ASX 24 Market Data Protocol Specification

Version 1.08 | February 2025

INFORMATION CLASSIFICATION - CONFIDENTIAL

ASX

Table of Contents

1. About This Document

This document describes the architecture and structure of the data streams used to make up the ASX Market Data Protocol (MDP) service.

1.1. Terms and Acronyms

Term/Acronym	Description
CFI code	The Classification of Financial Instruments (CFI) code is used to classify financial instruments in order to identify the type and characteristics of each financial instrument in accordance with international standards.
ISIN	International Securities Identification Number. Unique identifier issued to identify each financial instrument.
ISO 10962	An ISO standard maintained by ANNA (Association of National Numbering Agencies) acting as the Registration Authority.
MoldUDP64	A lightweight network protocol layer built on top of UDP that provides a mechanism for listeners to detect and re-request missed packets.
SoupBinTCP	A lightweight point-to-point network protocol, built on top of TCP/IP sockets that enable real-time delivery of sequenced messages in a client-server environment.
UTC	Coordinated Universal Time is a high-precision atomic standard often referred to as GMT.

1.2. Support Details

The Customer Technical Support (CTS) team operates a ‘service desk’ style support centre for ASX customers, which provides support coverage during business hours, 08:00 to 18:00 (AEST). Data content and test system problems are not supported during this period.

Email:	CTS@asx.com.au		
Phone:	Within Australia	—	1800 663 053
	Outside Australia	—	+612 9227 0372

After-hours support and escalation is provided via the ASX Production Services.

Phone:	Within Australia	—	+61 2 9227 0821

1.3. Version History

This document has been revised according to the table below:

Version	Date	Comment
V1.0	July 2016	Final release of specification.
V1.01	Sept 2016	• Added documentation errata and changes per delta document. • Updated document layout and style to match ASX 24 FIX specifications.
V1.02	April 2017	• Enhanced various message definitions as per update document • Layout changes to tables and pagination
V1.03	May 2017	• Removal of reference to dark, undisclosed and Centre Point orders. These were for equities usage which has been deferred.
V1.04	Aug 2017	• Clarify the dissemination of volatility in Option Symbol Directory (h) • Remove reference to volatility in Future Symbol Directory (f) • Clarify about OHLL (t) message for Glance requests during maintenance states
V1.05	Dec 2017	• Change of session state reason for "R" • Explicitly state that multicast sequence numbers and Glance implied sequence numbers do not correlate • State that snapshot complete sequence number may be lower than that received from multicast subscription
V1.06	Dec 2023	• Updated naming conventions
V1.07	Mar 2024	• Change of Data type for Requested Message count field
V1.08	Feb 2025	• Change the Price Minimum Tick field description for Combination Symbol Directory (M)

2. ASX Market Data Protocol Overview

The **ASX Market Data Protocol (MDP)** a high-speed market data service offering, which uses a binary message protocol to provide data in multicast form. The MDP service consists of three server components:

- **Multicast market data**: Provides reference and trading data disseminated in real time
- **Blink**: Provides the ability to recover multicast messages that are missed
- **Glance**: Provides a snapshot of the market on request of the subscriber

The components connect to subscribers at a network transport layer as follows:

- **Multicast market data**: A User Datagram Protocol (UDP) multicast protocol between the ASX Trading Multicast data server and subscribers. The transfer method of data uses the MoldUDP64 structure described in *3.1.1 Packet Header*. A data message header is used to carry each MDP message as described in *3.1.2 Data Message Header*.
- **Blink**: A UDP unicast protocol between the Blink server and subscribers. Subscribers detecting UDP multicast message loss can request missing messages as detailed *5.1.2 Request Packet*.
- **Glance**: A TCP/IP protocol between the Glance server and subscribers. The transfer of data uses the SoupBinTCP structure described *3. Administration*. A detailed explanation of Glance usage is provided in *5.1.3 Snapshot Recovery from Glance*.

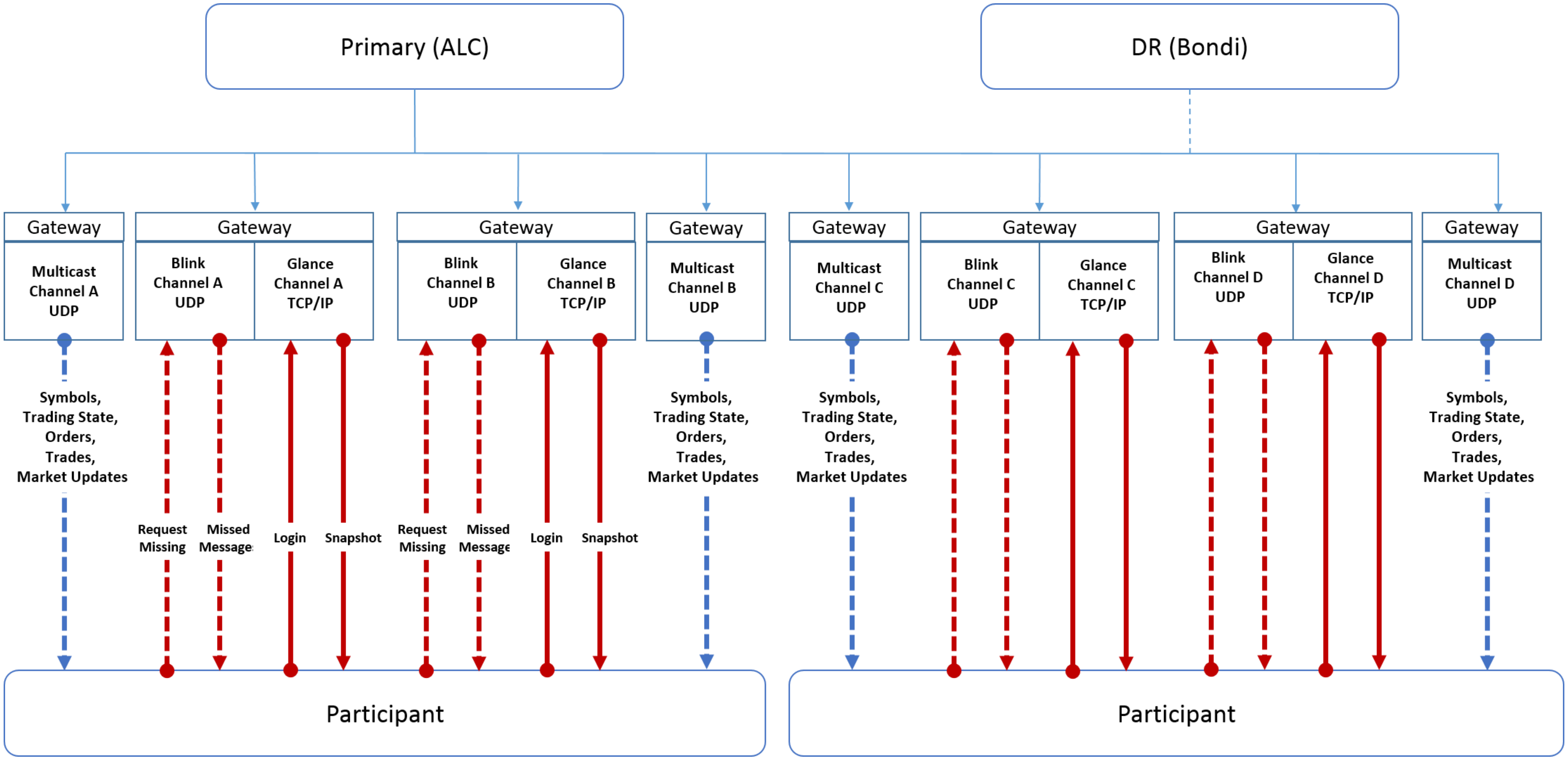
2.1. Sessions

A Session is a sequence of one or more messages. Each session is given a unique session identifier, which is transmitted in the MoldUDP64 packet header. A new session identifier is generated when the system is started and will last until the system shuts down. A new session identifier may be generated in the event of outages or maintenance that requires the system to be stopped and started. Once a new session identifier is seen, any market image created from the prior session's data is no longer valid and it is expected the subscriber will reset the market image as a result of receiving the new session number. Each new session will start its sequence number at 1. See *3.1.1 Packet Header* and *5.1 Multicast Session and Sequence* for further details.

2.2. Service Topology and Connection Recommendation

Services are available from the primary (ALC - Australian Liquidity Centre) and disaster recovery (Bondi) sites. At each site 2 channels are provided for each service. The ALC feeds are known as Feed A and Feed B. The Bondi feeds are Feed C and Feed D. Subscribers have the option to connect to both for full redundancy, or listen to one feed and switch over in the event of failure.

It is recommended that subscribers connect to both feeds and process simultaneously. If data is missing from one, use data from the other feed. This may obviate the need for a Blink request.


Primary (ALC)
DR (Bondi)
Gateway
Multicast Channel A UDP
Gateway
Blink Channel A UDP
Glance Channel A TCP/IP
Gateway
Blink Channel B UDP
Glance Channel B TCP/IP
Gateway
Multicast Channel B UDP
Gateway
Multicast Channel C UDP
Gateway
Blink Channel C UDP
Glance Channel C TCP/IP
Gateway
Blink Channel D UDP
Glance Channel D TCP/IP
Gateway
Multicast Channel D UDP
Symbols, Trading State, Orders, Trades, Market Updates
Request Missing
Missed Message
Login
Snapshot
Request Missing
Missed Message
Login
Snapshot
Symbols, Trading State, Orders, Trades, Market Updates
Symbols, Trading State, Orders, Trades, Market Updates
Symbols, Trading State, Orders, Trades, Market Updates
Participant
Participant

3. Message Protocol

The data feed is made up of a series of sequenced messages. Each message is variable in length based on the message type. The messages are delivered using a protocol that takes care of sequencing, stay-alive, and delivery guarantees. All messages delivered via multicast, or using either gap or full recovery, will use a defined packet header structure.

The packet header is used to deliver all administrative and application messages to and from the server on all communication channels. A packet header may contain zero, one, or more payload messages. While a packet header may contain multiple application messages, it will never contain more than one administrative message. A packet header will not contain both administrative and application messages.

3.1. Packet Header

All User Datagram Protocol (UDP) delivered messages will be self-contained. UDP delivered data will not exceed Ethernet packet boundary frames and a single Ethernet frame will contain only one packet header with associated data. Within each packet the data feed is comprised of a series of dynamically sized sequenced messages with each message defined using a message header indicating the length and type of each message.

