (trading Member to A2X or A2X to trading Member). The value does not increment until a business level message is sent.

3.3.1 Login Message

The Login message is the initial message used to establish a trading session. It is also used to re-establish a session after a break.

Field name	Type	Offset	Width	Comments

Message Header		0	7	msgType = 1
protocolVersion	u16	7	2	A two byte field to represent the version of the ATP protocol that the user conforms to with major version number in the most significant byte and minor version number in the least significant byte. i.e. The current version v2.9 is LSB =9 MSB = 2 represented by value 0x0209
senderId	char(n)	9	16	Value as agreed with A2X to identify the Members ATP session
password	char(n)	25	16	Password as agreed with A2X to verify the user
inactivityTimeout	u16	41	2	Optional inactivity timeout in seconds. If there are no messages received for this period of time then A2X will close the session (triggering cancel on disconnect)
atpSeqNo	u32	43	4	The next sequence number from A2X that the trading Member is expecting. After a drop or break in the session this can be used by A2X to identify a gap and trigger any missed messages to be re-sent (see Login Response).

The *protocolVersion* is a two-byte field to represent the version of the ATP protocol that the user conforms to, with major version number in the most significant byte and minor version number in the least significant byte.

The latest version, v2.9, is represented by value 0x0209 i.e. MSB = 2, LSB = 9.

When the Member logs in for the first time in the day, the ATP port stores the protocol version stated in the login message and will reject any subsequent login messages that state a different protocol version. This is to ensure that all ATP traffic for that day uses the same protocol version.

When logging in, the *atpSeqNo* is used to specify the next sequence number from A2X that the trading Member is expecting. After a drop or break in the session, this can be used by A2X to identify a gap and trigger any missed messages to be re-sent (as detailed in the Login Response section below).

3.3.2 Login Response

The Login Response message is sent by A2X to acknowledge a Login request, and either accept or reject it.

Field name	Type	Offset	Width	Comments
Message Header		0	7	msgType = 2
resultCode	u8	7	1	Result of logon attempt 0 = Successful login 1 = Already logged in 2 = Sequence number error 3 = Unsupported protocol 4 = Failed authentication (incorrect password) 5 = Unknown Source Network
clientSeqNo	u32	8	4	Next sequence number A2X expects to receive

If the Login request message is invalid, A2X will set an appropriate *resultCode* in its Login Response. Note that if the *senderId* field is not recognised as a valid id for a session assigned to a known trading Member, or the Login originates from an unexpected source network, then A2X will drop the connection without sending any response.

If the Login request is re-establishing the connection after a break, there may be messages that the trading Member has missed (cancellation of any open orders or potentially any trade reports that were in process at the time of the disconnect).

A2X will compare its current business message sequence number with the *atpSeqNo* provided on the Login request to detect any gap, and will immediately send any missed messages to the trading Member. A2X will then send the Login Response carrying the current (next expected) A2X sequence number in its header. On receipt of this Login Response message, the trading Member knows that they have caught up with any missed messages and is now free to resume trading.

3.3.3 Heartbeat

A Heartbeat message is simply a message header with a heartbeat id as the message content. For a heartbeat, *msgType* is set to 0 and *msgSeqNo* is set to the sequence number that will be set on the next business message.

Field name	Type	Offset	Width	Comments
Message Header		0	7	msgType = 0

For example, pre-market, before any orders have been sent each Heartbeat message will carry *msgSeqNo* = 1. The value does not increment because the heartbeat is an unsequenced, session level message.

A2X will respond to a Heartbeat message with an outbound Heartbeat message to confirm receipt and the reliability of the connection.

3.3.4 Logout Request Message

The Logout Request message is used by the trading Member to request the closure of a trading session.

There is no message body required, simply a message header with *msgType* = 3.

Field name	Type	Offset	Width	Comments
Message Header		0	7	msgType = 3

3.3.5 Logout Message

The Logout message is usually sent as a response to the Logout Request however it may also be sent by A2X in case of a low sequence number or other protocol violation, or for other reasons.

The TCP/IP connection is closed immediately after this message has been sent.

