ASX 24
FIX Market Data Specification

Version 2.02 | July 2023

Table of Contents

1. FIX Market Data Introduction

ASX provides the facility to receive public market data using the FIX protocol. This document provides guidance on using FIX protocol messages to:

- Establish and maintain a FIX session
- Subscribe to and receive details of orders and trades
- Obtain information about instruments available for trading
- Obtain information about the status of the market.

These rules of engagement are based on the FIX 5.0 SP2 Specification, Extension Pack 196 (FIX 5.0 SP2 EP 196) and best practice guidelines as published by the FIX Trading Community. Unless specifically stated, field numbers, names, and data types are as published by the FIX specification. A full explanation of the FIX protocol is out of scope of this document, and therefore customers should refer to FIX Trading Community for a full understanding of the protocol, prior to using this guide.

1.1. Document Information

This document describes:

- Standard Header and Standard Trailer definitions
- FIX session layer—how FIX sessions are established and maintained
- Market data messages—how the marketplace publishes market data and security status information
- Reference data—how reference data is handled
- General messages—messages that are not part of any of the above.

Please refer to the ASX FIX Market Data Specification Updates document to see changes between the V1.02 and V1.03 specifications.

1.1.1. Terms and Acronyms

Term / Acronym	Description
FIX	Financial Information eXchange Protocol
FPL	FIX Protocol Limited
ISIN	International Securities Identification Number. Unique identifier issued to identify each financial instrument.
UTC	Coordinated Universal Time is a high-precision, atomic time standard often referred to as GMT.
CFI	Classification of Financial Instruments.

1.1.2. Usage of String Datatypes

String datatype field definitions that do not have specified values or lengths in FIX have been updated to display field lengths, in square brackets, supported by ASX. For example, String [19] indicates that 19 characters is the maximum number of characters that ASX will send or process on incoming messages.

1.1.3. Handling of unsupported messages and tags

Any message that is not listed in this specification will be rejected.

If a message listed in this specification is received with a tag that is not in the specification, the message will be rejected.

1.1.4. Usage of UTCTimestamp

ASX 24 supports the UTCTimestamp second (YYYYMMDD-HH:MM:SS) and milliseconds (YYYYMMDD-HH:MM:SS.sss) formats. Times with higher precision or that do not meet those formats will be rejected.

1.1.5. Definition of required column values

For messages transmitted to ASX

Value	Definition
Y	Defines as required in FIX 5.0 SP2 EP 196.
C	Conditionally required by either FIX protocol or by ASX to implement functionality.

Blank	Tag may be omitted.

For messages transmitted from ASX

Value	Definition
Y	Defined as required in FIX 5.0 SP2 EP 196.
Blank	Always provided by ASX unless text specifies otherwise.

1.2. 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	August 2016	Added documentation errata and changes per delta document.
V1.02	September 2016	Corrective Release updates – details are included in the Updates document.
V1.03	December 2016	Added documentation errata and changes per delta document.
V1.04	April 2017	Miscellaneous clarifications to the following messages: • DerivativeSecurityList (AA)
V1.05	August 2017	• Clarify Logon (A) to request waiting for Logon ack before sending messages. • Detail how unsupported messages and tags are handled
V2.00	August 2018	• Clarify what UTCTimestamp format ASX 24 accepts • Clarify that MatchType (574) and TradeCondition (277) in MarketDataSnapshotFullRefresh (W) and MarketDataIncrementalRefresh (X) messages in response to an aggregated request may not represent the value of all underlying trades • Updated to the list of valid MDEntryType (269) values in MarketDataSnapshotFullRefresh (W) message
V2.01	October 2018	• Updated selected string fields to display length supported by ASX 24

V2.02	July 2023	• Updated naming conventions

2. Common Message Structures

2.1. Standard Header

All FIX messages described in this document contain a standard header, which is defined below.

Tag	Name	Data Type	Reqd	Comment
8	BeginString	String	Y	Identifies beginning of new message and protocol version. Always first field in message. Valid values: FIXT.1.1
9	BodyLength	Length	Y	Message length, in bytes, forward to the CheckSum field. Always second field in message.
35	MsgType	String	Y	Defines message type. Always third field in message. See individual messages for value to be used.
49	SenderCompID	String [64]	Y	Identifies the sender of the message.
56	TargetCompID	String [64]	Y	Identifies the receiver of the message.
34	MsgSeqNum	SeqNum	Y	Message sequence number.

Tag	Name	Data Type	Reqd	Comment
43	PossDupFlag	Boolean	C	Required for retransmitted messages. Indicates possible retransmission of message with this sequence number. Valid values: N = Original Transmission Y = Possible duplicate.
52	SendingTime	UTCTimestamp	Y	Time of transmission in UTC.
122	OrigSendingTime	UTCTimestamp	C	Required for messages sent as a result of a ResendRequest. Original time of message transmission in UTC.
369	LastMsgSeqNumProcessed	SeqNum		The last MsgSeqNum (34) value received by the FIX engine and processed by a downstream application, such as trading engine or order routing system. Can be specified on every message sent. Useful for detecting a backlog with a counterparty.

2.2. Standard Trailer

All FIX messages in this document contain a standard trailer, which is defined below.

Tag	Name	Data Type	Reqd	Comment
10	CheckSum	String	Y	Simple checksum (see Volume 2: "Checksum Calculation" for description in FIX Trading Community FIX Session Layer). ALWAYS LAST FIELD IN MESSAGE; i.e. serves, with the trailing <SOH>, as the end-of-message delimiter. Always defined as three characters. Always unencrypted.

3. FIX Session Layer

3.1. Connecting to the FIX Server Process

Each FIX client must maintain the address and port for the primary and the standby ASX FIX server. If a server cannot be reached, we recommend performing the following steps:

- Attempt to connect to the other server (standby, if primary attempted and primary, if standby attempted)
- Wait five seconds, and then try to connect to the server again.

3.2. Lost Connection

3.2.1. Recovery

FIX Market Data does not support standard FIX session recovery. The recovery mechanism is to renew your subscriptions to get a snapshot of the current market and to use Replay Trades functionality described below.

3.2.2. Replay Trades

All trades disseminated by the market are assigned a trade sequence number (TradeSeqNo) from a trade sequence number series (TradeSeqNoSeries). The trade sequence number starts with 0 (zero) after a matching engine (ME) startup, and increases until the next ME startup. An ME startup occurs every weekend. In the event of an ME startup outside of the regular weekend one, ASX will advise on TradeSeqNo handling.

Trade sequence numbers are unique and when combined with TradeSeqNoSeries, specifically identify a trade within an ME session. Trade sequence numbers increase sequentially within each TradeSeqNoSeries, however not all numbers are used for trades and there may be gaps.

When setting up a subscription for trades, a client can choose to only receive trades with a TradeSeqNo higher than a certain value. This procedure can be used when reconnecting to the FIX Market Data server after a FIX session has been down during the day.

To determine the suitable starting TradeSeqNo when setting up a subscription, the client must examine TradeSeqNo and TradeSeqNoSeries from previously received trades from the same ME session. For each TradeSeqNoSeries, the client should store information about the highest TradeSeqNo.

Each subscription request should specify the last received TradeSeqNo for each TradeSeqNoSeries.

3.3. Logon (A)

The logon message is the first message sent by a user and is used to authenticate the FIX session with the exchange. On successful authentication, the response is a Logon (A) message. The user should wait for the confirming Logon (A) message before sending other messages. Messages sent prior to this confirmation may not be processed.

In the event that logon fails, a Logout (5) message will be sent and the TCP/IP session will be terminated under most circumstances. Circumstances when a Logout (5) message is not sent, include an invalid SenderCompID (49) or TargetCompID (56). See the FIX Trading Community FIX Session Layer for circumstances when a Logout (5) message should not be issued.

If a logon attempt fails, the client should reattempt logon no more than 3 times before taking remedial action requested in the Logout message. Remedial action includes changing the password sent on the Logon (A) message, correcting sequence numbers, or contacting ASX regarding account administration.

The password can be changed by specifying the new password in the NewPassword (925) tag.

Market Data connections do not support FIX Session Layer recovery. Subsequent logins must set MsgSeqNum to 1, and set ResetSeqNumFlag to True (141=Y) to reset the connection. On disconnection, all subscriptions are cancelled and need to be re-established on reconnection.

Tag	Name	Data Type	Reqd	Comment
	StandardHeader		Y	MsgType = A
98	EncryptMethod	int	Y	Method of encryption. ASX does not use FIX encryption however this field is required by FIX 5.0 SP2. Please set to zero, to specify no encryption. Other valid values are accepted, but will not have any effect. Valid values: 0 = None/Other.

Tag	Name	Data Type	Reqd	Comment
108	HeartBtInt	int	Y	Heartbeat interval in seconds. The same value is used by both sides. Specified by the logon initiator and reflected by the acceptor (ASX). ASX supports values between 5 and 60, recommending 30. ASX does not support a value of zero. Any other values may be accepted but are not supported.
141	ResetSeqNumFlag	Boolean	C	Required to establish logon. Indicates both sides of a FIX session should reset sequence numbers. Valid value: Y = Yes, reset sequence numbers. Note MsgSeqNum (34) should be set to 1.
789	NextExpectedMsgSeqNum	SeqNum		Next expected MsgSeqNum value to be received. Required when connecting to the secondary (standby) FIX Drop Copy server. Please see the FIX Trading Community FIX Session Protocol description on the use of this tag for recovery.
553	Username	String [64]	C	FIX username.
554	Password	String [128]	C	Password for username.

Tag	Name	Data Type	Reqd	Comment
925	NewPassword	String [128]		Specifies a new password for the FIX Logon. The new password is used for subsequent logons. 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. The account will lock after 6 failed attempts.

Tag	Name	Data Type	Reqd	Comment
1409	SessionStatus	int		Status of the FIX session. Sent by ASX. Ignored if input by client. Valid values: 0 = Session active 1 = Session password changed.
1137	DefaultApplVerID	String	Y	Specifies the service pack release being applied, to the message at the session level. Valid values: 9 = FIX50SP2
58	Text	String [see comment]		Free format text string. Up to 128 characters on incoming message. Outgoing messages may exceed 128 characters.
	StandardTrailer		Y	

3.4. Heartbeat (0)

Heartbeat messages are sent by counterparties to indicate that a connection is still active and as a response to TestRequest (1) messages. The behaviour is as described in the FIX Trading Community FIX Session Layer.

Tag	Name	Data Type	Reqd	Comment
	StandardHeader		Y	MsgType = 0
112	TestReqID	String [64]		Identifier included in Test Request message to be returned in resulting Heartbeat. Required when the heartbeat is the result of a Test Request message.
	StandardTrailer		Y	

3.5. TestRequest (1)

To verify a connection is active, a TestRequest (1) message is sent to the counterparty. The recipient of the TestRequest responds with a Heartbeat (0) message. Failure to respond to a TestRequest message may trigger a disconnection by the sender.

Behaviour is as described in the FIX Trading Community FIX Session Layer.