Field Name	Offset	Length	Type	Notes
Session	0	10	Alpha	Identity of the multicast session the payload relates to.
Sequence	10	8	Numeric	Sequence Number of the first message to follow this header.
Count	18	2	Numeric	Number of messages to follow this header.
Payload	20	Variable		One or more payload data messages

3.1.1. Data Message Header

Each data message within the packet begins with Length and Message Type fields.

Field Name	Offset	Length	Type	Notes
Length	0	2	Numeric	Length of data message not including this field.
Message Type	2	1	Alpha	Code identifying this message type
Message Data	3	Variable		Message Content

3.1.2. Heartbeat

A Packet Header with a “Count” of zero will be used as a Heartbeat message. This message will never increment the sequence number and carries the sequence number of the next expected Packet Header for subscribers to detect a missing multicast message. To ensure subscribers detect a data feed, a Heartbeat is transmitted at least once per second. If no market data has been multicast in the last second, a Heartbeat will be multicast for subscribers to monitor multicast activity and detect message loss.

3.1.3. Data Types

All data elements used in packet headers, data message headers and payload messages will use the big-endian format for binary numeric data and left justified spaced padded for alphanumeric data. Numeric fields defined as bitwise flags will use a hexadecimal value indicating which bit flags are referenced. For instance, 0x04 will indicate the third least significant bit of the bit field.

The following time and date related fields have specific definitions:

- **Trade Date**: is defined as a number of days since 1970-01-01 and has range of 1970-01-01 to 2150-06-07 or 179 years. The “Trade Date” is used specifically to identify the trading period of each message. The current Trade Date will always be either the current business day or the next business day (typically for non-domestic or multi-session products). Transition between Trade Dates will occur in accordance with the relevant session schedule. To ensure there is no ambiguity in a data message, the Trade Date states the trading period of each message.

- **Second**: Used on the Time message is the date and time using “Unix Time” and is defined as the number of seconds since 1970-01-01 00:00:00 UTC and has a range of 1970-01-01 00:00:00 to 2038-01-19 03:14:07. The “Second” field in the Time message is used with the “TimeStamp” field to calculate the time of a data message.

- **Timestamp**: The nanosecond offset from the most recent Time message. The offset value is accurate to 1,000 nanoseconds.

Type	Length	Notes
Alpha	Variable	These fields are composed of non-control ISO 8859 (Latin-1) encoded bytes. They are left justified and padded on the right with spaces.
Delta	4	Big-Endian encoded 32 bit signed integer.
Numeric	1	8 bit unsigned integer.
Numeric	2	Big-Endian encoded 16 bit unsigned integer.
Numeric	4	Big-Endian encoded 32 bit unsigned integer.
Numeric	8	Big-Endian encoded 64 bit unsigned integer.
Price	8	Big-Endian encoded 64 bit signed integer.

3.2. Message Types

The following is the full list of message types supported, catering for the broad range of asset classes available.

Name	Type	Notes
Time Message	T	Precedes any application message where the second changes. Used by subsequent messages for time stamping.
End of Business Trade Date	S	Indicates the last message to be received for the specified business trade date.

Name	Type	Notes
State and Symbol Definition Messages		
Future Symbol Directory	f	Definition of the available future contracts disseminated at system start-up and as each instrument moves to a new Trade Date.
Option Symbol Directory	h	Definition of the available option contracts disseminated at system start-up and as each instrument moves to a new Trade Date.
Combination Symbol Directory	M	Definition of available Exchange defined combinations (intra-spreads, inter-spreads, strips, packs) disseminated at system start-up and as each instrument moves to a new Trade Date.
Bundles Symbol Directory	m	Definition of available Exchange defined combinations with greater than 6 legs (bundles) disseminated at system start-up and as each instrument moves to a new Trade Date.
Order Book State	O	Indicates the market status of an Order Book.
Order Book Messages		
Order Added	A	A new order is added to the order book.
Order Volume Cancelled	X	The volume of an order is adjusted down (no change to time priority).
Order Deleted	D	Order is removed from the order book
Order Executed	E	An order has fully or partially traded
Auction Order Executed	C	An order has fully or partially traded and the opposite order has fully traded during an auction uncrossing.
Combination Executed	e	A combination order has fully or partially traded with another resting order.
Implied Order Added	j	An implied order is added to the order book.
Implied Order Replaced	l	The price, volume, and/or priority of an implied order has been modified.
Implied Order Deleted	k	An implied order is removed from the order book.

Name	Type	Notes
Trade Messages		
Trade Executed	P	Indicates an execution of orders that were not pre-trade transparent.
Combination Trade Executed	p	Indicates an execution of orders that were not pre-trade transparent involving aggressive combinations matching with implied volume or undisclosed order volume.
Trade Cancellation	B	Indicates a previously published trade has been cancelled.
Market Data Messages		
Equilibrium Price (Auction Info)	Z	Reports the indicative equilibrium price and volume during the auction state
Open, High, Low, Last Trade Adjustment	t	Reports the Open, High, Low, Last, and volume statistics as an adjustment during snapshot downloads and at final close of contract.
Market Settlement	Y	Reports market settlement prices.
Text Message	x	Reports text messages.
Request for Quote	q	Reports RFQ messages.
Anomalous Order Threshold Publish	W	Reports the Extreme Trade Range (ETR) price and Anomalous Order Threshold (AOT) price.
Volume and Open Interest	V	Reports the Volume and Open Interest (VOI) information for the specified trade date.
Snapshot Complete	G	Reports the end of a snapshot download.

4. Application Data

4.1. Event Change Messages

4.1.1. Time Message (T)

A ‘seconds’ timestamp is issued every second and any message thereafter will be stamped with the number of nanoseconds past the last Time message received as a time reference point. The ‘seconds’ timestamp is the time as indicated by the matching engine at the time of processing

The server will transmit this message for every second during which at least one application message is generated. The time specified in this message serves as a reference for the times specified in subsequent non-Time messages until the next Time message is received. The timestamps in all non-Time Messages are specified as a nanosecond offset from the most recent Time message. The offset value is accurate to 1,000 nanoseconds.

Field Name	Offset	Length	Type	Notes
Message Type	0	1	Alpha	Set to “T”
Second	1	4	Numeric	A Timestamp – Second message will be disseminated for every second change where there is at least one payload message and is reported in seconds since 1970-01-01 00:00:00 UTC.
		5		

4.1.2. End of Business Trade Date (S)

This message will be sent as the last message for a Business Trade Date to signal no further messages will be disseminated for that date.

Field Name	Offset	Length	Type	Notes
Message Type	0	1	Alpha	Set to “S”
Timestamp	1	4	Numeric	Nanoseconds portion of the timestamp, accurate to 1,000 nanoseconds.
Trade Date	5	2	Numeric	Trade Date. Represented as number of days since 1 January 1970.
Event Code	7	1	Alpha	Valid values: “C” Business Trade Date has Ended. The trade date has completed and there will be no further messages transmitted with this trade date.
		8		

4.2. Reference Data Messages

4.2.1. Future Symbol Directory (f)

The Future Symbol Directory (f) message describes futures contracts. Attributes listed can be used to describe properties for display, valuation, description and order entry.

The message is disseminated when the following events occur:

- At system start-up, for all products
- As a product moves to a new trade date
- When a new contract is introduced intra-day

The Prior Day Settlement price can be modified using the Settlement (Y) message. Other fields that may be modified will be reported by re-issuing this message.

Field Name	Offset	Length	Type	Notes
Message Type	0	1	Alpha	Set to “f”
Timestamp	1	4	Numeric	Nanoseconds portion of the timestamp, accurate to 1,000 nanoseconds.
Trade Date	5	2	Numeric	Trade Date - Represented as number of days since 1 January 1970.
Tradeable Instrument Id	7	4	Numeric	Identifier of the Order Book.
Symbol name	11	32	Alpha	The unique tradable instrument name.
Long name	43	60	Alpha	Additional instrument series information.
ISIN	103	12	Alpha	Reserved for future use.
Exchange	115	6	Alpha	Denotes the exchange identifier of the contract
Instrument	121	6	Alpha	Denotes the instrument identifier of the contract.
CFI Code	127	6	Alpha	Indicates the type of security using ISO 10962 standard, CFI code values.
Expiry Year	133	2	Numeric	Year of expiry (YYYY) for futures contract types, zero otherwise.

Field Name	Offset	Length	Type	Notes
Expiry Month	135	1	Numeric	Month of expiry for futures contract types, zero otherwise. 1 – Jan (F) 2 – Feb (G) 3 – Mar (H) 4 – Apr (J) 5 – May (K) 6 – Jun (M) 7 – Jul (N) 8 – Aug (Q) 9 – Sep (U) 10 – Oct (V) 11 – Nov (X) 12 – Dec (Z).
Price Display Decimals	136	1	Numeric	Suggested number of decimals to display the price at.
Price Fractional Denominator	137	4	Numeric	Denominator of the price to yield the floating point price.
Price Minimum Tick	141	4	Numeric	Minimum tick size. Divide by the Price Fractional Denominator to determine the floating point value.
Last Trading Date	145	4	Numeric	The Last Trading Day as specified in the ASX 24 Contract Specifications. Reported in Unix Time (number of seconds since 00:00 1 January 1970 GMT). Convert to AEST/ADST as appropriate to determine the date.
Prior Day Settlement	149	8	Price	Prior Day Settlement price. Divide by the Price Fractional Denominator to determine the floating point value.
Currency	157	3	Alpha	Currency of contract.