Field name	Type	Offset	Width	Comments
Message Header		0	7	msgType = 4
reasonCode	u8	7	1	Reason for logout 0 = User requested 1 = Admin (market operations) 2 = Disconnect 3 = End of Day 4 = Inactivity timeout

				5 = Protocol error 6 = Sequence number error
reasonText	char(n)	8	32	Text describing reason for logout

3.4 Business Messages

3.4.1 Order Add Message

The Add Order message is sent by the trading Member to enter an order for a particular security.

Field name	Type	Offset	Width	Comments
Message Header		0	7	msgType = 24
securityId	u32	7	4	Numeric security identifier
orderType	u8	11	1	1 = Limit Order 2 = Auction on Demand (Limit Order) 3 = Auction on Demand (Trade at Middle/Limit or Pegged/Limit) 6 = Market at Close 8 = Post Only Cancel Replace (POCR) 9 = Post-Only (PO)
timeInForce	u8	12	1	Time in force for this order 1 = Day order 2 = Fill or Kill (FOK) 3 = Immediate or Cancel (IOC) 9 = Good for Auction (GFA) Only Day orders are valid for the Market at Close (MaC). GFA orders are only valid for the Auction on Demand

side	u8	13	1	1 = Buy order 2 = Sell order
quantity	u32	14	4	Number of shares
price	u64	18	8	Limit price of the order (0 for MaC order)
orderCapacity	u8	26	1	1 = Agency 2 = Principal
account	u16	27	2	Clearing account identifier possible values: 1 to 255 1=House Account >2=ID for agreed Client Account Code
userTag	u64	29	8	Free form tag assigned by trading Member
Reserved	u8	37	1	
Reserved	u8	38	1	
Reserved	u32	39	4	
Reserved	u8	43	1	
Reserved	u32	44	4	
Reserved	u8	48	1	
Reserved	u32	49	4	

3.4.2 Order Add Extended Message

The Order Add Extended message is sent by the trading Member to enter an order for a particular security with additional attributes.

Field name	**Type**	**Offset**	**Width**	**Comments**

Message Header		0	7	msgType = 25
securityId	u32	7	4	Numeric security identifier
orderType	u8	11	1	1 = Limit Order 2 = Auction on Demand Limit Order 3 = Auction on Demand Pegged Limit 6 = Market at Close 8 = Post Only Cancel Replace (POCR) 9 = Post-Only (PO)
timeInForce	u8	12	1	Time in force for this order 1 = Day order 2 = Fill or Kill (FOK) 3 = Immediate or Cancel (IOC) 9 = Good for Auction (GFA) Only Day orders are valid for the Market at Close (MaC). GFA orders are only valid for the Auction on Demand
side	u8	13	1	1 = Buy order 2 = Sell order
quantity	u32	14	4	Number of shares
price	u64	18	8	Limit price of the order (0 for MaC order)
orderCapacity	u8	26	1	1 = Agency 2 = Principal
account	u16	27	2	Clearing account identifier possible values: 1 to 65,535 1=House Account >2=ID for agreed Client Account Code

userTag	u64	29	8	Free form tag assigned by trading Member
Reserved	u8	37	1	
Reserved	u8	38	1	
Reserved	u32	39	4	
Reserved	u8	43	1	
Reserved	u32	44	4	
Reserved	u8	48	1	
Reserved	u32	49	4	
displayQuantity	u32	53	4	Display quantity for an iceberg order. Only valid for iceberg orders. For non-iceberg orders, value must be set to 0.
minQty	u32	57	4	Minimum quantity (shares) of an order to be executed. Only valid when OrderType = 2 or 3
Reserved	u8	61	1	

3.4.2.1 Iceberg Orders

Members can send an iceberg order by submitting an Order Add Extended message. The *displayQuantity* field specifies the quantity of the order to be displayed, the remainder of the order quantity is held in reserve and is not visible.

The display quantity of all orders at a given price level will be traded first and then the hidden quantity. When the display quantity has been fully traded it is refreshed from the reserve.

3.4.3 Order Cancel Message

The Order Cancel message is sent when a user wishes to cancel an open order.