Tag	Name	Data Type	Reqd	Comment
	StandardHeader		Y	MsgType = 1
112	TestReqID	String [64]	Y	Identifier of this Test Request. To be returned in the Heartbeat generated upon receipt of the Test Request.
	StandardTrailer		Y	

3.6. ResendRequest (2)

The ResendRequest (2) message is sent to request the retransmission of messages. Note that either party may send a ResendRequest.

On Market Data FIX sessions, ResendRequest (2) messages sent to ASX do not return trading data. SequenceReset (4) messages are used to gap fill in order to conform to the protocol. If ASX detects a sequence gap in incoming messages from the client, it will send a ResendRequest message to the client and process the messages resent. Until the requested messages have been received by ASX, messages sent by the client will be rejected with a BusinessMessageReject (j) message.

The behaviour is as described in the FIX Trading Community FIX Session Layer.

Tag	Name	Data Type	Reqd	Comment
	StandardHeader		Y	MsgType = 2
7	BeginSeqNo	SeqNum	Y	Message sequence number of first message in range to be resent.
16	EndSeqNo	SeqNum	Y	Message sequence number of last message in range to be resent. If all messages subsequent to BeginSeqNo are required, set EndSeqNo=0.
	StandardTrailer		Y	

3.7. Reject (3)

The reject message is issued when a message is received but cannot be properly processed due to a session-level rule violation. As an example, a reject can be issued on receipt of a message with invalid basic data which successfully passes decryption, check sum, and body length checks.

Behaviour is as described in the FIX Trading Community FIX Session Layer.

Tag	Name	Data Type	Reqd	Comment
	StandardHeader		Y	MsgType = 3
45	RefSeqNum	SeqNum	Y	MsgSeqNum of rejected message.
371	RefTagID	int		The tag number of the FIX field being referenced.
372	RefMsgType	String		The MsgType (35) of the FIX message being referenced.
373	SessionRejectReason	int		Code to identify reason for a session-level Reject message. Valid values: 0 = Invalid Tag Number 1 = Required Tag Missing 4 = Tag specified without a value 5 = Value is incorrect (out of range) for this tag 6 = Incorrect data format for value 9 = CompID problem 10 = SendingTime accuracy problem 11 = Invalid MsgType 13 = Tag appears more than once 14 = Tag specified out of required order.

Tag	Name	Data Type	Reqd	Comment
58	Text	String [see comment]		Where possible, message to explain reason for rejection. Up to 128 characters on incoming message. Outgoing messages may exceed 128 characters.
	StandardTrailer		Y	

3.8. SequenceReset (4)

The SequenceReset (4) message is used to inform the counterparty of a new higher sequence number. This is required in order to skip one or more messages when responding to a ResendRequest (2), or to set a new sequence number after an unrecoverable error.

The behaviour is as described in the FIX Trading Community FIX Session Layer.

Tag	Name	Data Type	Reqd	Comment
	StandardHeader		Y	MsgType = 4
123	GapFillFlag	Boolean		Indicates that the Sequence Reset message is replacing administrative or application messages, which will not be resent. Valid values: N = Sequence Reset, Ignore Msg Seq Num Y = Gap Fill Message, Msg Seq Num field valid.
36	NewSeqNo	SeqNum	Y	New sequence number.
	StandardTrailer		Y	

3.9. Logout (5)

The logout message is used to initiate or confirm the termination of a FIX session.

Logout is normally initiated by the client. The ASX may also initiate a logout, for example, prior to system shutdown. A Logout message is also used to respond to failed Login (A) requests. On completion of the logout procedure, ASX will close the TCP/IP connection. The logout process followed is as described in the FIX Trading Community FIX Session Layer.

Tag	Name	Data Type	Reqd	Comment
	StandardHeader		Y	MsgType = 5
58	Text	String [see comment]		Free format text string. Up to 128 characters on incoming message. Outgoing messages may exceed 128 characters.
1409	SessionStatus	int		Status of the FIX session. Not sent for scheduled server initiated log outs, for example, when the server is shutting down. Valid values: 3 = New session password does not comply with policy 4 = Session logout complete 5 = Invalid username or password 6 = Account locked 7 = Logons are not allowed at this time 8 = Password expired 9 = Received MsgSeqNum (34) is too low 10 = Received NextExpectedMsgSeqNum (789) is too high

Tag	Name	Data Type	Reqd	Comment
				101 = Requested MsgSeqNum unknown (a request has been made for a sequence number that has not been sent by the server). 102 = MsgSeqNum (34) must be equal to 1 and ResetSeqNumFlag (141) must be set to true for market data connections. 104 = HeartBtInt (108) must be greater than 1 105 = NextExpectedMsgSeqNum (789) must be set after FIX server failover 106 = Logon (A) message could not be recognised 107 = Logon (A) message rejected due to other reasons (see Text (58) for further information) 108 = Unsolicited logout from server (see Text (58) for further information)
	StandardTrailer		Y	

4. Market Data Messages

4.1. Subscription Access Profiles

ASX will offer two FIX Market Data subscription services – FIX Full and FIX Light. Market data connections will be designated with a Subscription Access Profile that permissions the level of market data that can be requested from the FIX server.

- FIX Full users can request order-by-order (full depth), top-of-book, and trade data in both aggregated and non-aggregated form.
- FIX Light users can only request non-aggregated, top-of-book, up to five levels of aggregated depth, and coalesced trades (trade-by-level).

Security Lists, Session Status, and other non-orderbook messages will be available and provide the same business content for both subscriptions.

4.2. MarketDataRequest (V)

This message is used to request market data from the market. It can be used to request a single snapshot (263=0), which will give a one-off picture at that point in time, or a snapshot and subscription (263=1), which will provide a snapshot and ongoing updates.

A request with MDUpdateType=Full Refresh (265=0) will result in a response with a MarketDataSnapshotFullRefresh (W) message. We recommend using SubscriptionRequestType=Snapshot (263=0) in this instance, due to the large volume of messages. A request with MDUpdateType=Incremental Refresh (265=1) will result in a MarketDataIncrementalRefresh (X) message and is recommended where SubscriptionRequestType=Snapshot + Updates (263=1).

Subscribers may choose to limit various parameters, such as the size of requests, whether just the top of book or the entire book is displayed, and whether full or incremental updates are subscribed to. It is possible to specify in the request a combination of instruments, with any imbalances, quotes, open, high, low, close, settlement, total traded volume, and open interest.

Subscription requests are limited to a specific market indicated by the MarketID (Tag 1301) and can be additionally limited to a specific market list indicated by the MarketSegmentID (1300).

Using the MarketDefinitionRequest (BT) message will provide the market lists within a market, and the segments within a market list.

Tag	Name	Data Type	Reqd	Comment
	StandardHeader		Y	MsgType = V
262	MDReqID	String [128]	Y	Identifier for Market Data Request. Must be unique when SubscriptionRequestType = Snapshot + Updates. Ignored when SubscriptionRequestType = Snapshot.

Tag	Name	Data Type	Reqd	Comment
263	SubscriptionRequestType	char	Y	Subscription Request Type. Valid values: 0 = Snapshot 1 = Snapshot + Updates (Subscribe) 2 = Disable previous Snapshot + Update Request (Unsubscribe).
264	MarketDepth	int	Y	Depth of market for Book Snapshot / Incremental updates. Valid values: 0 = Full Book 1 = Top of book 2 and above = number of levels. If AggregatedBook=true (266=Y), values of zero or 5 and above will return results for up to 5 levels of depth.
265	MDUpdateType	int		Specifies the type of Market Data update. A value of 1 should be assumed if this tag is left off the message. 0 = Full Refresh 1 = Incremental refresh.
266	AggregatedBook	Boolean		Specifies whether or not book entries should be aggregated. If not specified, defaults to N. Note that some FIX Light requests must be aggregated. If this tag is omitted from such a request, it will fail because the default of non-aggregated is used. See Subscription Access Profiles for more information. Valid values: Y = book entries (order & trades) to be aggregated

Tag	Name	Data Type	Reqd	Comment
				N = book entries (order & trades) should not be aggregated.
267	NoMDEntryTypes	NumInGroup	Y	Number of MDEntryType fields requested.
> 269	MDEntryType	char	Y	Type of market data entry. Valid values: 0 = Bid 1 = Offer 2 = Trade (Trade-By-Level if Request contains AggregatedBook (266=Y)) 4 = Opening Price 5 = Closing Price 6 = Settlement Price 7 = Trading session high price 8 = Trading session low price A = Imbalance B = Total Traded Volume / Volume (VOI) when paired with 269=C C = Open Interest M = Prior Settle Price Q = Auction Clearing Price (Equilibrium Price). u = Anomalous Order Threshold Reference Price v = Anomalous Order Threshold Upper Range w = Anomalous Order Threshold Lower Range x = Extreme Trade Range Reference Price y = Extreme Trade Range Upper Range z = Extreme Trade Range Lower Range.

Tag	Name	Data Type	Reqd	Comment
146	NoRelatedSym	NumInGroup	Y	Number of symbols requested. This tag is required to comply with the FIX protocol. Valid value: 1
> 55	Symbol	String [255]	Y	The common, human understood representation of the security. This tag is required to comply with FIX protocol. It is not possible to subscribe at an instrument level. Valid value: [N/A]
1310	NoMarketSegments	NumInGroup	C	Number of Market Segments on which a security may trade. Required by ASX.
> 1301	MarketID	Exchange	C	Identifies the market, which lists and trades the instrument. Required, if MarketSegmentID (1300) is supplied. Valid values: XSFE = Sydney Futures Exchange NZFX = New Zealand Futures and Options Exchange.
> 1300	MarketSegmentID	String [128]		Identifies the market list that data is required for. Requires MarketID (1301). If MarketSegmentID is not supplied, data will be supplied for all market lists for the market.
7565	NoTradeSeqNoSeries	NumInGroup		Number of TradeSeqNo repeating group instances. Used for trade replay.
> 7555	TradeSeqNoSeries	int		Subscription group.

Tag	Name	Data Type	Reqd	Comment
> 7554	TradeSeqNo	int		Trade sequence number.
	StandardTrailer		Y	

4.2.1. MarketDataRequest (V) Example for Bid, Offer, and Trade

```
8=FIXT.1.1|9=132|35=V|49=ABCM1|56=ASX|34=6419|52=20161129-06:08:02|262=ABC-V-134|263=1|264=0|267=3|269=0|269=1|
269=2|146=1|55=[N/A]|1310=1|1301=XSFE|10=176|
```

4.2.2. MarketDataRequest (V) Example for Trade Replay

```
8=FIXT.1.1|9=161|35=V|49=ABCM1|56=ASX|34=11967|52=20161130-05:11:23|262=ABC-V-46|263=1|264=0|267=3|269=0|269=1|
269=2|146=1|55=[N/A]|1310=1|1301=XSFE|7565=1|7555=256|7554=3016278|10=033|
```

4.3. MarketDataRequestReject (Y)

The MarketDataRequestReject is returned as a result of an invalid MarketDataRequest (V) due to business or technical reasons.