Field Name	Offset	Length	Type	Notes
Lot Size or Face Value	160	8	Numeric	The Contract Unit as specified in the ASX 24 Contract Specifications. Divide by 1,000,000 to determine the floating point value. For example, ASX SPI Index Futures (commodity code AP), this will be 25000000, which will evaluate to 25.
Maturity Value	168	1	Numeric	The number of days or years to maturity for bills or bonds as specified in the ASX 24 Contract Specifications. As an example, for ASX 90 Day Bank Bills (commodity code IR) this is 90 and is in days. For ASX 10 Year Treasury Bond Futures (commodity code XT) this is 10 and in years.
Coupon Rate	169	2	Numeric	The coupon percentage rate for bonds. Divide by 100 to determine the floating point value. For example, 600 indicates 6%
Payments per Year	171	1	Numeric	Number of payments per year for bonds.
Block Lot Size	172	4	Numeric	Indicates the number of lots that represent a block lot for the order book. **Note**: A value of 0 indicates that this lot type is undefined for the order book.
Expiry Date	176	4	Numeric	The Settlement Date as specified in the ASX 24 Contract Specifications. Reported in Unix Time (number of seconds since 00:00 1 January 1970 GMT). Convert to AEST/ADST as appropriate to determine the date.
		180		

4.2.2. Option Symbol Directory (h)

The Options Symbol Directory (h) message describes options and warrants contracts. Attributes listed can be used to describe properties for display, valuation, description and order entry. The message is disseminated when the following events occur:

- At system start up, for all products

- As a product moves to a new trade date
- When a new contract is introduced intra-day

Field Name	Offset	Length	Type	Notes
Message Type	0	1	Alpha	Set to “h”
Timestamp	1	4	Numeric	Nanoseconds portion of the timestamp, accurate to 1,000 nanoseconds.
Trade Date	5	2	Numeric	Trade Date - Represented as number of days since 1 January 1970.
Tradeable Instrument Id	7	4	Numeric	Identifier of the Order Book.
Symbol name	11	32	Alpha	The unique tradable instrument name.
Long name	43	60	Alpha	Additional instrument series information.
ISIN	103	12	Alpha	Reserved for future use.
Exchange	115	6	Alpha	Denotes the exchange identifier of the contract
Instrument	121	6	Alpha	Denotes the instrument identifier of the contract.
CFI Code	127	6	Alpha	Indicates the type of security using ISO 10962 standard, CFI code values.
Expiry Year	133	2	Numeric	Year of expiry (YYYY).
Expiry Month	135	1	Numeric	Month of expiry: 1 – Jan (F) 2 – Feb (G) 3 – Mar (H) 4 – Apr (J) 5 – May (K) 6 – Jun (M) 7 – Jul (N) 8 – Aug (Q)

Field Name	Offset	Length	Type	Notes
				9 – Sep (U) 10 – Oct (V) 11 – Nov (X) 12 – Dec (Z).
Option Type	136	1	Alpha	Indicates the type of option: “P” – Put Option “C” – Call Option.
Strike	137	8	Price	Strike price of contract. Divide by the Strike Price Fractional Denominator to determine the floating point value.
Underlying Tradeable Instrument Id	145	4	Numeric	Identifier of the Underlying Tradeable Instrument (can be zero if underlying not defined).
Price Display Decimals	149	1	Numeric	Suggested number of decimals to display the price at.
Price Fractional Denominator	150	4	Numeric	Denominator of the price to yield the floating point price.
Price Minimum Tick	154	4	Numeric	Minimum tick size. Divide by the Price Fractional Denominator to determine the floating point value.
Strike Price Decimal Position	158	1	Numeric	Number of decimals to display the strike price.
Strike Price Fractional Denominator	159	4	Numeric	Denominator of the Strike Price to yield the floating point strike price.
Strike Price Minimum Tick	163	4	Numeric	Minimum tick size for the Strike Price. Divide by the Strike Price Fractional Denominator to determine the floating point value.

Field Name	Offset	Length	Type	Notes
Last Trading Date	167	4	Numeric	The Last Trading Day as specified in the ASX 24 Contract Specifications. Reported in Unix Time (number of seconds since 00:00 1 January 1970 GMT). Convert to AEST/ADST as appropriate to determine the date.
Prior Day Settlement	171	8	Price	Prior Day Settlement price. Divide by the Price Fractional Denominator to determine the floating point value.
Volatility	179	8	Numeric	Volatility for an option contract. When the Option Symbol Directory (h) message is transmitted at the start of a business day, this will contain the volatility from the last settlement, which has a prior trade date to the message. See Market Settlement (Y) message for volatility transmitted when settlement occurs. Option Symbol Directory (h) messages transmitted as part of a Glance request after settlement, but before the new trading date, will have volatility for the given trade date. Divide by 1,000,000 to determine the floating point value.
Currency	187	3	Alpha	Currency of contract.
Lot Size or Face Value	190	8	Numeric	The Contract Unit as specified in the ASX 24 Contract Specifications. Divide by 1,000,000 to determine the floating point value. For example, ASX SPI Index Futures (commodity code AP), this will be 25000000, which will evaluate to 25.
Maturity Value	198	1	Numeric	The number of days or years to maturity for bills or bonds as specified in the ASX 24 Contract Specifications. For ASX 90 Day Bank Bills (commodity code IR), this is 90 and is in days. For ASX 10 Year Treasury Bond Futures (commodity code XT), this is 10 and in years.
Coupon Rate	199	2	Numeric	The coupon percentage rate for bonds.

Field Name	Offset	Length	Type	Notes
				Divide by 100 to determine the floating point value. For example, 600 indicates 6%
Payments per Year	201	1	Numeric	Number of payments per year for bonds.
Block Lot Size	202	4	Numeric	Indicates the number of lots that represent a block lot for the order book. **Note**: A value of 0 indicates that this lot type is undefined for the order book.
Expiry Date	206	4	Numeric	The Settlement Day as specified in the ASX 24 Contract Specifications. Reported in Unix Time (number of seconds since 00:00 1 January 1970 GMT). Convert to AEST/ADST as appropriate to determine the date.
Basis of quotation	210	10	Alpha	Blank filled. Not available for ASX 24 products.
		220		

4.2.3. Combination Symbol Directory (M)

The Combination Symbol Directory is a specialised reference data message used for defining combinations with up to 6 legs. It represents both standard combinations defined by ASX and user defined combinations created by participants.

For Exchange defined combinations (Intra-, Inter- spreads, Packs and Strips) the associated underlying contract reference data will be disseminated prior to the available combination symbol directory messages. Where pre-defined combinations are defined across 2 or more instruments, e.g. inter-spreads, the dissemination will occur after the last instrument has moved to the next trade date.

Messages will be disseminated at system start-up, as each product moves to a new trade date, and as new combinations are introduced intra-day. Attributes listed can be used to describe properties for display, valuation, description and order entry.

Where a combination order book has less than 6 legs, the field values Order Book Id, Ratio and Price will be zeroed and Side will be space filled for the legs not defined.

Field Name	Offset	Length	Type	Notes
Message Type	0	1	Alpha	Set to “M”
Timestamp	1	4	Numeric	Nanoseconds portion of the timestamp, accurate to 1,000 nanoseconds.
Trade Date	5	2	Numeric	Trade Date - Represented as number of days since 1 January 1970.
Tradeable Instrument Id	7	4	Numeric	Identifier of the Order Book for this combination.
Symbol Name	11	32	Alpha	The unique tradable instrument name.
Long Name	43	60	Alpha	Additional instrument series information.
CFI Code	103	6	Alpha	Indicates the type of security using CFI code values.
Price Method	109	1	Numeric	Pricing method: 0 = net price (leg ratios are used to calculate the net price) 2 = Yield Difference (leg ratios greater than 1 do not affect the net price calculation) 3 = Individual (first leg is fixed price, remaining legs are net price) 4 = Average Price (legs are same side, i.e. bid buys all legs or ask sells all legs)
Price Display Decimals	110	1	Numeric	This value should be ignored. Use the highest Price Display Decimals from the symbol directory messages of the underlying leg instruments.
Price Fractional Denominator	111	4	Numeric	Denominator of the price to yield the floating point price.
Price Minimum Tick	115	4	Numeric	Minimum tick size. Divide by the Price Fractional Denominator to determine the floating point value. For UDC instruments, this value should be disregarded and instead derived from the lowest Price minimum Tick from the symbol directory messages of the underlying leg instruments.
Legs	119	1	Numeric	Number of legs.

Field Name	Offset	Length	Type	Notes
Tradeable Instrument Id Leg 1	120	4	Numeric	Order Book Identifier of this leg.
Side Leg 1	124	1	Alpha	Side of this leg: “B” – Buy Order “S” – Sell Order.
Ratio Leg 1	125	4	Numeric	Volume Ratio of this leg.
Price Leg 1	129	8	Price	Price of this leg. For exchange defined combinations this will always be zero. For user defined combinations this may be non-zero, depending on the Price Method. Divide by the Price Fractional Denominator of the instrument identified in Tradeable Instrument Id Leg 1 to determine the floating point value.
Tradeable Instrument Id Leg 2	137	4	Numeric	Order Book Identifier of this leg.
Side Leg 2	141	1	Alpha	Side of this leg: “B” – Buy Order “S” – Sell Order.
Ratio Leg 2	142	4	Numeric	Volume Ratio of this leg.
Price Leg 2	146	8	Price	Price of this leg. Always zero.
Tradeable Instrument Id Leg 3	154	4	Numeric	Order Book Identifier of this leg.
Side Leg 3	158	1	Alpha	Side of this leg: “B” – Buy Order “S” – Sell Order <space> if leg is not defined.

Field Name	Offset	Length	Type	Notes
Ratio Leg 3	159	4	Numeric	Volume Ratio of this leg.
Price Leg 3	163	8	Price	Price of this leg. Always zero.
Tradeable Instrument Id Leg 4	171	4	Numeric	Order Book Identifier of this leg.
Side Leg 4	175	1	Alpha	Side of this leg: "B" – Buy Order "S" – Sell Order <space> if leg is not defined.
Ratio Leg 4	176	4	Numeric	Volume Ratio of this leg.
Price Leg 4	180	8	Price	Price of this leg. Always zero.
Tradeable Instrument Id Leg 5	188	4	Numeric	Order Book Identifier of this leg.
Side Leg 5	192	1	Alpha	Side of this leg: "B" – Buy Order, "S" – Sell Order or <space> if leg is not defined.
Ratio Leg 5	193	4	Numeric	Volume Ratio of this leg.
Price Leg 5	197	8	Price	Price of this leg. Always zero.
Tradeable Instrument Id Leg 6	205	4	Numeric	Order Book Identifier of this leg.
Side Leg 6	209	1	Alpha	Side of this leg: "B" – Buy Order, "S" – Sell Order or

Field Name	Offset	Length	Type	Notes
				<space> if leg is not defined.
Ratio Leg 6	210	4	Numeric	Volume Ratio of this leg.
Price Leg 6	214	8	Price	Price of this leg. Always zero.
		222		

4.2.4. Bundles Symbol Directory (m)

The Bundles Symbol Directory is a specialised reference data message used for defining combinations with legs greater than 6, specifically Bundle products with legs of 8, 12, 16 or 20 legs.