Field name	Type	Offset	Width	Comments

Message Header		0	7	msgType = 7
orderRef	u32	7	4	Order reference number
userTag	u64	11	8	Free form tag assigned by trading Member

3.4.4 Order Modify Message

The Order Modify message is sent when a user wishes to modify an open order. Order quantity and/or limit price may be modified. Both values must be included, even if one of them is unchanged.

For MaC orders, the *orderCapacity* field can be modified up to the point when the trade is published.

NB orders in the AoD cannot be modified (only cancelled and subsequently replaced if required).

Field name	Type	Offset	Width	Comments
Message Header		0	7	msgType = 9
orderRef	u32	7	4	Order reference number
price	u64	11	8	The new price of the order
quantity	u32	19	4	The new order quantity
userTag	u64	23	8	Free form tag assigned by trading Member
Reserved	u8	31	1	
Reserved	u8	32	1	
Reserved	u32	33	4	
Reserved	u8	37	1	
Reserved	u32	38	4	
Reserved	u8	42	1	

Reserved	u32	43	4	
orderCapacity	u8	47	1	1 = Agency 2 = Principal

3.4.5 Order Modify Extended Message

The Order Modify Extended message is sent when a user wants to modify an open Order Add Extended. Order quantity and/or limit price and/or display quantity and/or minimum quantity may be modified. All values must be included, even if some are unchanged.

Field name	Type	Offset	Width	Comments
Message Header		0	7	msgType = 22
orderRef	u32	7	4	Order reference number
price	u64	11	8	The new price of the order
quantity	u32	19	4	The new order quantity
userTag	u64	23	8	Free form tag assigned by trading Member
Reserved	u8	31	1	
Reserved	u8	32	1	
Reserved	u32	33	4	
Reserved	u8	37	1	
Reserved	u32	38	4	
Reserved	u8	42	1	
Reserved	u32	43	4	
orderCapacity	u8	47	1	1 = Agency 2 = Principal

displayQuantity	u32	48	4	New display quantity of the iceberg order
Reserved	u32	52	4	

3.4.6 Order Add Response Message

A2X sends an Add Order Response message to acknowledge the receipt of an Order Add message.

The message is used to notify the trading Member whether or not their order was accepted and, if so, whether it executed (partially or fully) and whether any residual quantity has been added to the book or has been cancelled

Field name	Type	Offset	Width	Comments
Message Header		0	7	msgType = 6
orderRef	u32	7	4	Order reference number
marketDataId	u32	11	4	The id of this order as seen in the A2X market data. This value will be zero if the order is cancelled (IOC) or fully traded on entry
status	u8	15	1	Status of order and reject or cancel code (if needed) First 3 most significant bits - order status; Lower 5 bits - reason code (if rejected or cancelled); Otherwise zero. See Section 4 for values.
tradedQuantity	u32	16	4	Number of shares traded if any
timestamp	u64	20	8	Time that the order was accepted or rejected by the A2X system
userTag	u64	28	8	Free form tag as assigned by trading Member on the Order Add message

If the order traded on entry, i.e. *tradedQuantity* is not zero, then this message will be immediately followed by the related Trade message(s).

Note that this is correct behaviour even if the status is Cancelled, for example for an IOC order that partially trades. The Cancelled status confirms that the residual quantity has been cancelled back to the trading Member and the subsequent Trade message(s) provide details for the traded quantity.

3.4.7 Order Cancel Response Message

A2X sends an Order Cancel Response to accept or reject an Order Cancel message. The message is also used if an order is cancelled by Market Operations or due to cancel on disconnect.

For Market at Close orders, the message is used to cancel unmatched order quantity at lock time or if the match is cancelled for a particular security.