Tag	Name	Data Type	Reqd	Comment
	StandardHeader		Y	MsgType = Y
262	MDReqID	String [128]	Y	Must refer to the MDReqID of the request.
281	MDReqRejReason	char		Reason for the rejection of a Market Data request. Valid values: 0 = Unknown symbol 1 = Duplicate MDReqID 3 = Insufficient Permissions 4 = Unsupported SubscriptionRequestType 5 = Unsupported MarketDepth 6 = Unsupported MDUpdateType 7 = Unsupported AggregatedBook 8 = Unsupported MDEntryType.
58	Text	String [see comment]		Free format text string. Outgoing messages may exceed 128 characters.
	StandardTrailer		Y	

4.3.1. MarketDataRequestReject (Y) Example

MarketDataRequest (V) with an unknown exchange

```
8=FIXT.1.1|9=133|35=V|49=ABCM1|56=ASX|34=10347|52=20161129-22:28:13|262=ABC-V-12|263=1|264=0|267=3|269=0|269=1|
269=2|146=1|55=[N/A]|1310=1|1301=XSFEX|10=001|
```

Reject message

```
8=FIXT.1.1|9=0000111|35=Y|49=ASX|56=ABCM1|34=83418|52=20161129-22:28:13.557|369=10347|262=ABC-V-12|281=0|
58=Invalid MarketID: XSFEX|10=074|
```

4.4. MarketDataSnapshotFullRefresh (W)

This message is received in response to the MarketDataRequest (V) message, where Tag 263 = 0 (Snapshot) and Tag 265 = 0 (Full Refresh), and is used to give a one-off complete snapshot of a book and its related statistics.

The snapshot will include active orders (bid and offer) to a set market depth and, if requested, statistical information like Opening Price, Closing Price, Trading Session High, Trading Session Low, Total Trade Volume, Open Interest, etc.

Tag	Name	Data Type	Reqd	Comment
	StandardHeader		Y	MsgType = W
75	TradeDate	LocalMktDate		Business trade date.
55	Symbol	String [255]	Y	The common, human understood representation of the security.
48	SecurityID	String [10]		Security identifier value of SecurityIDSource (22) type
22	SecurityIDSource	String		Identifies the source of the SecurityID (48) value. Valid values: 8 = Exchange defined. Available as Tradeable Instrument ID
461	CFICode	String [6]		Indicates the type of security using ISO 10962 standard, CFI code values.
262	MDReqID	String [128]		Conditionally required if this message is in response to a MarketDataRequest.
268	NoMDEntries	NumInGroup	Y	Number of MDEntry repeating group instances in this market data message.

Tag	Name	Data Type	Reqd	Comment
> 269	MDEntryType	char	Y	Type of market data entry. Valid values: 0 = Bid 2 = Trade (Trade-By-Level if Request contains AggregatedBook (266=Y)) 1 = Offer 4 = Opening Price 5 = Closing Price 6 = Settlement Price 7 = Trading session high price 8 = Trading session low price A = Imbalance B = Total Traded Volume / Volume (VOI) when paired with 269=C C = Open Interest J = Empty book M = Prior Settle Price Q = Auction Clearing Price (Equilibrium Price).
> 83	RptSeq	int		Allows sequence number to be specified within a feed type. Time Priority value (order book only)
> 278	MDEntryID	String [41]		Unique Market Data Entry identifier.
> 1023	MDPriceLevel	int		Integer to convey the level of a bid or offer at a given price level. This is in contrast to MDEntryPositionNo, which is used to convey the position of an order within a price level.

Tag	Name	Data Type	Reqd	Comment
> 290	MDEntryPositionNo	int		Display position of a bid or offer, numbered from most competitive to least competitive, per market side, beginning with 1.
> 346	NumberOfOrders	int		In an Aggregated Book, used to show how many individual orders make up an MDEntry.
> 270	MDEntryPx	Price		Price of the Market Data Entry.
> 271	MDEntrySize	Qty		Size of the Market Data Entry
> 272	MDEntryDate	UTCDateOnly		Transaction date of Market Data Entry.
> 273	MDEntryTime	UTCTimeOnly		Transaction time of Market Data Entry.
> 276	QuoteCondition	MultipleStringValue		Used to indicate an Implied order or price Valid values: K = Implied.
> 277	TradeCondition	MultipleStringValue		Type of market data entry. If this message is in response to a MarketDataRequest (V) subscription with AggregatedBook=true (266=Y), the value will be for one of the trades. Other trades that have been aggregated may have different values. Valid values: B = Average Price Trade P = Imbalance More Buyers Q = Imbalance More Sellers U = Exchange Last

Tag	Name	Data Type	Reqd	Comment
				X = Crossed a = Volume Only (Multi leg trade, do not want to report price; only volume) 0 = Cancel R = Opening Price AX = High price. Trade establishes new high price for the session AY = Low price. Trade establishes new low price for the session.
> 574	MatchType	String		The point in the matching process at which this trade was matched. If this message is in response to a MarketDataRequest (V) subscription with AggregatedBook=true (266=Y), the value will be for one of the trades. Other trades that have been aggregated may have different values. Valid values: 5 = Auction 4 = Normal Trade S = Combo-to-Outright R = Combo-to-Combo A = Strip-to-Strip. Indicates an Average Price Type combination order has matched with another Average Price Type combination order. B = Strip-to-Outright. Indicates an Average Price Type combination order has matched with an outright order.
> 828	TrdType	int		Type of trade.

Tag	Name	Data Type	Reqd	Comment
				Valid values: 0 = Regular Trade.
> 811	PriceDelta	float		The rate of change in the price of a derivative with respect to the movement in the price of the underlying instrument(s) upon which the derivative instrument price is based. This value is normally between -1.0 and 1.0. This is reported for options only.
> 1188	Volatility	float		Annualized volatility for option model calculations. This is reported for options only (outside FIX 5.0 Spec).
> 7555	TradeSeqNoSeries	int		Subscription group.
> 7554	TradeSeqNo	int		Trade sequence number.
	StandardTrailer		Y	

4.4.1. MarketDataSnapshotFullRefresh (W) Example

This example shows 2 messages, covering bids and prior settlement price.

```
8=FIXT.1.1|9=0000416|35=W|49=ASX|56=ABCM1|34=147900|52=20161130-06:18:25.748|369=12238|75=20161201|262=ABC-V-48|
55=IBZ6|48=64757|22=8|461=FFNCSO|268=3|269=0|278=6209609344296697857|270=96.1|271=6|272=20161130|273=06:10:36.643|
290=1|1023=1|83=2727|269=0|278=620960934435961241 7|270=96|271=7|272=20161130|273=06:10:36.658|290=1|1023=2|
83=2728|269=1|278=6209609344418332673|270=96.2|271=7|272=20161130|273=06:14:00.998|290=1|1023=1|83=2729|10=212|

8=FIXT.1.1|9=0000176|35=W|49=ASX|56=ABCM1|34=147901|52=20161130-06:18:25.748|369=12238|75=20161201|262=ABC-V-48|
55=IBZ6|48=64757|22=8|461=FFNCSO|268=1|269=M|270=98.56|272=20161130|273=05:40:00.311|10=245|
```

4.5. MarketDataIncrementalRefresh (X)

This message is received in response to the MarketDataRequest (V) message where MDUpdateType = Incremental Refresh (265=1).

A message may contain any combination of new, changed, or deleted Market Data Entries, for any combination of instruments. With any combination of trades, imbalances, quotes, open, high, low, close, settlement, total traded volume, and VOI changes.

Tag	Name	Data Type	Reqd	Comment
	StandardHeader		Y	MsgType = X
75	TradeDate	LocalMktDate		Specifies the business trading date to which the data in this message applies.
262	MDReqID	String [128]		Conditionally required if this message is in response to a MarketDataRequest.
268	NoMDEntries	NumInGroup	Y	Number of MDEntry repeating group instances in this market data message.
> 279	MDUpdateAction	char	Y	Must be the first tag in each repeating group. Valid values: 0 = New 1 = Change 2 = Delete.
> 83	RptSeq	int		Allows sequence number to be specified within a feed type. Time Priority value (Order Book Only).

Tag	Name	Data Type	Reqd	Comment
> 55	Symbol	String [255]		The common, human understood representation of the security.
> 48	SecurityID	String [10]		Security identifier value of SecurityIDSource (22) type.
> 22	SecurityIDSource	String		Identifies the source of the SecurityID (48) value. Valid values: 8 = Exchange defined. Available as Tradeable Instrument ID.
> 461	CFICode	String [6]		Indicates the type of security using ISO 10962 standard, CFI code values.
> 278	MDEntryID	String [41]		Unique Market Data Entry identifier.
> 272	MDEntryDate	UTCDateOnly		Transaction Date of Market Data Entry.
> 273	MDEntryTime	UTCTimeOnly		Transaction Time of Market Data Entry.
> 1023	MDPriceLevel	int		Integer to convey the level of a bid or offer at a given price level. This is in contrast to MDEntryPositionNo, which is used to convey the position of an order within a Price level.

Tag	Name	Data Type	Reqd	Comment
> 269	MDEntryType	char		Type of market data entry. Valid values: 0 = Bid 1 = Offer 2 = Trade (Trade-By-Level if Request contains AggregatedBook (266=Y)) 4 = Opening Price 5 = Closing Price 6 = Settlement Price 7 = Trading session high price 8 = Trading session low price A = Imbalance B = Total Traded Volume / Volume (VOI) when paired with 269=C C = Open Interest J = Empty book M = Prior Settle Price Q = Auction Clearing Price (Equilibrium Price) u = Anomalous Order Threshold Reference Price v = Anomalous Order Threshold Upper Range w = Anomalous Order Threshold Lower Range x = Extreme Trade Range Reference Price y = Extreme Trade Range Upper Range z = Extreme Trade Range Lower Range.
> 290	MDEntryPositionNo	int		Display position of a bid or offer, numbered from most competitive to least competitive, per market side, beginning with 1.

Tag	Name	Data Type	Reqd	Comment
> 346	NumberOfOrders	int		In an Aggregated Book, used to show how many individual orders make up an MDEntry.
> 270	MDEntryPx	Price		Price of the MDEntry.
> 271	MDEntrySize	Qty		Size of the MDEntry.
> 276	QuoteCondition	MultipleStringValue		Used to indicate an Implied order or price. Valid values: K = Implied.
> 277	TradeCondition	MultipleStringValue		Type of market data entry. If this message is in response to a MarketDataRequest (V) subscription with AggregatedBook=true (266=Y), the value will be for one of the trades. Other trades that have been aggregated may have different values. Valid values: 0 = Cancel B = Average Price Trade P = Imbalance More Buyers Q = Imbalance More Sellers R = Opening Price X = Crossed U = Exchange Last a = Volume Only (Do not want to report price; only volume) AX = High price. Trade establishes new high price for the session AY = Low price. Trade establishes new low price for the session.