Messages will be disseminated at system start-up, as each product moves to a new trade date, and as new combinations are introduced intra-day. Attributes listed can be used to describe properties for display, valuation, description and order entry.

Where a combination order book has less than 20 legs, the field values Order Book Id, Ratio and Price will be zeroed and Side will be space filled for legs not defined.

Field Name	Offset	Length	Type	Notes
Message Type	0	1	Alpha	Set to “m”
Timestamp	1	4	Numeric	Nanoseconds portion of the timestamp, accurate to 1,000 nanoseconds.
Trade Date	5	2	Numeric	Trade Date - Represented as number of days since 1 January 1970.
Tradeable Instrument Id	7	4	Numeric	Order Book Identifier for this combination definition.
Symbol Name	11	32	Alpha	The unique tradeable combination name.
Long Name	43	60	Alpha	Additional instrument series information.

Field Name	Offset	Length	Type	Notes
CFI Code	103	6	Alpha	Indicates the type of security using ISO 10962 standard, CFI code values.
Price Method	109	1	Numeric	Pricing method: 0 = net price (leg ratios are used to calculate the net price) 2 = Yield Difference (leg ratios greater than 1 do not affect the net price calculation) 3 = Individual (first leg is fixed price, remaining legs are net price) 4 = Average Price (legs are same side, i.e. bid buys all legs or ask sells all legs)
Price Display Decimals	110	1	Numeric	Suggested number of decimals to display the price at.
Price Fractional Denominator	111	4	Numeric	Denominator of the price to yield the floating point price.
Price Minimum Tick	115	4	Numeric	Minimum tick size. Represented as a signed integer. Divide by the Price Fractional Denominator to determine the floating point value.
Legs	119	1	Numeric	Number of legs.
Tradeable Instrument Id Leg 1	120	4	Numeric	Order Book Identifier of this leg.
Side Leg 1	124	1	Alpha	Side of this leg: “B” – Buy Order, “S” – Sell Order.
Ratio Leg 1	125	4	Numeric	Volume Ratio of this leg.
Price Leg 1	129	8	Price	Price of this leg. Always zero. Divide by the Price Fractional Denominator of the instrument, identified in Tradeable Instrument Id Leg 1, to determine the floating point value.
Tradeable Instrument Id Leg 2	137	4	Numeric	Order Book Identifier of this leg.

Field Name	Offset	Length	Type	Notes
Side Leg 2	141	1	Alpha	Side of this leg: “B” – Buy Order, “S” – Sell Order.
Ratio Leg 2	142	4	Numeric	Volume Ratio of this leg.
Price Leg 2	146	8	Price	Price of this leg. Always zero.
…				
Tradeable Instrument Id Leg 20	443	4	Numeric	Order Book Identifier of this leg.
Side Leg 20	447	1	Alpha	Side of this leg: “B” – Buy Order “S” – Sell Order <space> if leg is not defined.
Ratio Leg 20	448	4	Numeric	Volume Ratio of this leg.
Price Leg 20	452	8	Price	Price of this leg. Always zero.
		460		

4.2.5. Representation of decimal numbers

All number values are represented without a decimal place. Fields with the Price data type are converted to a decimal value by dividing by the Price Fractional Denominator, transmitted in symbol directory messages. Fields with Numeric data types are treated as whole number values except where stated otherwise in the description.

For example if the value of a field with Price data type is 97175000 and the Price Fractional Denominator is 1,000,000 then the decimal value is 97.175000.

Two other fields are useful in understanding prices. The decimal value of a price is further validated using the Price Minimum Tick value which is the smallest increment a price can change by. The number of significant digits for a price is the number of significant digits it takes to display the Price Minimum Tick. The Price Display Decimals contains the number suggested decimal digits past the decimal point to be displayed in a price.

If for example, the contract has Price Display Decimals defined as 3 and the decimal value of the Price Minimum Tick is 0.025000, then applying the Price Displays Decimals value will show Price Minimum Tick as 0.025, and the price 97.175000 from the above example, yields 97.175. Note that the Price Minimum Tick also allows prices to be validated. With a tick size of 0.025, 97.175 is a valid price but 97.176 is not.

4.3. State Change Messages

4.3.1. Order Book State (O)

This message indicates the current session state of a contract to the market.

Field Name	Offset	Length	Type	Notes
Message Type	0	1	Alpha	Set to "O"
Timestamp	1	4	Numeric	Nanoseconds portion of the timestamp, accurate to 1,000 nanoseconds.
Trade Date	5	2	Numeric	Trade Date - Represented as number of days since 1 January 1970.
Tradeable Instrument Id	7	4	Numeric	Identifier of the Order Book.
Session State	11	1	Alpha	Indicates the current session state for the contract as per *4.3.2 Session State Explanation*.
		12		

4.3.2. *Session State Explanation*

The following codes and description will be made available as part of the reference data for a state.

Code	Trading Status Explanation
“P”	Pre-Open
“O”	Opened
“R”	Regulatory halt
“H”	Halted
“C”	Closed
“M”	Maintenance

4.4. Order Messages

Each order accepted by the matching engine is assigned 2 order identifiers, a private order id, which is unique for the life of the order, and a public order id, which is reported on the public order feeds. The owner of the order will receive the private and public order ids and can use the public order id to track the order on the public data feeds.

Modifications to orders will result in a new Public Order Id being assigned for that order. On the public order data feeds, orders that change priority will be reported as two transactions, an Order Delete (“D”) and an Order Added (“A”).

Where an order is modified for volume down, this is reported as an Order Volume Cancelled (“X”).

4.4.1. *Order Added (A)*

An Order Added message indicates a new order or a restated order that has been accepted by the matching engine and added to the order book.

At the time of creation, the Order Book Priority reported in this message represents the time priority in this order book.

Field Name	Offset	Length	Type	Notes
Message Type	0	1	Alpha	Set to “A”
Timestamp	1	4	Numeric	Nanoseconds portion of the timestamp, accurate to 1,000 nanoseconds.
Trade Date	5	2	Numeric	Trade Date - Represented as number of days since 1 January 1970.
Tradeable Instrument Id	7	4	Numeric	Identifier of the Order Book.
Side	11	1	Alpha	Type of order: “B” – Buy Order “S” – Sell Order.
Order Id	12	8	Numeric	Public order id of the order.
Order Book Priority	20	8	Numeric	Time priority of this order within the order book.
Quantity	28	4	Numeric	Number of lots added to the book (all volume is visible).
Price	32	8	Price	Price of the order.
		40		

4.4.2. Order Volume Cancelled (X)

An Order Volume Cancelled message is sent when an order is modified as a result of partial cancellation of volume (downward quantity amend). The priority of the order will not be changed.

An Order Volume Cancelled message is not sent when an order has partially traded.

Field Name	Offset	Length	Type	Notes
Message Type	0	1	Alpha	Set to 'X'
Timestamp	1	4	Numeric	Nanoseconds portion of the timestamp, accurate to 1,000 nanoseconds.
Trade Date	5	2	Numeric	Trade Date - Represented as number of days since 1 January 1970.
Tradeable Instrument Id	7	4	Numeric	Identifier of the Order Book.
Side	11	1	Alpha	Type of order. 'B' – Buy 'S' – Sell.
Order Id	12	8	Numeric	Public order id of the order.
Quantity	20	4	Numeric	New displayed quantity of order.
		24		

4.4.3. Order Deleted (D)

This message is sent whenever an order in the book is cancelled by the owner, expired, cancelled by the exchange, purged or volume cancelled. All remaining volume is no longer accessible so the order must be removed from the book.

An Order Deleted message is not sent when an order has traded out.

Field Name	Offset	Length	Type	Notes
Message Type	0	1	Alpha	Set to “D”
Timestamp	1	4	Numeric	Nanoseconds portion of the timestamp, accurate to 1,000 nanoseconds.

Field Name	Offset	Length	Type	Notes
Trade Date	5	2	Numeric	Trade Date - Represented as number of days since 1 January 1970.
Tradeable Instrument Id	7	4	Numeric	Identifier of the Order Book.
Side	11	1	Alpha	Type of order: “B” – Buy Order “S” – Sell Order.
Order Id	12	8	Numeric	Public order id of the order being removed.
		20		

4.4.4. Order Executed (E)

This message is sent whenever a resting order of the book is executed in whole or in part and the counterparty to the trade is an aggressive order. It is possible to receive several Order Executed messages for the same order reference number if that order is executed in several parts.

The Order Executed reports a trade, updates trade statistics, and requires action on the order book to update order volume or delete an order. The order reported in the message can be an order book of an equity, future, option or combination. For combination order books, the trade price reported may not be the same as the price of the order due to rounding, e.g. inter-spread combination orders with different tick sizes.

The opposite order is always an order the subscriber has not seen such as an aggressive order.

Where the order book is non-anonymous, the Counter Party Id is the owner of the opposite order.

- The order reported in this message is always the passive order traded.
- The trade side of the order is the same as the “Side”, i.e. Buy Order is also the Buyer side of the trade.
- The trade price is included in this message as the price of the order could vary from the order price.
- For anonymous markets, Counter Party Id will be blank filled.

The Combination Trade Id is used to identify or link a combination order with the traded legs to assist the subscriber to track the matching sequence. If the Combination Trade Id is zero, then this is a non-Combination execution.

The Order Executed message is sent under the following circumstances:

- Only during the continuous trading period
- Where a match affects visible orders in an order book

Where an action causes an Order Executed message to be generated, it is possible other messages will be reported:

- Where a match is due to an existing order modification, order executions will always follow the Delete Order message
- Where an aggressive order creates a match with remaining visible volume, order executions will always precede the resulting Order Added message.
- Order executions will precede any implied order activity reporting.