Field name	Type	Offset	Width	Comments
Message Header		0	7	msgType = 8
orderRef	u32	7	4	Order reference number
requestRef	u32	11	4	Sequence number (msgSeqNo) of the trading Members cancel request message. Note that this field will be zero in the case of a forced cancel by A2X Market Operations.
status	u8	15	1	Status of order and reject or cancel code (if needed) First 3 most significant bits - order status; Lower 5 bits - reason code (if rejected or cancelled); See Section 4 for values.
timestamp	Time	16	8	Time that the cancellation was accepted or rejected by the A2X system
userTag	u64	24	8	Free form tag as assigned by trading Member on the Order Cancel message

3.4.8 Order Modify Response Message

A2X sends an Order Modify Response to accept or reject an Order Modify or an Order Modify Extended message.

Field name	Type	Offset	Width	Comments
Message Header		0	7	msgType = 10
orderRef	u32	7	4	Order reference number
requestRef	u32	11	4	Sequence number (msgSeqNo) of the trading Members modify request message.

status	u8	15	1	Status of order and reject or cancel code (if needed)
				First 3 most significant bits - order status;
				Lower 5 bits - reason code;
				See Section 4 for values.

timestamp	u32	16	8	Time that the modification was accepted or rejected by the A2X system
userTag	u64	24	8	Free form tag as assigned by trading Member on the Order Modify message
Reserved	u8	32	1	

3.4.9 Iceberg Order Refresh Message

A2X sends an Iceberg Order Refresh message to notify the Member when an iceberg order is refreshed from the non-displayed reserve quantity. This new displayed quantity is published on A2X market data as a new order. The message displays the market data ID from the original order (*origA2xOrdId*) as well as the new market data ID for the new displayed quantity (*newA2xOrdId*), for tracking purposes against the market data feed.

Note that the *orderRef* and *origA2xOrdId* applies to the iceberg order as a whole and remains unchanged for the lifetime of the order.

Field name	Type	Offset	Width	Comments
Message Header		0	7	msgType = 23
orderRef	u32	7	4	Order reference number

origA2xOrdId	u32	11	4	The ID of the original order as seen in the A2X market data
newA2xOrdId	u32	15	4	The ID of the Iceberg order refresh as seen in the A2X market data
Quantity	u32	19	4	Refreshed display quantity of the iceberg order

3.4.10 Trade Capture Message

A2X uses a Trade Capture Report message for the purpose of allowing members to submit Large in Scale (LS), Negotiated Benchmark Cross (NBC) and Matched Principal (MP) trades to the exchange.

Field name	Type	Offset	Width	Comments
Message Header		0	7	msgType = 26
quantity	u32	7	4	Number of shares traded if any
price	u64	11	8	The price of the trade
securityId	u32	19	4	Numeric security identifier
tradeCaptureType	u8	23	1	1 = Large in Scale (LS) 3 = Negotiated Benchmark Cross (NBC) 5 = Matched Principal (MP)
Reserved	u8	24	1	
userTag	u64	25	8	Free form tag assigned by trading Member
Reserved	u64	33	8	
buyAccount	u16	41	2	Clearing account identifier possible values: 1 to 65,535 1=House Account

				>2=ID for agreed Client Account Code
buyOrderCapacity	u8	43	1	1 = Agency 2 = Principal
Reserved	u64	44	8	
sellAccount	u16	52	2	Clearing account identifier possible values: 1 to 65,535 1=House Account >2=ID for agreed Client Account Code
sellOrderCapacity	u8	54	1	1 = Agency 2 = Principal

3.4.11 Trade Capture Response Message

A2X sends a Trade Capture Response Message to accept or reject a Trade Capture Message.

Field name	Type	Offset	Width	Comments
Message Header		0	7	msgType = 18

status	u8	7	1	Status of order and reject or cancel code (if needed)
				First 3 most significant bits - order status;
				Lower 5 bits - reason code;
				See Section 4 for values.

tradeRef	u32	8	4	Trade reference number.
requestRef	u32	12	4	Sequence number (msgSeqNo) of the trading Member’s Trade Capture Message.
userTag	u64	16	8	Free form tag as assigned by trading Member on the Trade Capture Message.

3.4.12 Trade Message

The Trade message is published when an order executes. The Trade message is also used to communicate indicative trade reports at lock time for Market at Close orders. Note that the price and *tradeRef* values will be zero for indicative trade reports published when MaC orders are matched at lock time.