Tag	Name	Data Type	Reqd	Comment
> 828	TrdType	int		Type of Trade Valid values: 0 = Regular Trade.
> 574	MatchType	String		The point in the matching process at which this trade was matched. If this message is in response to a MarketDataRequest (V) subscription with AggregatedBook=true (266=Y), the value will be for one of the trades. Other trades that have been aggregated may have different values. Valid values: 4 = Normal Trade 5 = Auction A = Strip-to-Strip (Indicates an Average Price Type combination order has matched with another Average Price Type combination order) B = Strip-to-Outright (Indicates an Average Price Type combination order has matched with an outright order) R = Combo-to-Combo S = Combo-to-Outright.
> 811	PriceDelta	float		The rate of change in the price of a derivative with respect to the movement in the price of the underlying instrument(s) upon which the derivative instrument price is based. This value is normally between -1.0 and 1.0. This is reported for options only.

Tag	Name	Data Type	Reqd	Comment
> 1188	Volatility	float		Annualized volatility for option model calculations. This is reported for options only (outside FIX 5.0 Spec).
> 7555	TradeSeqNoSeries	int		Subscription group.
> 7554	TradeSeqNo	int		Trade sequence number.
	StandardTrailer		Y	

4.5.1. MarketDataIncrementalRefresh (X) Example

This example covers the placement of a couple of orders, change to orders and a trade.

New bid placed

```
8=FIXT.1.1|9=0000233|35=X|49=ASX|56=ABCM1|34=108893|52=20161130-00:06:34.367|369=10755|75=20161130|262=ABC-V-134|
268=1|279=0|269=0|278=6209517731138519041|55=APH7|48=58950|22=8|461=FFICSO|270=5376|271=3|272=20161130|
273=00:06:34.364|290=1|1023=1|83=2228|10=075|
```

New ask placed

```
8=FIXT.1.1|9=0000233|35=X|49=ASX|56=ABCM1|34=108901|52=20161130-00:07:15.106|369=10758|75=20161130|262=ABC-V-134|
268=1|279=0|269=1|278=6209517901993492481|55=APH7|48=58950|22=8|461=FFICSO|270=5377|271=4|272=20161130|
273=00:07:15.099|290=1|1023=1|83=2229|10=084|
```

Deletion of bid and ask orders, creation of a new ask for volume of 1

```
8=FIXT.1.1|9=0000487|35=X|49=ASX|56=ABCM1|34=108910|52=20161130-00:08:26.099|369=10763|75=20161130|262=ABC-V-134|
268=3|279=2|269=0|278=6209517731138519041|55=APH7|48=58950|22=8|461=FFICSO|270=5376|271=3|272=20161130|
273=00:08:26.087|290=1|1023=1|279=2|269=1|278=6209517901993492481|55=APH7|48=58950|22=8|461=FFICSO|270=5377|271=4|
272=20161130|273=00:08:26.087|290=1|1023=1|279=0|269=1|278=6209518199738744833|55=APH7|48=58950|22=8|461=FFICSO|
270=5376|271=1|272=20161130|273=00:08:26.087|290=1|1023=1|83=2231|10=049|
```

Trade of 3

```
8=FIXT.1.1|9=0000262|35=X|49=ASX|56=ABCM1|34=108913|52=20161130-00:08:26.099|369=10763|75=20161130|262=ABC-V-134|
268=1|279=0|269=2|278=6209518199738744833|55=APH7|48=58950|22=8|461=FFICSO|270=5376|271=3|272=20161130|
273=00:08:26.087|277=U AY AX R X|828=0|574=4|7555=256|7554=3016279|10=149|
```

4.6. MarketDataIncrementalRefresh (X) Settlement or Closing Version

The final settlement price, prior day settlement price, and prior day close price of an order book is disseminated using MarketDataIncrementalRefresh (X) messages.

Closing price is reported using MDEntryType=Closing Price (269=5). If sent prior to market open, it will be in reference to the previous day's close. If sent on or after the close, it will be in reference to that trade date.

Settlement price is reported as follows:

- **Prior** to the contract moving to the next trade date, the final settlement price is disseminated with this message using MDEntryType=Settlement Price (269=6).
- **After** the contract has moved to the next trade date, the final settlement price is disseminated with this message, using MDEntryType=Prior Settle Price (269=M) with TradeDate (75) set to the current trade date for the contract.

The back-end system may distribute settlement and closing prices more than once, in which case, the same process will apply.

Tag	Name	Data Type	Reqd	Comment
	StandardHeader		Y	MsgType = X
75	TradeDate	LocalMktDate		Specifies the business trading date to which the data in this message applies.
262	MDReqID	String [128]		Conditionally required, if this message is in response to a MarketDataRequest.

Tag	Name	Data Type	Reqd	Comment
268	NoMDEntries	NumInGroup	Y	Number of MDEntry repeating group instances in this market data message.
> 279	MDUpdateAction	char	Y	Must be the first tag in each repeating group. Valid values: 0 = New
> 55	Symbol	String [255]		The common, human understood representation of the security.
> 48	SecurityID	String [10]		Security identifier value of SecurityIDSource (22) type.
> 22	SecurityIDSource	String		Identifies the source of the SecurityID (48) value. Valid values: 8 = Exchange defined. Available as Tradeable Instrument ID.
> 461	CFICode	String [6]		Indicates the type of security using ISO 10962 standard, CFI code values.
> 269	MDEntryType	char		Type of market data entry. Valid values: 5 = Closing Price 6 = Settlement Price M = Prior Settle Price.

Tag	Name	Data Type	Reqd	Comment
> 270	MDEntryPx	Price		Price of either the Closing Price, Settlement Price, or Prior Settlement Price. A value of zero indicates that there is no price for the entry. This may be sent after close and before pre-open. When sent at the close of a business trade date (end of day session, start of night session) TradeDate (75) will be the business trade date of the new day. When sent intraday (end of night session, start of day session) TradeDate (75) will be the current business trade date. Note that multi-leg contracts (spreads) can have a price of zero.
> 272	MDEntryDate	UTCDateOnly		Transaction date of Market Data Entry.
> 273	MDEntryTime	UTCTimeOnly		Transaction time of Market Data Entry.
> 811	PriceDelta	float		The rate of change in the price of a derivative with respect to the movement in the price of the underlying instrument(s) upon which the derivative instrument price is based. This value is normally between -1.0 and 1.0. This is reported for options only when: MDEntryType = Settlement price (269 = 6), or MDEntryType = Prior settle price (269 = M).
> 1188	Volatility	float		Annualized volatility for option model calculations. This is reported for options only (Outside FIX 5.0 Spec) when: MDEntryType = Settlement price (269 = 6), or MDEntryType = Prior settle price (269 = M).
	StandardTrailer		Y	

4.6.1. MarketDataIncrementalRefresh (X) Settlement Example

Settlement on day 1

8=FIXT.1.1|9=0000263|35=X|49=ASX|56=ABCM3|34=19151|52=20161129-05:41:00.585|369=696|**75=20161129**|262=ABC-V-3|268=2|279=0|**269=6**|55=BNH9|48=60252|22=8|461=FCICSO|**270=67.42**|272=20161129|273=05:33:02.099|279=0|269=M|55=BNH9|48=60252|22=8|461=FCICSO|270=66.68|272=20161128|273=06:45:00.110|10=172|

Prior settlement on day 2 (day 1 prices)

8=FIXT.1.1|9=0000191|35=X|49=ASX|56=ABCM3|34=45364|52=20161129-06:45:00.186|369=828|**75=20161130**|262=ABC-V-3|268=1|279=0|**269=M**|55=BNH9|48=60252|22=8|461=FCICSO|270=67.42|272=20161129|273=06:45:00.019|811=0|1188=0|10=110|

Zero settlement price message at start of day 2

8=FIXT.1.1|9=0000187|35=X|49=ASX|56=ABCM3|34=45365|52=20161129-06:45:00.186|369=828|**75=20161130**|262=ABC-V-3|268=1|279=0|**269=6**|55=BNH9|48=60252|22=8|461=FCICSO|**270=0**|272=20161129|273=06:45:00.019|811=0|1188=0|10=140|

4.6.2. MarketDataIncrementalRefresh (X) Example for Closing Price

Closing for day 1

8=FIXT.1.1|9=0000183|35=X|49=ASX|56=ABCM1|34=133576|52=20161130-05:30:00.865|369=12040|**75=20161130**|262=ABC-V-3S90|268=1|279=0|269=5|55=APH7|48=58950|22=8|461=FFICSO|270=5376|272=20161130|273=05:30:00.855|10=033|

Closing zero price message at start of day 2

8=FIXT.1.1|9=0000180|35=X|49=ASX|56=ABCM1|34=139355|52=20161130-05:45:00.935|369=12101|**75=20161201**|262=ABC-V-3S90|268=1|279=0|269=5|55=APH7|48=58950|22=8|461=FFICSO|270=0|272=20161130|273=05:45:00.920|10=122|

4.7. MarketDataIncrementalRefresh (X) Equilibrium Price Version

Equilibrium price information during a pre-open is disseminated using MarketDataIncrementalRefresh (X) messages. During the pre-open period where there are matching or overlapping orders, an indicative match quantity, match price, and an imbalance is sent out to give a picture of the market auction.

When there is a new, deleted, or amended order, that effects either the match quantity, match price, or the imbalance, this message reports the change. In all instances, the UpdateAction is always New.

Tag	Name	Data Type	Reqd	Comment
	StandardHeader		Y	MsgType = X
75	TradeDate	LocalMktDate		
262	MDReqID	String [128]		Unique identifier of the MarketDataRequest.
268	NoMDEntries	NumInGroup	Y	Number of MDEntry repeating group instances in this market data message.
> 279	MDUpdateAction	char	Y	Must be the first tag in each repeating group. When the auction price is cleared or removed, a message with MDUpdateAction=New (279=0) with MDEntryPx=0 (270=0) is sent. Valid values: 0 = New
> 55	Symbol	String [255]		The common, human understood representation of the security.

Tag	Name	Data Type	Reqd	Comment
> 48	SecurityID	String [10]		Security identifier value of SecurityIDSource (22) type.
> 22	SecurityIDSource	String		Identifies the source of the SecurityID (48) value. Valid values: 8 = Exchange defined. Available as Tradeable Instrument ID.
> 461	CFICode	String [6]		Indicates the type of security using ISO 10962 standard, CFI code values.
> 269	MDEntryType	char		Type of market data entry. Valid values: Q = Auction Clearing Price (Equilibrium) A = Imbalance.
> 270	MDEntryPx	Price		Price of the Market Data Entry
> 271	MDEntrySize	Qty		If MDEntryType=Q, MDEntrySize = Matched volume If MDEntryType=A, MDEntrySize = Size of the Imbalance (See Tag 277 for side of Imbalance)
> 272	MDEntryDate	UTCDateOnly		Transaction Date of Market Data Entry.
> 273	MDEntryTime	UTCTimeOnly		Transaction Time of Market Data Entry.