Field Name	Offset	Length	Type	Notes
Message Type	0	1	Alpha	Set to “E”
Timestamp	1	4	Numeric	Nanoseconds portion of the timestamp, accurate to 1,000 nanoseconds.
Trade Date	5	2	Numeric	Trade Date - Represented as number of days since 1 January 1970.
Tradeable Instrument Id	7	4	Numeric	Identifier of the Order Book.
Side	11	1	Alpha	Type of order: “B” – Buy Order “S” – Sell Order.
Order Id	12	8	Numeric	Public order id of the order on the passive side. All trade details refer to this order.
Quantity Remaining	20	4	Numeric	Volume remaining on Order after matching, if 0, order has traded out and should be deleted.
Trade Type	24	1	Alpha	Nature of the match as per *4.6.4 Trade Type Explanation*

Field Name	Offset	Length	Type	Notes
Trade Id	25	8	Numeric	Public trade id of the trade.
Executed Quantity	33	4	Numeric	Volume traded.
Trade Price	37	8	Price	Price of trade (may be negative for combination trades).
Combination Trade Id	45	8	Numeric	A non-zero value indicates that this trade is part of an execution cycle for a combination trade. This allows subscribers to trace the leg trades back to the originating combination execution using the Combo Trade Id as the linking element. For the combination trade itself, the trade id and the Combo Trade Id are the same.
Counter Party Id	53	3	Alpha	Participant identifier of the aggressive party to the execution in non-anonymous markets. Space filled for anonymous instruments.
		56		

4.4.5. Auction Order Executed (C)

This message is sent when orders are executed in the auction at the open of the market. The order information reported will either have the volume adjusted or the order will be removed from the order book if the Quantity Remaining is zero. Where the Opposite Order Id is non-zero, the order referenced will always be removed from the same order book from the opposite side.

It is possible to receive multiple “Auction Order Executed” messages for the same order if that order is executed in several parts during the course of matching. As each message is received the remaining volume will reflect the volume as a result of that trade.

The Auction Order Executed message will only be generated and sent during the auction period.

Both buying and selling information is contained in this message indicating the subscriber must update for both orders.

Field Name	Offset	Length	Type	Notes
Message Type	0	1	Alpha	Set to “C”
Timestamp	1	4	Numeric	Nanoseconds portion of the timestamp, accurate to 1,000 nanoseconds.
Trade Date	5	2	Numeric	Trade Date - Represented as number of days since 1 January 1970.
Tradeable Instrument Id	7	4	Numeric	Identifier of the Order Book.
Side	11	1	Alpha	Type of order: “B” – Buy Order “S” – Sell Order.
Order Id	12	8	Numeric	Public order id of order.
Quantity Remaining	20	4	Numeric	Volume remaining on Order after matching, if 0 order has traded out and should be deleted.
Trade Type	24	1	Alpha	Nature of the match as per *4.6.4 Trade Type Explanation*
Trade Id	25	8	Numeric	Public trade id of the trade.
Executed Quantity	33	4	Numeric	Volume traded.
Trade Price	37	8	Price	Price of trade (may be negative for Combination trades).
Opposite Order Id	45	8	Numeric	Public Order Id of the order to be removed from the opposite side of the same Order Book id, as order has traded out.
		53		

4.4.6. Combination Order Executed (e)

This message is sent whenever a resting order is matched with an open leg of a resting combination order in whole or in part. It is possible to receive several Combination Executed messages for the same order if that order is executed in several parts.

Where a combination order book trades either as a combination-to-combination or combination-to-underlying, an additional Order Executed message is reported to indicate the combination traded price.

As this message involves 2 resting orders where at least one of the resting orders is a combination order, the opposite side is reported to provide sufficient information to track which order is party to the trade. The opposite order details are informational only and are not be used to update the order book. An Order Executed or Combination Executed will precede or follow indicating the order update. Both orders reported in this message are always the passive order traded.

Field Name	Offset	Length	Type	Notes
Message Type	0	1	Alpha	Set to "e"
Timestamp	1	4	Numeric	Nanoseconds portion of the timestamp, accurate to 1,000 nanoseconds.
Trade Date	5	2	Numeric	Trade Date - Represented as number of days since 1 January 1970.
Tradeable Instrument Id	7	4	Numeric	Identifier of the (Combination) Order Book
Side	11	1	Alpha	Type of order: "B" – Buy Order "S" – Sell Order.
Order Id	12	8	Numeric	Public order id of the order
Quantity Remaining	20	4	Numeric	Volume remaining on Order after matching, if 0, order has traded out and should be deleted
Trade Type	24	1	Alpha	Nature of the match as per *4.6.4 Trade Type Explanation*
Trade Id	25	8	Numeric	Public trade id of the trade.
Executed Quantity	33	4	Numeric	Volume traded.
Trade Price	37	8	Price	Price of trade.
Opposite Tradeable Instrument Id	45	4	Numeric	Identifier of the opposite Tradeable Instrument Id (references the combination Tradeable Instrument Id party to this trade).

Field Name	Offset	Length	Type	Notes
Opposite Side	49	1	Alpha	Type of order: "B" – Buy Order "S" – Sell Order.
Opposite Order Id	50	8	Numeric	Public order id of the opposite order.
Combination Trade Id	58	8	Numeric	A non-zero value indicates that this trade is part of an execution cycle for a combination trade. This allows subscribers to trace the leg trades back to the originating combination execution using the Combo Trade Id as the linking element.
		66		

4.5. Implied Order Messages

Reporting of implied prices derived from spreadable contracts is a special feature of market reporting and gives the market a clear picture of the available volume at the best price. To ensure a full market picture is reported, implied prices are reported using a special order type called an "Implied Order". Implied orders can only be traded or modified indirectly.

The Implied Order Added message will refer to a single "implied" order and the Order Id assigned will always be unique for the life of the implied order. The priority assigned to an implied order will be based on a "real" order and although the same priority can be referenced in multiple order books, it will always be unique within the same order book and side when compared to the priority of the "real" orders within the order book and side. However, the priority can be a duplicate with other implied orders in the same order book and side. If this is the case, the Order Id of the implied order will be the determining factor in regards to the position in the queue.

There can be implied orders reported for an order book with no "real" orders and for reporting purposes, the price and volume of the Tradeable Instrument is denoted as being "implied" if the top of book order is an "implied" order.

Due to the nature of implied price reporting, it is possible to receive Implied Order Replaced messages for an existing implied order even though the price, priority or volume of the implied order does not change.

To remove implied orders, the Implied Order Deleted message is issued to remove only the implied orders for a specific order book and side. Once an implied order is removed from the order book, the order id referencing the removed implied order is available for reuse in a subsequent update to the implied pricing for that order book side.

4.5.1. Implied Order Added (j)

This message is a similar format to the Order Added message, except the nature of this order is it has an implied price, implied volume and a priority that is derived from two or more "real" orders deriving the implied price.

Field Name	Offset	Length	Type	Notes
Message Type	0	1	Alpha	Set to "j"
Timestamp	1	4	Numeric	Nanoseconds portion of the timestamp, accurate to 1,000 nanoseconds.
Trade Date	5	2	Numeric	Trade Date - Represented as number of days since 1 January 1970.
Tradeable Instrument Id	7	4	Numeric	Identifier of the Order Book.
Side	11	1	Alpha	Type of order: "B" – Buy Order "S" – Sell Order.
Order Id	12	8	Numeric	Public order id of implied order.
Order Book Priority	20	8	Numeric	Time priority within the order book.
Quantity	28	4	Numeric	Number of implied lots added to the book (All volume is visible).
Price	32	8	Price	Implied price of order.
		40		

4.5.2. *Implied Order Replaced (l)*

This message notifies the subscriber that an existing implied order is to be replaced when the implied price, volume or priority has changed due to a price or volume movement of “real” orders for a side of a single Tradeable Instrument Id.

Field Name	Offset	Length	Type	Notes
Message Type	0	1	Alpha	Set to “l” (lowercase L)
Timestamp	1	4	Numeric	Nanoseconds portion of the timestamp, accurate to 1,000 nanoseconds.
Trade Date	5	2	Numeric	Trade Date - Represented as number of days since 1 January 1970.
Tradeable Instrument Id	7	4	Numeric	Identifier of the Order Book.
Side	11	1	Alpha	Type of order: “B” – Buy Order “S” – Sell Order.
Order Id	12	8	Numeric	Public order id of implied order.
Order Book Priority	20	8	Numeric	Time priority within the order book.
Quantity	28	4	Numeric	Number of implied lots replaced in this book (All volume is visible).
Price	32	8	Price	Implied price of order.
		40		

4.5.3. *Implied Order Deleted (k)*

This message is the same format as the Order Deleted message, except this deletion refers to an implied order.

This message is sent whenever the implied pricing or volume has changed due to price or volume movement of “real” orders for a side of a single contract number. When received, any implied volume is no longer accessible so the “implied” order of the indicated side must be removed from the book.

Field Name	Offset	Length	Type	Notes
Message Type	0	1	Alpha	Set to “k”
Timestamp	1	4	Numeric	Nanoseconds portion of the timestamp, accurate to 1,000 nanoseconds.
Trade Date	5	2	Numeric	Trade Date - Represented as number of days since 1 January 1970.
Tradeable Instrument Id	7	4	Numeric	Identifier of the Order Book.
Side	11	1	Alpha	Type of order: “B” – Buy Order “S” – Sell Order.
Order Id	12	8	Numeric	Public order id of implied order.
		20		

4.6. Trade Messages

4.6.1. Trade Executed (P)

The Trade Message is designed to provide details for executions in combination order books. Trade Executed messages are reported in the same sequence as order execution messages are reported.

The Combination Trade Id is used to identify or link a combination order with the traded legs to assist the subscriber to track the matching sequence. If the Combination Trade Id is zero, then this is a non-Combination execution.

Field Name	Offset	Length	Type	Notes
Message Type	0	1	Alpha	Set to “P”
Timestamp	1	4	Numeric	Nanoseconds portion of the timestamp, accurate to 1,000 nanoseconds.

Field Name	Offset	Length	Type	Notes
Trade Date	5	2	Numeric	Trade Date - Represented as number of days since 1 January 1970.
Tradeable Instrument Id	7	4	Numeric	Identifier of the Order Book.
Trade Type	11	1	Alpha	Nature of the match as per *4.6.4 Trade Type Explanation*
Trade Id	12	8	Numeric	Public trade id of the trade.
Executed Quantity	20	4	Numeric	Volume traded.
Trade Price	24	8	Price	Price of trade.
Combination Trade Id	32	8	Numeric	A non-zero value indicates that this trade is part of an execution cycle for a combination trade. This allows subscribers to trace the leg trades back to the originating combination execution using the Combo Trade Id as the linking element. For the combination trade itself, the trade id and the Combo Trade Id are the same.
Participant Id Buyer	40	3	Alpha	Participant identifier of the owner of the buying order. Blank for anonymous instruments.
Participant Id Seller	43	3	Alpha	Participant identifier of the owner of the selling order. Blank for anonymous instruments.
		46		

4.6.2. Combination Trade Executed (p)

The Combination Trade Executed Message is designed to provide details for executions where one or both orders are resting from a different order book id from the order book id of the trade. This message covers reporting of combination matches with aggressive combination orders, or undisclosed combination volume as a part of a spread chain.