Field name	Type	Offset	Width	Comments
Message Header		0	7	msgType = 11
orderRef	u32	7	4	Order reference number
quantity	u32	11	4	Number of shares traded if any
price	u64	15	8	The price of the trade
side	u8	23	1	1 = Buy order 2 = Sell order
tradeRef	u32	24	4	Trade reference number
ccpCode	u8	28	1	Clearing CCP code 1 = Self clearing 2 = EMCFNL2A
liqIndicator	u8	29	1	1 = Added liquidity 2 = Removed liquidity 3 = Removed liquidity (hidden quantity)
securityId	u16	30	2	Numeric security identifier
timestamp	u64	32	8	Time that the trade occurred on the A2X system
userTag	u64	40	8	Free form tag as assigned by trading Member on the Order Add message or the most recent Order Modify message

Reserved	u8	48	1	

3.4.13 Trade Bust Message

If a trade is invalidated, a Trade Bust message is sent to notify the trading Member.

Field name	Type	Offset	Width	Comments
Message Header		0	7	msgType = 12
orderRef	u32	7	4	Order reference number
quantity	u32	11	4	Number of shares on invalidated trade
price	u64	15	8	The price of the invalidated trade
side	u8	23	1	1 = Buy order 2 = Sell order
tradeRef	u32	24	4	Trade reference number for the invalidated trade
timestamp	u64	28	8	Time that the trade was busted by A2X Market Operations

4 Status and Reason Codes

4.1 Order Status

This Order Status value is provided in the three most significant bits of the status byte field in the Order Add Response, Order Cancel Response and Order Modify Response messages.

Code	Order Status
1	Pending New (Internal use only)
2	Acknowledged

Code	Order Status
3	Cancelled
4	Rejected
5	Filled
6	Modified

For example Acknowledged for a day order is encoded as 0x40 (01000000), whereas Cancelled is encoded as 0x60 (01100000).

4.2 Modification Reasons

The Modification Reason code is provided in the lower five bits of the status byte field in the Order Modify Response message.

Please note that these codes only relate to version ATP 1.4 and above.

Code	Modification Reason
1	Modification Accepted
2	Order cancelled as a result of modification which updates the remaining quantity to zero

4.3 Cancel Reasons

The Cancel Reason code is provided in the lower five bits of the status byte field in the Order Cancel Response message. It is also provided in the Order Add Response message if the incoming order is unexpectedly cancelled on entry.

Code	Cancel Reason
1	Member request
2	A2X forced cancel
3	Market close

Code	Cancel Reason
4	Expired
5	*Reserved*
6	Aborted
7	Self-trade prevention
8	Cancel on disconnect
9	Post-only cancel (cancel to prevent aggressive trade)
10	Cancel residual quantity
11	Post-only cancel resting (cancel to prevent aggressive trade)
12	Minimum resting value

This is combined with the Cancelled order status, for example a cancel in response to a Trading Member's request is encoded as 01100001 (0x61) and a cancel of unmatched quantity at MaC lock time is encoded as 01100100 (0x64).

4.4 Reject Reasons

If an Order Add, or an Order Cancel or Order Modify request, is rejected then the Rejected order status is combined with a Reject Reason in the lower five bits of the status byte field.

The current set of reject reason codes are given below.

For example a reject because the given price does not conform to the tick table for the security would be encoded as 0x85 (10000101).

Code	Reject Reason
1	Not Authorised To Trade
2	Invalid Quantity
3	Invalid Price

Code	Reject Reason
4	Unknown Security
5	Price Does Not Conform To Tick
6	Invalid Order Type
7	Invalid Side
8	Invalid Order Capacity
9	Market Is Closed
10	Halted
11	Suspended
12	Invalid TimeInForce
13	Order Not Found / Not Open
14	No Clearing In Place
15	Failed Price Range Check
16	Invalid Clearing Account
17	Not Supported
18	Max Value Exceeded
19	Auction Ended
20	Drop Feed Is Down
21	Technical Reject
23	Stock Restricted

Code	Reject Reason
24	Minimum Consideration
27	Invalid
29	Bad Date
30	Duplicate
31	Reject Internal