Tag	Name	Data Type	Reqd	Comment
> 277	TradeCondition	MultipleStringValue		Type of market data entry. Valid values: P = Imbalance more buyers Q = Imbalance more sellers.
	StandardTrailer		Y	

4.7.1. MarketDataIncrementalRefresh (X) Equilibrium Price Example

```
8=FIXT.1.1|9=0000274|35=X|49=ASX|56=ABCM1|34=146173|52=20161130-06:10:36.675|369=12206|75=20161201|262=ABC-V-47|
268=2|279=0|269=A|55=IBZ6|48=64757|22=8|461=FFNCSO|271=4|272=20161130|273=06:10:36.672|277=P|279=0|269=Q|55=IBZ6|
48=64757|22=8|461=FFNCSO|270=96.2|271=8|272=20161130|273=06:10:36.672|10=235|

8=FIXT.1.1|9=0000275|35=X|49=ASX|56=ABCM1|34=146178|52=20161130-06:10:36.695|369=12206|75=20161201|262=ABC-V-47|
268=2|279=0|269=A|55=IBZ6|48=64757|22=8|461=FFNCSO|271=5|272=20161130|273=06:10:36.693|277=Q|279=0|269=Q|55=IBZ6|
48=64757|22=8|461=FFNCSO|270=96.2|271=12|272=20161130|273=06:10:36.693|10=038|
```

4.8. MarketDataIncrementalRefresh (X) Open High Low Last and Volume Open Interest Version

Open, High, Low and Last (OHLL) and Volume and Open Interest (VOI) market statistic information is disseminated using the MarketDataIncrementalRefresh (X) message.

Tag	Name	Data Type	Reqd	Comment
	StandardHeader		Y	MsgType = X
75	TradeDate	LocalMktDate		Specifies the business trading date to which the data in this message applies.
262	MDReqID	String [128]		Unique identifier of the Market Data Request.
268	NoMDEntries	NumInGroup	Y	Number of MDEntry repeating group instances in this market data message.
> 279	MDUpdateAction	char	Y	Must be the first tag in each repeating group. Valid values: 0 = New 2 = Delete.
> 55	Symbol	String [255]		The common, human understood representation of the security.
> 48	SecurityID	String [10]		Security identifier value of SecurityIDSource (22) type.

Tag	Name	Data Type	Reqd	Comment
> 22	SecurityIDSource	String		Identifies the source of the SecurityID (48) value. Valid values: 8 = Exchange defined. Available as Tradeable Instrument ID.
> 461	CFICode	String [6]		Indicates the type of security using ISO 10962 standard, CFI code values.
> 269	MDEntryType	char		Type of market data entry. Valid values: 4 = Opening Price 5 = Closing Price 7 = Trading session high price 8 = Trading session low price B = Total Traded Volume / Volume (VOI) when paired with 269=C C = Open Interest.
> 270	MDEntryPx	Price		Price of either the Opening Price, Closing Price, Trading session high price, Trading session low price, Settlement Price or Prior Settlement Price. A value of zero indicates that there is no price for the entry. This may be sent after close and before pre-open. When sent at the close of a business trade date (end of day session, start of night session) TradeDate (75) will be the business trade date of the new day. When sent intraday (end of night session, start of day session) TradeDate (75) will be the current business trade date. Note that multi-leg contracts (spreads) can have a price of zero.

Tag	Name	Data Type	Reqd	Comment
> 271	MDEntrySize	Qty		Quantity or volume of either the Total Trade Volume or Open Interest. A value of zero indicates that there is no quantity for the entry. This may be sent after close and before pre-open. When sent at the close of a business trade date (end of day session, start of night session) TradeDate (75) will be the business trade date of the new day. When sent intraday (end of night session, start of day session) TradeDate (75) will be the current business trade date.
> 272	MDEntryDate	UTCDateOnly		Transaction date of Market Data Entry.
> 273	MDEntryTime	UTCTimeOnly		Transaction time of Market Data Entry.
	StandardTrailer		Y	

4.8.1. MarketDataIncrementalRefresh (X) OHLL and VOI Example — Open, High, Low, Volume

```
8=FIXT.1.1|9=0000479|35=X|49=ASX|56=ABCM1|34=108915|52=20161130-00:08:26.099|369=10763|75=20161130|262=ABC-V-
43423|268=4|279=0|269=B|278=6209518199738744833|55=APH7|48=58950|22=8|461=FFICSO|271=3|272=20161130|
273=00:08:26.087|7555=256|7554=3016279|279=0|269=7|55=APH7|48=58950|22=8|461=FFICSO|270=5376|272=20161130|
273=00:08:26.087|279=0|269=8|55=APH7|48=58950|22=8|461=FFICSO|270=5376|272=20161130|273=00:08:26.087|279=0|269=4|
55=APH7|48=58950|22=8|461=FFICSO|270=5376|272=20161130|273=00:08:26.087|10=214|
```

4.8.2. MarketDataIncrementalRefresh (X) Example for OHLL and VOI — VOI

```
8=FIXT.1.1|9=0000265|35=X|49=ASX|56=ABCM3|34=76054|52=20161129-10:10:06.580|369=1263|75=20161130|262=ABC-V-4|
268=2|279=0|269=C|55=YTZ6|48=69363|22=8|461=FFDCSO|271=575114|272=20161129|273=10:10:06.580|279=0|269=B|55=YTZ6|
48=69363|22=8|461=FFDCSO|271=14300|272=20161129|273=10:10:06.580|10=093|
```

4.9. QuoteRequest (R)

The Quote Request message is used to broadcast an RFQ entered by a market participant to market data subscribers listening for quotes specified in RFQRequest (AH) messages. The QuoteRequest can be for a Bid, Offer, Two-Sided Quote, or for a Crossing.

Tag	Name	Data Type	Reqd	Comment
	StandardHeader		Y	MsgType = R (uppercase)
131	QuoteReqID	String [128]	Y	Unique identifier for a QuoteRequest (35=R).
146	NoRelatedSym	NumInGroup	Y	Number of related symbols (instruments) in this request.
> 55	Symbol	String [255]	C	The common, human understood representation of the security.
> 48	SecurityID	String [10]	C	Security identifier value of SecurityIDSource (22) type. Requires SecurityIDSource (22).
> 22	SecurityIDSource	String	C	Identifies the source of the SecurityID (48) value. Valid values: 8 = Exchange defined. Available as Tradeable Instrument ID.
> 54	Side	char		Side of order. Where this tag is not defined, tag indicates a two-sided quote. Valid values: 1 = Buy 2 = Sell 8 = Crossing.

Tag	Name	Data Type	Reqd	Comment
> 38	OrderQty	Qty		OrderQty requested for RFQ.
	StandardTrailer		Y	

4.9.1. QuoteRequest (R) Example

This example is for an option contract, no volume, buy and sell

```
8=FIXT.1.1|9=0000129|35=R|49=ASX|56=ABCM1|34=38660|52=20161129-06:16:56.429|369=6459|131=6209248549360648193|
146=1|55=IRZ70095000C|48=65319|22=8|54=1|10=162|

8=FIXT.1.1|9=0000129|35=R|49=ASX|56=ABCM1|34=38661|52=20161129-06:16:56.430|369=6459|131=6209248549360648194|
146=1|55=IRZ70095000C|48=65319|22=8|54=2|10=157|
```

4.10. RFQRequest (AH)

The RFQRequest (AH) message is used to subscribe to QuoteRequest (R) messages entered by other trading participants.

Tag	Name	Data Type	Reqd	Comment
	StandardHeader		Y	MsgType = AH
644	RFQReqID	String [128]	Y	RFQ Request ID - used to identify an RFQ Request.
453	NoPartyIDs	NumInGroup		Number of parties in the request. Used to specify the exchange to subscribe to.
> 448	PartyID	String [4]		Used to identify source of PartyID. Required if NoPartyIDs > 0 Required, if PartyIDSource is specified. Valid values: XSFE = Sydney Futures Exchange NZFX = New Zealand Futures and Options Exchange.
> 447	PartyIDSource	char		Identifies class or source of the PartyID (448) value. Required if PartyID is specified. Valid values: G = MIC (ISO 10383 - Market Identifier Code).
> 452	PartyRole	int		The role of the party in the transaction. Required if PartyID is specified. Valid values: 73 = Execution Venue.

Tag	Name	Data Type	Reqd	Comment
146	NoRelatedSym	NumInGroup	Y	Number of related symbols (instruments) in Request.
> 55	Symbol	String [255]	Y	The common, human understood representation of the security. Required, if SecurityID (48) is not present. If SecurityID (48) is specified, set to [N/A]. If not subscribing by specifying instrument, set NoRelatedSym = 1 and Symbol = [N/A]. Request will be rejected if SecurityID (48) is present and Symbol (55) is not [N/A].
> 48	SecurityID	String [10]		Security identifier value of SecurityIDSource (22) type. Requires SecurityIDSource (22). Symbol (55) must be set to [N/A] if SecurityID (48) is specified. Request will be rejected if SecurityID (48) is present and Symbol (55) is not [N/A].
> 22	SecurityIDSource	String		Identifies the source of the SecurityID (48) value. Valid values: 8 = Exchange defined. Available as Tradeable Instrument ID.
	StandardTrailer		Y	

4.10.1. RFQRequest (AH) Example for All Symbols

```
8=FIXT.1.1|9=107|35=AH|49=ABCM1|56=ASX|34=6458|52=20161129-06:16:36|644=ABC-AH-9|453=1|448=XSFE|447=G|452=73|
146=1|55=[N/A]|10=139|
```

5. Reference Data Message Details

The following sections cover the supported reference data messages.

5.1. MarketDefinitionRequest (BT)

The MarketDefinitionRequest message is used to request for market structure information from the respondent that receives this request.

Tag	Name	Data Type	Reqd	Comment
	StandardHeader		Y	MsgType = BT
1393	MarketReqID	String [128]	Y	Unique ID of a Market Definition Request message.
263	SubscriptionRequestType	char	Y	Subscription Request Type. Valid values: 0 = Snapshot
1301	MarketID	Exchange	C	Identifies the market which lists and trades the instrument. Required by ASX. Valid values: XSFE = Sydney Futures Exchange NZFX = New Zealand Futures and Options Exchange.

Tag	Name	Data Type	Reqd	Comment
1300	MarketSegmentID	String [128]		Identifies the market list that option series data is required for. Requires MarketID (1301). If MarketSegmentID is not supplied, data will be supplied for all market lists for the market.
	StandardTrailer		Y	

5.1.1. MarketDefinitionRequest (BT) Example

```
8=FIXT.1.1|9=78|35=BT|49=ABCM1|56=ASX|34=5|52=20161128-03:17:11|1393=ABC-BT-0|263=0|1301=XSFE|10=189|
```

5.2. MarketDefinition (BU)

The MarketDefinition (BU) message is used to provide the structure of a market. Markets are arranged into one or more market lists. Each market list contains one or more segments. A MarketDefinition (BU) message will convey one of the following details:

- Market List
- Segment

When conveying information about a segment, the message will indicate the market list the segment belongs to using the ParentMktSegmID (1325) tag.