Since Combination Trade Messages do not affect the displayed book, they may be ignored by subscribers looking to build and track the order book view. For tracking order activity, the buyer and seller order details are made available.

Field Name	Offset	Length	Type	Notes
Message Type	0	1	Alpha	Set to “p”
Timestamp	1	4	Numeric	Nanoseconds portion of the timestamp, accurate to 1,000 nanoseconds.
Trade Date	5	2	Numeric	Trade Date - Represented as number of days since 1 January 1970.
Tradeable Instrument Id	7	4	Numeric	Identifier of the Order Book.
Trade Type	11	1	Alpha	Nature of the match as per *4.6.4 Trade Type Explanation*
Trade Id	12	8	Numeric	Public trade id of the trade.
Executed Quantity	20	4	Numeric	Volume traded.
Trade Price	24	8	Price	Price of trade (may be negative for spread order books).
Buyer Tradeable Instrument Id	32	4	Numeric	Identifier of the Buyers Order Book.
Buyer Side	36	1	Alpha	Type of order: “B” – Buy Order “S” – Sell Order.
Buyer Order ID	37	8	Numeric	Reference number of the Buyer’s order – 0 if order is aggressive.
Buyer Combination Trade Id	45	8	Numeric	A non-zero value indicates that this trade is part of an execution cycle for a combination trade from the E or P message. This allows subscribers to trace the leg trades back to the originating combination execution using the Combo Trade Id as the linking element.
Buyer Participant Id	53	3	Alpha	Buying participant identifier of the owner of the aggressive order. Blank for anonymous instruments.

Field Name	Offset	Length	Type	Notes
Seller Tradeable Instrument Id	56	4	Numeric	Identifier of the Seller's Order Book.
Seller Side	60	1	Alpha	Type of order: "B" – Buy Order "S" – Sell Order.
Seller Order Id	61	8	Numeric	Reference number of the Seller's order – 0 if order is aggressive.
Seller Combination Trade Id	69	8	Numeric	A non-zero value indicates that this trade is part of an execution cycle for a combination trade from the E or P message. This allows subscribers to trace the leg trades back to the originating combination execution using the Combo Trade Id as the linking element.
Seller Participant Id	77	3	Alpha	Selling participant identifier of the owner of the aggressive order. Blank for anonymous instruments.
		80		

4.6.3. Trade Cancellation (B)

This message is sent whenever a trade is cancelled by ASX. This message does not affect any resting orders and is for information purposes. Adjustments to the Open, High, Low, Last will be performed as separate action using the Open, High, Low, Last Trade Adjustment message.

Field Name	Offset	Length	Type	Notes
Message Type	0	1	Alpha	Set to "B"
Timestamp	1	4	Numeric	Nanoseconds portion of the timestamp, accurate to 1,000 nanoseconds.

Field Name	Offset	Length	Type	Notes
Trade Date	5	2	Numeric	Trade Date - Represented as number of days since 1 January 1970.
Tradeable Instrument Id	7	4	Numeric	Identifier of the Order Book.
Trade Id	11	8	Numeric	Public trade id of the trade.
		19		

4.6.4. *Trade Type Explanation*

These codes will assist the subscriber in determining the nature of the execution made such as if the match was crossed, as a result of an auction, etc.

Code	Code if Crossing	Notes
“T”	“t”	Normal Trade, an outright-to-outright trade.
“L”	“l”	Auction Trade, trade price may be different than price of resting order(s).
“S”	“s”	Combination-to-Underlying trade (price is based on the order price of an underlying order book relating to intra-spread, inter-spread, or user defined combination matching with an outright order).
“R”	“r”	Combination-to-Combination trade (price is based on the current indicative pricing from the underlying order book).
“A”	“a”	Strip to Strip Trade: Indicates an Average Price Type (e.g. pack, and bundle) combination order has matched with another Average Price Type combination order.
“B”	“b”	Strip to Outright Trade: Indicates an Average Price Type combination order has matched with an outright order.

4.7. Market Updates

4.7.1. *Equilibrium Price (Auction Info) (Z)*

This message is disseminated in these circumstances:

- In response to changes in the equilibrium price for an order book during a pre-open period. As orders are added, modified, or cancelled, a message will be reported informing the market of the potential opening price. Note that subtracting the Ask Quantity from the Bid Quantity will yield the Surplus Volume.
- At the end of the pre-open a message is sent with zeros for all price and quantity fields (after all the auction trades).

Field Name	Offset	Length	Type	Notes
Message Type	0	1	Alpha	Set to “Z”
Timestamp	1	4	Numeric	Nanoseconds portion of the timestamp, accurate to 1,000 nanoseconds.
Trade Date	5	2	Numeric	Trade Date - Represented as number of days since 1 January 1970.
Tradeable Instrument Id	7	4	Numeric	Identifier of the Order Book.
Equilibrium Price	11	8	Price	Price at which matching will occur.
Matched Quantity	19	8	Numeric	The indicative auction volume.
Bid Quantity	27	8	Numeric	Total Bid Quantity available for execution.
Ask Quantity	35	8	Numeric	Total Ask Quantity available for execution.
		43		

4.7.2. *Open, High, Low, Last Trade Adjustment (t)*

Market trade price adjustments are sent resulting from modifications of Open, High, Low, Last and volume statistics due to market operations (for example, a trade cancellation), prior to the instrument moving to the next session within the trade date, or at the close of instrument. This message is also issued during a snapshot download to enable the requestor to generate an up-to-date market image without having to regenerate this from previous trades and market trade adjustment messages.

ⓘ **Note**: The OHLL (t) message returned in a Glance request after the roll of a business date but before the next maintenance state message will have the new trade date, but the statistics from the previous date.

Field Name	Offset	Length	Type	Notes
Message Type	0	1	Alpha	Set to “t”
Timestamp	1	4	Numeric	Nanoseconds portion of the timestamp, accurate to 1,000 nanoseconds.
Trade Date	5	2	Numeric	Trade Date - Represented as number of days since 1 January 1970.
Tradeable Instrument Id	7	4	Numeric	Identifier of the Order Book.
Opening Trade	11	8	Price	Opening Traded Price.
Highest Trade	19	8	Price	Highest Traded Price.
Lowest Trade	27	8	Price	Lowest Traded Price.
Last Trade	35	8	Price	Last Traded Price.
Last Volume	43	4	Numeric	Last Traded Volume.
Total Traded Volume	47	8	Numeric	Total Traded Volume.
		55		

4.7.3. Market Settlement (Y)

This message disseminates the final settlement price of an order book (future or option contracts) for the trade date it pertains to.

In certain circumstances the final settlement price may be sent out more than once. Customers should therefore support the handling of more than one Market Settlement (Y) message with Settlement Type = F for the same trade date in their software.

Field Name	Offset	Length	Type	Notes
Message Type	0	1	Alpha	Set to “Y”
Timestamp	1	4	Numeric	Nanoseconds portion of the timestamp, accurate to 1,000 nanoseconds.
Trade Date	5	2	Numeric	Trade Date - Represented as number of days since 1 January 1970.
Tradeable Instrument Id	7	4	Numeric	Identifier of the Order Book.
Settlement Price	11	8	Price	Settlement Price as indicated by Settlement Type.
Volatility	19	8	Numeric	Volatility for an option contract. Divide by 1,000,000 to determine the floating point value.
Delta	27	4	Delta	Delta of the option contract. If a non-option contract value is zero, range of delta is 1.000000 to -1.000000. Divide by 1,000,000 to determine the floating point value.
Settlement Type	31	1	Alpha	Settlement Type qualifies the Settlement Price as one of: “F” – Settlement Price is the Final settlement price “A” – Settlement Price is an adjustment for the Prior Day.
		32		

4.7.4. Text Message (x)

This message disseminates free text information describing ad-hoc market events.

Field Name	Offset	Length	Type	Notes
Message Type	0	1	Alpha	Set to “x”
Timestamp	1	4	Numeric	Nanoseconds portion of the timestamp, accurate to 1,000 nanoseconds.

Field Name	Offset	Length	Type	Notes
Trade Date	5	2	Numeric	Trade Date - Represented as number of days since 1 January 1970.
Source Id	7	6	Alpha	Indicates source of message.
Text message	13	100	Alpha	Message being communicated.
		113		

4.7.5. Request for Quote (q)

This message is disseminated for anonymous Request for Quotes (RFQs) from the market.

Field Name	Offset	Length	Type	Notes
Message Type	0	1	Alpha	Set to “q”
Timestamp	1	4	Numeric	Nanoseconds portion of the timestamp, accurate to 1,000 nanoseconds.
Trade Date	5	2	Numeric	Trade Date - Represented as number of days since 1 January 1970.
Tradeable Instrument Id	7	4	Numeric	Identifier of the Order Book.
Side	11	1	Alpha	Identifies side of the RFQ Valid options T - Two sided quote B - Bid quote S - Ask quote X - Crossing
Quantity	12	4	Numeric	Requested quote volume. Zero if no volume is specified.
		16		

4.7.6. Anomalous Order Threshold Publish (W)

This message disseminates price band information to the market advising the setting, zeroing or adjusting of the Extreme Trade Range price (ETR), the Anomalous Order Threshold price (AOT) and the associated ranges for a contract that requires Market Integrity Rules (MIR) enforcement. If the ETR Price is zero, then order price validation performed by the matching engine is no longer enforced.

Field Name	Offset	Length	Type	Notes
Message Type	0	1	Alpha	Set to "W"
Timestamp	1	4	Numeric	Nanoseconds portion of the timestamp, accurate to 1,000 nanoseconds.
Trade Date	5	2	Numeric	Trade Date - Represented as number of days since 1 January 1970.
Tradeable Instrument Id	7	4	Numeric	Identifier of the Order Book.
AOT Price	11	8	Price	Anomalous Order Threshold price (AOT).
AOT Upper Price	19	8	Price	Upper range of the AOT.
AOT Lower Price	27	8	Price	Lower range of the AOT.
ETR Price	35	8	Price	Extreme Trade Range price (ETR), if zero, then MIR is not enforced.
ETR Upper Price	43	8	Price	Upper range of the ETR.
ETR Lower Price	51	8	Price	Lower range of the ETR.
		59		

4.7.7. Volume and Open Interest (V)

This message disseminates Volume and Open Interest (VOI) information for each trade date. Due to the timing of the reporting of this information, it can be reported for the prior trade date where that trade date has been closed. The VOI Trade Date field will indicate the trade date the information is applicable to.