Knowledge of market lists and segments is necessary to understand the values required to make subscription requests, in addition to interpreting TradingSessionStatus (h) status messages when they are disseminated.

Tag	Name	Data Type	Reqd	Comment
	StandardHeader		Y	MsgType = BU
1394	MarketReportID	String [128]	Y	Market Definition message identifier.
1393	MarketReqID	String [128]		Unique ID of a Market Definition Request message.
1301	MarketID	Exchange	Y	Identifies the market which lists and trades the instrument. Valid values: XSFE = Sydney Futures Exchange NZFX = New Zealand Futures and Options Exchange.
1300	MarketSegmentID	String [128]		Identifies the market list or segment of the market to which the specific trading rules and listing rules apply.

Tag	Name	Data Type	Reqd	Comment
1396	MarketSegmentDesc	String [255]		Description or name of Market Segment.
1205	NoTickRules	NumInGroup		Number of tick rules.
> 1206	StartTickPriceRange	Price		Starting price range for specified tick increment.
> 1208	TickIncrement	Price		Tick increment for stated price range. Specifies the valid price increments at which a security can be quoted and traded.
1325	ParentMktSegmID	String [128]		Specifies that the market segment specified in this message is a sub-segment of the market segment defined in this field. If present, specifies the ASX Market List that the ASX Market Segment belongs to.
2400	EffectiveBusinessDate	LocalMktDate		Business trade date reference data applies to. Not provided for *ASX Market Lists.*
	StandardTrailer		Y	

5.2.1. MarketDefinition (BU) Example for Market List

```
8=FIXT.1.1|9=0000144|35=BU|49=ASX|56=ABCM1|34=5|52=20161128-03:17:11.766|369=4|1394=ABC-BT-0158728631764-1|
1393=ABC-BT-0|1301=XSFE|1300=COMMODITIES|1396=COMMODITIES|10=078|
```

5.2.2. MarketDefinition (BU) Example for Segments in a Market List

```
8=FIXT.1.1|9=0000198|35=BU|49=ASX|56=ABCM1|34=6|52=20161128-03:17:11.768|369=4|1394=ABC-BT-0158728631764-2|
1393=ABC-BT-0|1301=XSFE|1300=UB_FUTURES|1396=UB_FUTURES|1325=COMMODITIES|1205=1|1206=0.1|1208=0.1|2400=20161128|
10=109|

8=FIXT.1.1|9=0000208|35=BU|49=ASX|56=ABCM1|34=7|52=20161128-03:17:11.768|369=4|1394=ABC-BT-0158728631764-3|
1393=ABC-BT-0|1301=XSFE|1300=UB_INTRASPREADS|1396=UB_INTRASPREADS|1325=COMMODITIES|1205=1|1206=0.1|1208=0.1|
2400=20161128|10=043|

8=FIXT.1.1|9=0000198|35=BU|49=ASX|56=ABCM1|34=8|52=20161128-03:17:11.769|369=4|1394=ABC-BT-0158728631764-4|
1393=ABC-BT-0|1301=XSFE|1300=UB_OPTIONS|1396=UB_OPTIONS|1325=COMMODITIES|1205=1|1206=0.1|1208=0.1|2400=20161128|
10=110|
```

5.3. SecurityListRequest (x)

The SecurityListRequest (x) message is used to request data about single instruments, such as futures and multi-leg combination instruments. Data will be returned in SecurityList (y) messages.

Subscription requests are limited to a market by specifying MarketID (1301). They can be further restricted to a market list by specifying the market list in MarketSegmentID (1300).

Market structure information showing the market lists in a market, and the segments in a market list is provided in MarketDefinition (BU) messages.

Tag	Name	Data Type	Reqd	Comment
	StandardHeader		Y	MsgType = x (lowercase)
320	SecurityReqID	String [128]	Y	ID of a Security Definition Request. Must be unique when SubscriptionRequestType = Snapshot + Updates (263=1). Ignored when SubscriptionRequestType = Snapshot (263=0).
263	SubscriptionRequestType	char		Subscription Request Type. Valid values: 0 = Snapshot 1 = Snapshot + Updates (Subscribe) 2 = Disable previous Snapshot + Update Request (Unsubscribe).

Tag	Name	Data Type	Reqd	Comment
559	SecurityListRequestType	int	Y	Type of Security List Request being made. Valid values: 4 = All Securities 5 = MarketID or MarketID + MarketSegmentID.
1301	MarketID	Exchange	C	Identifies the market which lists and trades the instrument. Required by ASX. Valid values: XSFE = Sydney Futures Exchange NZFX = New Zealand Futures and Options Exchange.
1300	MarketSegmentID	String [128]		Identifies the market list that data is required for. Requires MarketID (1301). If MarketSegmentID is not supplied, data will be supplied for all market lists for the market.
	StandardTrailer		Y	

5.3.1. SecurityListRequest (x) Example Subscribing for Market

```
8=FIXT.1.1|9=82|35=x|49=ABCM1|56=ASX|34=14|52=20161128-03:19:21|320=ABC-x-1|559=5|263=1|1301=XSFE|10=140|
```

5.3.2. SecurityListRequest (x) Example Subscribing to a Market List

```
8=FIXT.1.1|9=102|35=x|49=ABCM1|56=ASX|34=15|52=20161128-03:19:29|320=ABC-x-2|559=5|263=1|1301=XSFE|
1300=INTEREST_RATES|10=221|
```

5.4. SecurityList (y) Multi Leg Version

The security definition for multi-leg products is disseminated using the Security List (y). This message is used for multi-leg products such as combination contracts for inter-spread, intra-spread, packs, strips, and UDCs.

Tag	Name	Data Type	Reqd	Comment
	StandardHeader		Y	MsgType = y (lowercase)
320	SecurityReqID	String [128]	Y	Unique client generated ID of the Security Definition Request.
322	SecurityResponseID	String [128]		ID of Security Response message.
393	TotNoRelatedSym	int		Used to indicate the total number of securities being returned for this request. Used in the event that message fragmentation is required.
560	SecurityRequestResult	int		Result of the Security Request identified by the SecurityReqID. Valid values: 0 = Valid request 1 = Invalid or unsupported request 2 = No instruments found that match selection criteria
1301	MarketID	Exchange		Identifies the market which lists and trades the instrument. Valid values: XSFE = Sydney Futures Exchange NZFX = New Zealand Futures and Options Exchange.

Tag	Name	Data Type	Reqd	Comment
1300	MarketSegmentID	String [128]		Identifies the segment of the market to which the specific trading rules and listing rules apply.
893	LastFragment	Boolean		Indicates whether this message is the last in a sequence of messages for those messages that support fragmentation, such as Security List. Valid values: N = Not last message Y = Last message.
715	ClearingBusinessDate	LocalMktDate		The business trade date that the security description applies to.
146	NoRelatedSym	NumInGroup		Specifies the number of repeating symbols specified.
> 55	Symbol	String [255]		The common, human understood representation of the security.
> 48	SecurityID	String [10]		The security identifier value of SecurityIDSource (22) type.
> 22	SecurityIDSource	String		Identifies the source of the SecurityID (48) value. Valid values: 8 = Exchange defined. Available as Tradeable Instrument ID.

Tag	Name	Data Type	Reqd	Comment
> 207	SecurityExchange	Exchange		Market used to help identify a security. Valid values: XSFE = Sydney Futures Exchange NZFX = New Zealand Futures and Options Exchange.
> 461	CFICode	String [6]		Indicates the type of security using ISO 10962 standard, Classification of Financial Instruments (CFI code) values.
> 1151	SecurityGroup	String [6]		The commodity code of the contract - e.g. XT.
> 107	SecurityDesc	String [255]		Textual description for the instrument.
> 969	MinPriceIncrement	float		The minimum price increment for the instrument.
> 15	Currency	Currency		Identifies currency used for price.
> 1378	MultilegPriceMethod	int		This code is used to represent how the multileg price should be interpreted when applied to legs. Valid values: 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 (legs are same side, i.e. bid buys all legs or ask sells all legs).
> 555	NoLegs	NumInGroup		Number of legs that make up the Security.

Tag	Name	Data Type	Reqd	Comment
> > 600	LegSymbol	String [255]		The common, human understood representation of the security.
> > 602	LegSecurityID	String [10]		Security identifier value of LegSecurityIDSource (603) type.
> > 603	LegSecurityIDSource	String		Identifies the source of the LegSecurityID (602) value. Valid values: 8 = Exchange defined.
> > 623	LegRatioQty	float		The ratio of quantity for this individual leg per single lot of the multileg security.
> > 624	LegSide	char		The side of this individual leg of the multileg security. Specifies if this leg is buy or sell when the combination order it belongs to is buy. Valid values: 1 = Buy 2 = Sell.
> > 566	LegPrice	Price		Used to specify a fixed price for a leg as part of the definition of the strategy. Returned for the first leg only, when MultilegPriceMethod=Individual (1378=3).
> 7555	TradeSeqNoSeries	int		Subscription group.
	StandardTrailer		Y	

5.4.1. SecurityList (y) Example for Multi Leg

```
8=FIXT.1.1|9=0039488|35=y|49=ASX|56=ABCM1|34=262|52=20161128-03:19:21.424|369=14|715=20161128|320=ABC-x-1|
322=RABC-x-1|560=0|393=2373|1301=XSFE|1300=IR_INTRASPREADS|893=N|146=190|55=IRZ6H7|48=71269|22=8|1151=IR|
461=FFNXXS|969=0.01|207=XSFE|107=ASX 90 Day Bank Accepted Bills Futures|1378=0|15=AUD|555=2|600=IRZ6|602=65014|
603=8|623=1|624=1|600=IRH7|602=64999|603=8|623=1|624=2|7555=256|55=IRZ6M7|48=71270|22=8|1151=IR|461=FFNXXS|
969=0.01|207=XSFE|107=ASX 90 Day Bank Accepted Bills Futures|1378=0|15=AUD|555=2|600=IRZ6|602=65014|603=8|623=1|
624=1|600=IRM7|602=65004|603=8|623=1|624=2|7555=256|
```

… rest of message omitted for brevity …

```
55=IRM1U1|48=77963|22=8|1151=IR|461=FFNXXS|969=0.01|207=XSFE|107=ASX 90 Day Bank Accepted Bills Futures|1378=0|
15=AUD|555=2|600=IRM1|602=65008|603=8|623=1|624=1|600=IRU1|602=77937|603=8|623=1|624=2|7555=256|10=187|
```

5.5. SecurityList (y) Single Future Version

Tradeable futures instrument data is returned using the SecurityList (y) message. Several SecurityList messages may be issued as indicated by the LastFragment (893) and TotNoRelatedSym (393) tags.

This describes the required, optional, and conditional tags required for outright futures contracts.

Tag	Name	Data Type	Reqd	Comment
	StandardHeader		Y	MsgType = y (lowercase)
322	SecurityResponseID	String [128]		Identifier for the Security List message.
320	SecurityReqID	String [128]		Unique ID of a Security Definition Request.
393	TotNoRelatedSym	int		Used to indicate the total number of securities being returned for this request. Used in the event that message fragmentation is required.
560	SecurityRequestResult	int		Result of the Security Request identified by the SecurityReqID. Valid values: 0 = Valid request 1 = Invalid or unsupported request 2 = No instruments found that match selection criteria.

Tag	Name	Data Type	Reqd	Comment
1301	MarketID	Exchange		Identifies the market which lists and trades the instrument. Valid values: XSFE = Sydney Futures Exchange NZFX = New Zealand Futures and Options Exchange.
1300	MarketSegmentID	String [128]		Identifies the segment of the market to which the specific trading rules and listing rules apply.
893	LastFragment	Boolean		Indicates whether this message is the last in a sequence of messages for those messages that support fragmentation, such as Security List. Valid values: N = Not last message Y = Last message.
715	ClearingBusinessDate	LocalMktDate		The business trade date that the security description applies to.
146	NoRelatedSym	NumInGroup		Specifies the number of repeating symbols specified.
> 55	Symbol	String [255]		The common, human understood representation of the security.
> 48	SecurityID	String [10]		The security identifier value of SecurityIDSource (22) type.

Tag	Name	Data Type	Reqd	Comment
> 22	SecurityIDSource	String		Identifies the source of the SecurityID (48) value. Valid values: 8 = Exchange defined. Available as Tradeable Instrument ID.
> 207	SecurityExchange	Exchange		Market used to help identify a security. Valid values: XSFE = Sydney Futures Exchange NZFX = New Zealand Futures and Options Exchange.
> 461	CFICode	String [6]		Indicates the type of security using ISO 10962 standard, CFI code values.
> 107	SecurityDesc	String [255]		Textual description for the instrument.
> 1151	SecurityGroup	String [6]		The commodity code of the contract, e.g. XT.
> 969	MinPriceIncrement	float		The minimum price increment for the instrument.
> 200	MaturityMonthYear	MonthYear		Can be used with standardized derivatives versus the MaturityDate (541) field. Month and Year of the maturity (used for standardized futures and options).
> 541	MaturityDate	LocalMktDate		The Last Trading Date as specified in the ASX 24 contract specifications.
> 1079	MaturityTime	TZTimeOnly		Time of security's maturity on its last trading date, expressed in local time with offset to UTC specified.

Tag	Name	Data Type	Reqd	Comment
> 223	CouponRate	Percentage		The rate of interest that, when multiplied by the principal, par value, or face value of a bond, provides the currency amount of the periodic interest payment. The coupon is always cited, along with maturity, in any quotation of a bond's price. Exists only for debt products.
> 231	ContractMultiplier	float		The Contract Unit as specified in the ASX 24 contract specifications, e.g. for ASX 90 Day Bank Bills (commodity code IR), this will be 1,000,000. For Options on ASX SPI Index Futures (commodity code AP), this will be 25.
> 870	NoInstrAttrib	NumInGroup		The number of derivative events.
> > 871	InstrAttribType	int		The type of event being described. Valid values: 99 = Other (Used by ASX to define the settlement date for the futures instrument).
> > 872	InstrAttribValue	String [see comment]		The Settlement Day as specified in the ASX 24 contract specifications. Enter in LocalMktDate format.
> 1234	NoLotTypeRules	NumInGroup		Number of Lot Types; set to 1.
> > 1093	LotType	char		Defines the lot type assigned to the order. Valid values: 3 = Block Lot 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.

Tag	Name	Data Type	Reqd	Comment
> > 1231	MinLotSize	Qty		Minimum lot size allowed based on lot type specified in LotType (1093).
> 15	Currency	Currency		Identifies currency used for price.
> 7555	TradeSeqNoSeries	int		Subscription group.
	StandardTrailer		Y	

5.5.1. SecurityList (y) Example for Single Future

```
8=FIXT.1.1|9=0001509|35=y|49=ASX|56=ABCM1|34=210|52=20161128-03:19:21.410|369=14|715=20161128|320=ABC-x-1|
322=RABC-x-1|560=0|393=2373|1301=XSFE|1300=AP_FUTURES|893=N|146=7|55=APH7|48=58950|22=8|1151=AP|461=FFICSO|
200=201703|541=20170316|1079=12:00:00|231=25|969=1|207=XSFE|107=ASX SPI 200 Index Futures|870=1|871=99|
872=20170317|1234=1|1093=3|1231=1|15=AUD|7555=256|55=APM7|48=58951|22=8|1151=AP|461=FFICSO|200=201706|
541=20170615|1079=12:00:00|231=25|969=1|207=XSFE|107=ASX SPI 200 Index Futures|870=1|871=99|872=20170616|1234=1|
1093=3|1231=1|15=AUD|7555=256|55=APU7|48=58954|22=8|1151=AP|461=FFICSO|200=201709|541=20170921|1079=12:00:00|
231=25|969=1|207=XSFE|107=ASX SPI 200 Index Futures|870=1|871=99|872=20170922|1234=1|1093=3|1231=1|15=AUD|
7555=256|55=APZ6|48=58956|22=8|1151=AP|461=FFICSO|200=201612|541=20161215|1079=12:00:00|231=25|969=1|207=XSFE|
107=ASX SPI 200 Index Futures|870=1|871=99|872=20161216|1234=1|1093=3|1231=1|15=AUD|7555=256|55=APZ7|48=58957|
22=8|1151=AP|461=FFICSO|200=201712|541=20171221|1079=12:00:00|231=25|969=1|207=XSFE|107=ASX SPI 200 Index Futures|
870=1|871=99|872=20171222|1234=1|1093=3|1231=1|15=AUD|7555=256|55=APH8|48=79493|22=8|1151=AP|461=FFICSO|
200=201803|541=20180315|1079=12:00:00|231=25|969=1|207=XSFE|107=ASX SPI 200 Index Futures|870=1|871=99|
872=20180316|1234=1|1093=3|1231=1|15=AUD|7555=256|55=APF7|48=81941|22=8|1151=AP|461=FFICSO|200=201701|
541=20170119|1079=12:00:00|231=25|969=1|207=XSFE|107=ASX SPI 200 Index Futures|870=1|871=99|872=20170120|1234=1|
1093=3|1231=1|15=AUD|7555=256|10=037|
```

5.6. DerivativeSecurityList (AA)

Option series data is returned using the DerivativeSecurityList (AA) message. The series of contracts are delivered by Classification of Financial Instruments (CFI) code, so a transmission will contain only puts or calls for a specific maturity date. Several DerivativeSecurityList messages may be used to satisfy one combination of option series, CFI code, and maturity date, as indicated by the LastFragment (893) and TotNoRelatedSym (393) tags.

Tag	Name	Data Type	Reqd	Comment
	StandardHeader		Y	MsgType = AA
322	SecurityResponseID	String [128]	Y	Identifier for this DerivativeSecurityList message.
320	SecurityReqID	String [128]		Unique client generated ID of the Security Definition Request.
893	LastFragment	Boolean		Indicates whether this message is the last in a sequence of messages for those messages that support fragmentation, such as Security List. Valid values: N = Not last message Y = Last message.
560	SecurityRequestResult	int		Result of the Security Request identified by the SecurityReqID. Valid values: 0 = Valid request 1 = Invalid or unsupported request 2 = No instruments found that match selection criteria.

Tag	Name	Data Type	Reqd	Comment
715	ClearingBusinessDate	LocalMktDate		The business trade date that the security description applies to.
393	TotNoRelatedSym	int		Used to indicate the total number of securities being returned for this request. Used in the event that message fragmentation is required.
311	UnderlyingSymbol	String [255]		Underlying security's Symbol – the common, human understood representation of the security.
309	UnderlyingSecurityID	String [10]		Underlying security's SecurityID value of UnderlyingSecurityIDSource (305) type.
305	UnderlyingSecurityIDSource	String		Underlying security's SecurityIDSource. Valid values: 8 = Exchange defined. Available as Tradeable Instrument ID.
308	UnderlyingSecurityExchange	Exchange		Underlying security's SecurityExchange.
435	UnderlyingCouponRate	Percentage		The rate of interest that, when multiplied by the principal, par value, or face value of a bond, provides the currency amount of the periodic interest payment. The coupon is always cited, along with maturity, in any quotation of a bond's price. Exists only for debt products.
246	UnderlyingFactor	float		For debt products, indicates the face value of the bond or bill.

Tag	Name	Data Type	Reqd	Comment
1248	DerivativeCFICode	String [6]		Indicates the type of security using ISO 10962 standard, CFI code values.
1247	DerivativeSecurityGroup	String [6]		The commodity code of the option series, e.g. XT.
1279	DerivativeSecurityDesc	String [255]		Textual description for the instrument.
1272	DerivativeSecurityExchange	Exchange		Market used to help identify a security. Valid values: XSFE = Sydney Futures Exchange NZFX = New Zealand Futures and Options Exchange.
1576	DerivativePriceQuoteCurrency	Currency		Default currency in which the price is quoted.
1267	DerivativeMinPriceIncrement	float		Minimum price increment for the instrument.
1251	DerivativeMaturityMonthYear	MonthYear		Specifies the month and year of maturity.
1252	DerivativeMaturityDate	LocalMktDate		The Last Trading Day as specified in the ASX 24 contract specifications.
1253	DerivativeMaturityTime	TZTimeOnly		Time of security's maturity on its last trading date expressed in local time with offset to UTC specified.
1266	DerivativeContractMultiplier	float		The Contract Unit of the derivative instrument as specified in the ASX 24 contract specifications, e.g. for ASX 90 Day Bank Bills (commodity code IR), this will be 1,000,000. For Options on ASX SPI Index Futures (commodity code AP), this will be 25.
1286	NoDerivativeEvents	NumInGroup		Number of derivative events.

Tag	Name	Data Type	Reqd	Comment
> 1287	DerivativeEventType	int		The type of event being described. Valid values: 99 = Other (Used for derivative settlement date by ASX).
> 1288	DerivativeEventDate	LocalMktDate		Date of event type.
1310	NoMarketSegments	NumInGroup		Number of Market Segments on which a security may trade.
> 1301	MarketID	Exchange		Identifies the market which lists and trades the instrument. Valid values: XSFE = Sydney Futures Exchange NZFX = New Zealand Futures and Options Exchange.
> 1300	MarketSegmentID	String [128]		Identifies the segment of the market to which the specific trading rules and listing rules apply.
> 1234	NoLotTypeRules	NumInGroup		Number of Lot Types; set to 1.
> > 1093	LotType	char		Defines the lot type assigned to the order. Valid values: 3 = Block Lot (Indicates the number of lots that represent a block lot for the order book).
> > 1231	MinLotSize	Qty		Minimum lot size allowed based on lot type specified in LotType (1093).
146	NoRelatedSym	NumInGroup		Specifies the number of repeating symbols specified.