Field Name	Offset	Length	Type	Notes
Message Type	0	1	Alpha	Set to “V”
Timestamp	1	4	Numeric	Nanoseconds portion of the timestamp, accurate to 1,000 nanoseconds.
Trade Date	5	2	Numeric	Trade Date - Represented as number of days since 1 January 1970.
Tradeable Instrument Id	7	4	Numeric	Identifier of the Order Book.
Cumulative Volume	11	8	Numeric	Volume traded for the specified VOI Trade Date.
Open Interest	19	8	Numeric	Number of open contracts at EOD for the specified VOI Trade Date.
VOI Trade Date	27	2	Numeric	Trade Date the VOI information is applicable to.
		29		

4.7.8. Snapshot Complete (G)

Subscribers using the Glance service to recover a market image will receive this message signalling the snapshot download is complete and the sequence number indicates to the subscriber the next Packet Header to process to keep continuity with the market.

Subscribers using the Change Password service will receive this message signalling the password process has completed.

Field Name	Offset	Length	Type	Notes
Message Type	0	1	Alpha	Set to "G"
Sequence Number	1	8	Numeric	Glance service • Sequence number of the next expected message to receive to synchronise with the market. Password service • Not used. Set to zero.
		9		

5. Administration

Administration consists of subscribers obtaining start-up and multicast recovery of market information for a product grouping. There will be one or more dedicated servers available for subscribers to obtain market data for start-up and multicast recovery for each product grouping. Each product grouping will have a specific set of pre-assigned addresses to service snapshot and multicast recovery functions.

5.1. Multicast Session and Sequence

For each product grouping every packet is multicast with a session, sequence number and a count of data messages. The session is an alphanumeric identifier indicating the market data is from a continuous multicast session when the back-end trading server began. If the session changes, the back-end trading server has been re-started as a result of a scheduled start-up or recovery from a failure.

If a subscriber receives a packet with a session that differs from the session of the last packet received, it is expected the subscriber will either purge the current market image or create a new market image identified by the new session.

For each new session, the sequence number will always start at 1. The sequence number is sequentially assigned and incremented for each packet multicast with a data message. If the packet is a heartbeat, the sequence number will not be incremented but will be the expected sequence number of the next packet. The sequence number on each packet is the sequence number for the first data message and each data message within the packet has an implied sequence number one greater than the previous data message. The sequence number of the first data message of the next packet can be determined by adding the sequence number and message count of the prior packet.

As multicast is an unreliable protocol the subscriber is responsible for the detection of missing packets and the recovery. The detection of missing multicasts is simply inspecting the sequence number of the Packet Header and checking if it is greater than the expected sequence number based on the last Packet Header received. Subscribers should be prepared to handle duplicate message packets, particularly before or after the receipt of recovered message packets has been made.

5.1.1. Message Packet Recovery from Blink

The “Blink” server is a recovery mechanism for subscribers to obtain missing multicasts during the course of the multicast session for a product grouping. The “Blink” server will keep a store of message packets specifically for multicast loss.

To recover missing message packets where a subscriber has detected a loss, a request can be made to the “Blink” server. To obtain the missing message packets a subscriber unicasts to a pre-assigned server address and makes a Request Packet indicating the sequence number of the first message and the number of data messages to re-transmit.

The server will respond with the requested data messages with the originally broadcast Time Message of the multicast (i.e. with the same timestamp). A single packet will be unicast back to the requestor with a packet header containing as many data messages that can fit into a single UDP Ethernet frame.

Where a subscriber requests more data messages than can fit within one Ethernet frame, the subscriber must make another request starting from the next sequence number calculated based on the last received packet.

For instance, if a subscriber makes a request for sequence number 150 with a count of 40 messages (i.e. messages 150-189) the process is as follows:

- Data messages 150 to 175 fit into one Ethernet frame, Blink unicasts this packet to the subscriber.
- The subscriber makes another request for sequence number 176 with a count of 14 messages.
- Blink creates a single packet of data messages 176 to 189 and unicasts this packet to the subscriber.

If a request for missing multicast is made, where the request is outside the range of available messages, is an invalid request (incorrect session or corrupt), or the Blink server does not currently have the requested data, the subscriber’s request will be ignored and no response will be issued. The request should be resent after a 50ms delay.

5.1.2. Request Packet

To initiate a multicast recovery request the subscriber unicasts this message to a pre-assigned “Blink” server. The port used is specific to a product grouping of the multicast of the missing messages. The number of messages returned by the “Blink” server will be based on the number of messages fitting into a single UDP Ethernet frame, or approximately 1500 bytes. The subscriber will need to make additional requests if the initial response is not filled.

Field Name	Offset	Length	Type	Notes
Session	0	10	Alpha	Identity of the multicast session request relates to. See *3.1.1 Packet Header*.

Field Name	Offset	Length	Type	Notes
Sequence Number	10	8	Numeric	Sequence number of the first message to re-transmit.
Requested Message Count	18	2	Integer	Number of messages to be resent.
		20		

5.1.3. Snapshot Recovery from Glance

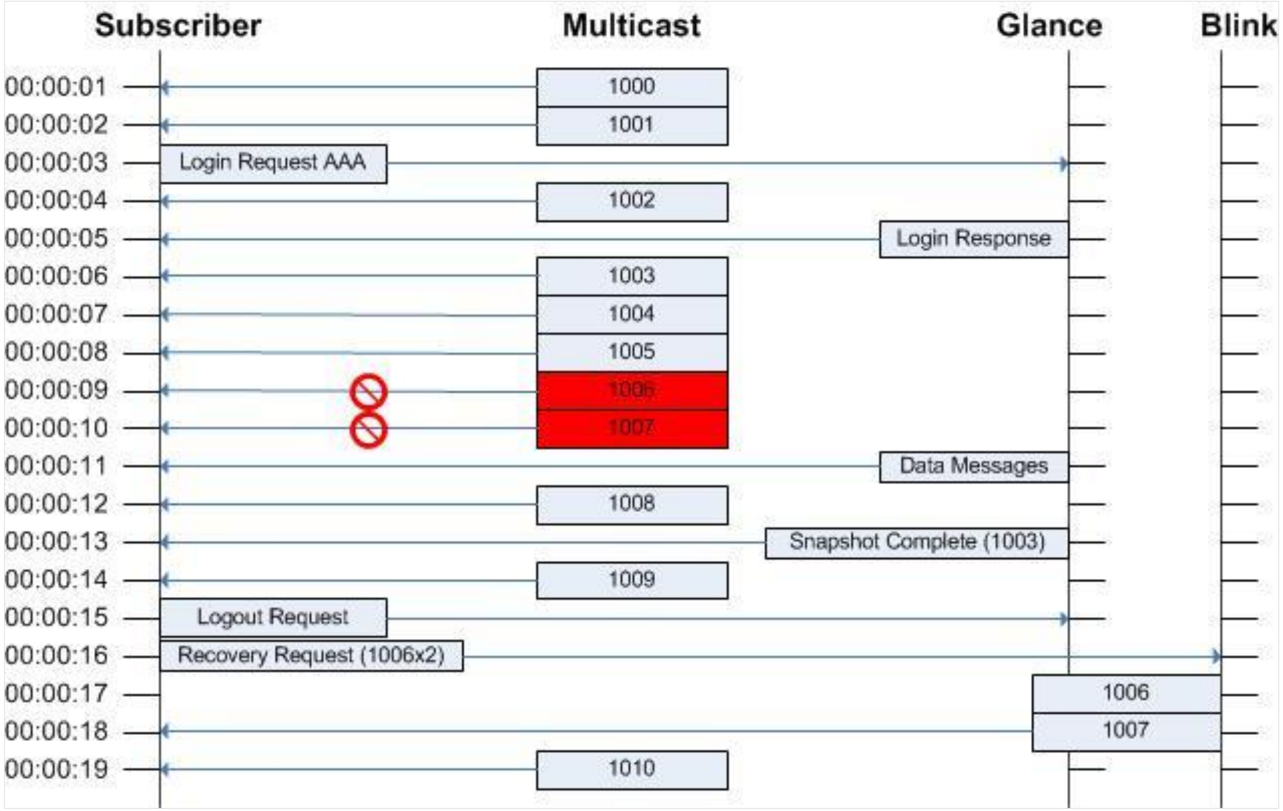
“Glance” is a point-to-point connection providing a direct data feed delivering a snapshot of the current state of the market for a specific product grouping for subscriber applications. Glance uses the same message format as the multicast and delivers data messages using the SoupBinTCP protocol. Although Glance uses the same message format as the multicast stream, the implied sequence numbers of packets in SoupBinTCP do not correlate to the sequence numbers in MoldUDP64 multicast.

The Glance server will deliver messages to enable a subscriber to re-create the current market data picture. The snapshot recovery will not contain data messages of trades or of orders no longer in the book.

While receiving a snapshot, the subscriber must continue to buffer any packets received. When the snapshot is complete, the Snapshot Complete message is transmitted and contains the sequence number of the next message to process in the multicast stream.

If the sequence number in the Snapshot Complete message is lower than the lowest sequence number buffered, Blink can be used to fill the gap.

Below is an example of making a snapshot request for start-up, where the subscriber detects missing sequences during the recovery and recovers the missing message packets.


Subscriber
Multicast
Glance
Blink
00:00:01
00:00:02
00:00:03
00:00:04
00:00:05
00:00:06
00:00:07
00:00:08
00:00:09
00:00:10
00:00:11
00:00:12
00:00:13
00:00:14
00:00:15
00:00:16
00:00:17
00:00:18
00:00:19
1000
1001
Login Request AAA
1002
Login Response
1003
1004
1005
1006
1007
Data Messages
1008
Snapshot Complete (1003)
1009
Logout Request
Recovery Request (1006x2)
1006
1007
1010

5.1.4. Administrative Packets

List of SoupBinTCP administrative packets:

	Message Name	Type	Notes
Server Initiated Packets	Login Accepted Packet	“A”	Server response to valid login.
	Login Reject Packet	“J”	Server response to an invalid login or a forced logoff. This message has field extensions to SoupBinTCP.
	Sequenced Data Packet	“S”	Data messages sent to a subscriber.
	Server Heartbeat Packets	“H”	Heartbeats to indicate server is alive.
	Password Change Response Packet	“Y”	Server response to a password change request. This is an extension to SoupBinTCP for password administration.
	End of Session Packet	“Z”	End of session packet to denote the current Glance session is finished.
Subscriber Initiated Packets	Login Request Packet	“L”	Request to login sent by a subscriber. This message has field extensions to SoupBinTCP.
	Un-sequenced Data Packet	“U”	Not supported.
	Subscriber Heartbeat Packets	“R”	Heartbeats to indicate subscribe is alive.
	Logout Request Packet	“O”	Request to logout sent by a subscriber.
	Password Change Request Packet	“W”	Subscriber request to change password. This is an extension to SoupBinTCP for password administration.

All data elements used in SoupBinTCP administrative packets use the big-endian format for binary numeric data, and left justified, spaced padded for alphanumeric data.

Type	Length	Notes
Alpha	Variable	These fields are composed of non-control ISO 8859 (Latin-1) encoded bytes. They are left justified and padded on the right with spaces.
Integer	2	Big-Endian encoded 16 bit signed integer.
Integer	4	Big-Endian encoded 32 bit signed integer.
Integer	8	Big-Endian encoded 64 bit signed integer.

5.2. Server Initiated Packets

5.2.1. Login Accepted Packet

The Glance server will always respond to a successful Login Request Packet with this message. After this message has been received, the subscriber should expect a download of Sequenced Data Packet contains data messages to create a snapshot of the market image.

When a valid Login Request packet is accepted, the server will respond with a Login Accepted packet and commence a Glance download. Any subsequent request packets sent by the client will be ignored.

Field Name	Offset	Length	Type	Notes
Packet Length	0	2	Integer	Number of bytes after this field until the next packet.
Packet Type	2	1	Alpha	Set to “A”.
Session id	3	10	Alpha	The session identifier that Glance is providing information on.
Sequence Number	13	8	Integer	The sequence number of the next sequenced message to be sent (Glance will always set this value to 1).
		21		

5.2.2. *Login Reject Packet*

The Login Reject packet is sent by the server and informs the client of why the login was rejected or terminated. After a configurable number of consecutive login failures the account will be locked and the server will stop responding to requests. After a configurable time period the server will close the TCP/IP connection.

Field Name	Offset	Length	Type	Notes
Packet Length	0	2	Integer	Number of bytes after this field until the next packet.
Packet Type	2	1	Alpha	Set to “J”.
Reject Reason Code	3	4	Integer	Code indicating why the login was rejected. Valid values: -1 = The login is rejected due to invalid password or invalid user id. -2 = User account is locked due to too many erroneous login attempts. -3 = The password has expired. -4 = User does not have access to login service for this application. -5 = Client and TAX server versions are not compatible. -6 = Initial login, password must be changed. -7 = Account disabled by operational staff. -8 = Sequence number too high, indicates the client is attempting to recover an unavailable session. -9 = Other. Error code will be populated with additional information
Error Code	7	4	Integer	Additional server error code provided when the Reject Reason Code = -9
		11		

5.2.3. *Sequenced Data Packet*

The Sequenced Data Packets act as an envelope to carry the actual sequenced data messages that are transferred from the server to the client. Each Sequenced Data Packet carries one message from the higher-level protocol. The sequence number of each message is implied; the initial sequence number of the first Sequenced Data Packet for a given TCP/IP connection is specified in the Login Accepted Packet and the sequence number increments by 1 for each Sequenced Data Packet transmitted.

Since logical packets are carried via TCP/IP sockets, the only way logical packets can be lost is in the event of a TCP/IP socket connection failure. In this case, the client can reconnect to the server and request the next expected sequence number and pick up where it left off.

Field Name	Offset	Length	Type	Notes
Packet Length	0	2	Integer	Number of bytes after this field until the next packet.
Packet Type	2	1	Alpha	Set to “S”
Message	3	Variable	Any	Defined by higher-level protocol.
		*		

Glance will issue multiple messages using the following format:

Packet Length 2 bytes	“S”	Message Type	Message Data *

5.2.4. *Server Heartbeat Packets*

After a successful login, the server will send a Server Heartbeat Packet anytime more than 1 second passes where no data has been sent to the client. The client can then assume that the link is lost if it does not receive anything for an extended period of time.

Field Name	Offset	Length	Type	Notes
Packet Length	0	2	Integer	Number of bytes after this field until the next packet.
Packet Type	2	1	Alpha	Set to “H”

Field Name	Offset	Length	Type	Notes
		3		

5.2.5. *Password Change Response Packet*

The Password Change Response Packet is sent by the server in response to a Password Change Request Packet sent by the client. Once the password Change Response Packet has been sent the server will issue a Snapshot Complete (G) packet and disconnect the TCP/IP connection.

Field Name	Offset	Length	Type	Notes
Packet Length	0	2	Integer	Number of bytes after this field until the next packet.
Packet Type	2	1	Alpha	Set to "Y"
Member	3	64	Alpha	Member Id.
Username	67	64	Alpha	User Id.
Status	131	4	Integer	Status of password change request. Valid values: 3001 = Password successfully changed 52006 = Invalid user id or password 52007 = Invalid password, new password cannot be same as old 52009 = Invalid password format -1 = Request rejected due to system issue.
		135		

5.2.6. *End of Session Data Packet*

The Glance service will send an End of Session Packet to denote that the current Glance session is finished. The TCP/IP connection will be closed shortly after this packet is sent. Note the End of Session Packet is not transmitted over MoldUDP64 multicast.

Field Name	Offset	Length	Type	Notes
Packet Length	0	2	Integer	Number of bytes after this field until the next packet.
Packet Type	2	1	Alpha	Set to “Z”
		3		

5.3. Subscriber Initiated Packets

5.3.1. Login Request Packet

Each subscriber must connect and log into a Glance server using a valid username and password.

Field Name	Offset	Length	Type	Notes
Packet Length	0	2	Integer	Number of bytes after this field until the next packet.
Packet Type	2	1	Alpha	Set to “L”
Member	3	64	Alpha	Mandatory member id.
Username	67	64	Alpha	The user must belong to the member.
Password	131	64	Alpha	Password for the user.
Ticket	195	8	Integer	Not used by Glance and will be ignored. Set to zero.
Requested Sequence Number	203	8	Integer	Specifies the next sequence number the client wants to receive upon connection, or 0 to start receiving the most recently generated message. This field is ignored by the Glance service as all messages are downloaded.
Version	211	12	Alpha	API version.

Field Name	Offset	Length	Type	Notes
				Note: The current API version is 0.1.0. This is subject to change with new releases of the software.
		223		

5.3.2. *Subscriber Heartbeat Packets*

The subscriber should send a heartbeat packet where more than 1 second passes when no data has been sent to the server. The server will send a Logout Response and disconnect if nothing is received for an extended period of time.

Field Name	Offset	Length	Type	Notes
Packet Length	0	2	Integer	Number of bytes after this field until the next packet.
Packet Type	2	1	Alpha	Set to “R”
		3		

5.3.3. *Logout Request*

The client will send a Logout message when it no longer wishes to use the session. Any actions that are to be taken on disconnect will be triggered by the logout.

Field Name	Offset	Length	Type	Notes
Packet Length	0	2	Integer	Number of bytes after this field until the next packet.
Packet Type	2	1	Alpha	Set to “O”
		3		

5.3.4. Password Change Request Packet

The Password Change Request Packet is sent by the client to the server to change the password. It can be sent any time after the user has established a TCP/IP connection with the server. A Login request should not be sent prior to sending a Password Change as this will commence a Glance download. Once a Glance download has been initiated any subsequent request packets sent are ignored.

Field Name	Offset	Length	Type	Notes
Packet Length	0	2	Integer	Number of bytes after this field until the next packet.
Packet Type	2	1	Alpha	Set to “W”
Member	3	64	Alpha	Member Id.
Username	67	64	Alpha	User Id. The user must belong to the member.
Old Password	131	64	Alpha	Current password.
New Password	195	64	Alpha	New password.
		259		

The new password must meet at least three of the following criteria:

- Contain a English upper case character (A-Z)
- Contain a English lower case character (a-z)
- Contain a Hindu Arabic numeral (0-9)
- Contain one or more of the following non-alphanumeric special characters: !@#$%^&;*()_+|~-=\`{}[]:";'<>?,./)
- Contain any character that is categorized as an alphabetic character but is not uppercase or lowercase, this includes characters from Asian languages.

Passwords must be a minimum of 8 characters in length.

Passwords are valid for 90 Days, and when reset must be different to the previous 12 passwords used.

6. Appendix

6.1. MoldUDP64 and SoupBinTCP

Communication between the ASX Trading Multicast, Blink, Glance servers and the subscriber is based on the following protocols:

- UDP multicast MoldUDP64 protocol is used by the ASX Trading Multicast server,
- UDP unicast MoldUDP64 protocol is used by the Blink server, and
- TCP/IP SoupBinTCP protocol is used by the Glance server.

For further information, refer to MoldUDP64 Specification (http://www.asx.com.au/documents/products/Moldudp64V1.pdf), and SoupBinTCP Specification (http://www.asx.com.au/documents/products/SoupBinTCP30.pdf).

This document provides general information only and may be subject to change at any time without notice. ASX Limited (ABN 98 008 624 691) and its related bodies corporate ("ASX") makes no representation or warranty with respect to the accuracy, reliability or completeness of this information. To the extent permitted by law, ASX and its employees, officers and contractors shall not be liable for any loss or damage arising in any way, including by way of negligence, from or in connection with any information provided or omitted, or from anyone acting or refraining to act in reliance on this information. The information in this document is not a substitute for any relevant operating rules, and in the event of any inconsistency between this document and the operating rules, the operating rules prevail to the extent of the inconsistency.

ASX Trade Marks

The trademarks listed below are trademarks of ASX. Where a mark is indicated as registered it is registered in Australia and may also be registered in other countries. Nothing contained in this document should be construed as being any licence or right to use of any trade mark contained within the document.

ASX®