Tag	Name	Data Type	Reqd	Comment
> 55	Symbol	String [255]		The common, human understood representation of the security. Provided NoRelatedSym > 0.
> 48	SecurityID	String [10]		Security identifier value of SecurityIDSource (22) type.
> 22	SecurityIDSource	String		Identifies the source of the SecurityID (48) value. Valid values: 8 = Exchange defined. Available as Tradeable Instrument ID.
> 202	StrikePrice	Price		Strike Price for an Option.
	StandardTrailer		Y	

5.6.1. DerivativeSecurityList (AA) Example

```
8=FIXT.1.1|9=0001527|35=AA|49=ASX|56=ABCM1|34=378|52=20161128-03:29:09.018|369=54|320=ABC-z-3|322=ABC-z-3-R15|
560=0|715=20161128|311=IRH8|309=65000|305=8|308=XSFE|1247=IR|1248=OCAFPS|1576=AUD|1251=201803|1252=20180302|
1253=12:30:00|1266=1000000|1267=0.005|1272=XSFE|1279=Options on ASX 90 Day Bank Accepted Bills Futures|1286=1|
1287=99|1288=20180302|1310=1|1301=XSFE|1300=IR_OPTIONS|1234=1|1093=3|1231=1|393=12842|893=N|146=28|
55=IRH80096500C|48=77760|22=8|202=96.5|55=IRH80096625C|48=77761|22=8|202=96.625|55=IRH80096750C|48=77762|22=8|
202=96.75|55=IRH80096875C|48=77763|22=8|202=96.875|55=IRH80097000C|48=77764|22=8|202=97|55=IRH80097125C|48=77765|
22=8|202=97.125|55=IRH80097250C|48=77766|22=8|202=97.25|55=IRH80097375C|48=77767|22=8|202=97.375|55=IRH80097500C|
48=77768|22=8|202=97.5|55=IRH80097625C|48=77769|22=8|202=97.625|55=IRH80097750C|48=77770|22=8|202=97.75|
55=IRH80097875C|48=77771|22=8|202=97.875|55=IRH80098000C|48=77772|22=8|202=98|55=IRH80098125C|48=77773|22=8|
202=98.125|55=IRH80098250C|48=77774|22=8|202=98.25|55=IRH80098375C|48=77775|22=8|202=98.375|55=IRH80098500C|
48=77776|22=8|202=98.5|55=IRH80098625C|48=77777|22=8|202=98.625|55=IRH80098750C|48=77778|22=8|202=98.75|
55=IRH80098875C|48=77779|22=8|202=98.875|55=IRH80099000C|48=77780|22=8|202=99|55=IRH80099125C|48=77781|22=8|
202=99.125|55=IRH80099250C|48=77782|22=8|202=99.25|55=IRH80099375C|48=77783|22=8|202=99.375|55=IRH80099500C|
```

```
48=77784|22=8|202=99.5|55=IRH80099625C|48=77785|22=8|202=99.625|55=IRH80099750C|48=77786|22=8|202=99.75|
55=IRH80099875C|48=77787|22=8|202=99.875|10=190|
```

5.7. DerivativeSecurityListRequest (z)

DerivativeSecurityListRequest (z) is used to query for option series. Other types of securities, for example, futures, spreads, and UDCs are requested using the SecurityListRequest (x) message.

Securities are returned using DerivativeSecurityList (AA) messages.

Tag	Name	Data Type	Reqd	Comment
	StandardHeader		Y	MsgType = z (lowercase)
320	SecurityReqID	String [128]	Y	ID of a Security Definition Request. Must be unique when SubscriptionRequestType = Snapshot + Updates (263=1). Ignored when SubscriptionRequestType = Snapshot (263=0).
263	SubscriptionRequestType	char		Subscription Request Type. Valid values: 0 = Snapshot 1 = Snapshot + Updates (Subscribe). 2 = Disable previous Snapshot + Update Request (Unsubscribe).
559	SecurityListRequestType	int	Y	Type of Security List Request being made. Valid values: 4 = All Securities 5 = MarketID or MarketID + MarketSegmentID.

Tag	Name	Data Type	Reqd	Comment
1301	MarketID	Exchange	C	Identifies the market which lists and trades the instrument. Required if MarketSegmentID (1300) is supplied or SecurityListRequestType=MarketID (559=5). Request will be rejected if MarketID (1301) is provided and SecurityListRequestType=All Securities (559=4). Valid values: XSFE = Sydney Futures Exchange NZFX = New Zealand Futures and Options Exchange.
1300	MarketSegmentID	String [128]	C	Identifies the market list that option series data is required for. Requires MarketID (1301). If MarketSegmentID is not supplied, data will be supplied for all market lists for the market.
	StandardTrailer		Y	

5.7.1. DerivativeSecurityListRequest (z) Example

```
8=FIXT.1.1|9=76|35=z|49=ABCM1|56=ASX|34=54|52=20161128-03:29:08|320=ABC-z-3|559=5|1301=XSFE|10=149|
```

5.8. SecurityStatus (f)

A SecurityStatus (f) will be disseminated if a tradeable instrument is placed into a trading state due to a non-scheduled trading event, such as an early close due to expiry or a regulatory halt.

Only those subscribers who have subscribed to (or requested a snapshot of) the market list, for that tradeable instrument using the TradingSessionStatusRequest (g), will receive this message.

Once a SecurityStatus (f) message has been received for a tradeable instrument, all further changes in status will be communicated using SecurityStatus (f) messages until the end of the current business trade date for that market segment.

Tag	Name	Data Type	Reqd	Comment
	StandardHeader		Y	MsgType = f (lowercase)
324	SecurityStatusReqID	String [128]		ID supplied in TradSesReqID (335) of the TradingSessionStatusRequest (g) message subscription that generated this message.
55	Symbol	String [255]	Y	The common, human understood representation of the security.
48	SecurityID	String [10]	Y	Security identifier value of SecurityIDSource (22) type.
22	SecurityIDSource	String	Y	Identifies the source of the SecurityID (48) value. Valid values: 8 = Exchange defined. Available as Tradeable Instrument ID.
75	TradeDate	LocalMktDate		The business trade date that the state change applies to.

Tag	Name	Data Type	Reqd	Comment
326	SecurityTradingStatus	int		Identifies the trading status applicable to the transaction. Valid values: 2 = Trading halt 3 = Resume 17 = Ready to trade (start of session) – OPEN 18 = Not available for trading (end of session) – CLOSE 20 = System close 21 = Pre-open. 101 = Maintenance.
327	HaltReason	int		Only required when Tag 326=2 (Halt). Valid values: 103 = The order book was halted due to technical problems. 104 = There is a regulatory reason for the Trade Halt.
461	CFICode	String [6]		Indicates the type of security using ISO 10962 standard, Classification of Financial Instruments (CFI code) values.
58	Text	String [see comment]		String representation of the Session State (Tag 326) or Halt Reason (Tag 327). Outgoing messages may exceed 128 characters.
	StandardTrailer		Y	

5.8.1. Security Status (f) Example

```
8=FIXT.1.1|9=0000150|35=f|49=ASX|56=ABCM1|34=6724|52=20161129-00:03:02.490|369=4964|324=ABC-g-20161128-03:34:35|
55=IBX7|48=64756|22=8|461=FFNCSO|75=20161129|326=2|327=104|10=049|

8=FIXT.1.1|9=0000160|35=f|49=ASX|56=ABCM1|34=7103|52=20161129-01:30:00.847|369=5308|324=ABC-g-20161128-03:34:35|
55=IBX60098500C|48=75254|22=8|461=OCAFPS|75=20161129|326=18|58=CLOSE|10=177|
```

5.9. TradingSessionStatusRequest (g)

The TradingSessionStatusRequest (g) message is used to manage subscriptions to updates about trading status on the market.

Subscription requests must be limited to a market by specifying the MarketID (1301). Additionally, they can be limited to a market list by specifying the market list in MarketSegmentID (1300). Notification of status changes will be communicated using the TradingSessionStatus (h) and SecurityStatus (f) messages. Please refer to them for full details.

Market structure information, showing the market lists in a market and the segments in a market list, is provided in MarketDefinition (BU) messages.

Tag	Name	Data Type	Reqd	Comment
	StandardHeader		Y	MsgType = g (lowercase)
335	TradSesReqID	String [128]	Y	Unique ID of a Trading Session Status message. If SubscriptionRequestType = 2, ID of previous Trading Session Status Request to disable.
1301	MarketID	Exchange		Identifies the market which lists and trades the instrument. Valid values: XSFE = Sydney Futures Exchange NZFX = New Zealand Futures and Options Exchange.
1300	MarketSegmentID	String [128]	C	Identifies the market list that the data is required for. Requires MarketID (1301). If MarketSegmentID is not supplied, data will be supplied for all market lists for the market.

Tag	Name	Data Type	Reqd	Comment
263	SubscriptionRequestType	char	Y	Valid values: 0 = Snapshot 1 = Snapshot + Updates (Subscribe) 2 = Disable previous Snapshot + Update Request (Unsubscribe).
	StandardTrailer		Y	

5.9.1. TradingSessionStatusRequest (g) Example

```
8=FIXT.1.1|9=92|35=g|49=ABCM1|56=ASX|34=77|52=20161128-03:34:35|335=ABC-g-20161128-03:34:35|1301=XSFE|263=1|
10=157|
```

5.10. TradingSessionStatus (h)

The TradingSessionStatus (h) message conveys the trading status of segment.

This message will be sent out in response to scheduled status changes, such as moving from pre-open to open, with one exception.

- For non-scheduled trading changes or known schedules outside the normal segment schedule (early expiry), the instrument status will be notified using the SecurityStatus (f) message.

Note that status is reported at the market segment level, not the market list level. Each market segment in a market list may have a different schedule to other market segments in the market list.

Tag	Name	Data Type	Reqd	Comment
	StandardHeader		Y	MsgType = h (lowercase)
335	TradSesReqID	String [128]	Y	Unique ID of the TradingSessionStatusRequest (g) message that this message is in response to.
336	TradingSessionID	String [1]	Y	Identifier for trading session Valid values: 1 = Day (Day session).
1301	MarketID	Exchange		Identifies the market which lists and trades the instrument. Valid values: XSFE = Sydney Futures Exchange NZFX = New Zealand Futures and Options Exchange.

Tag	Name	Data Type	Reqd	Comment
1300	MarketSegmentID	String [128]		Segment to which the trading session status applies.
75	TradeDate	LocalMktDate		Business day for the trading session status.
58	Text	String [see comment]		Free format text string describing status change. Outgoing messages may exceed 128 characters.
340	TradSesStatus	int		State of the trading session. Valid values: 0 = Unknown 1 = Halted 2 = Open 3 = Closed 4 = Pre-Open 101 = Maintenance.
	StandardTrailer		Y	

5.10.1. TradingSessionStatus (h) Example

```
8=FIXT.1.1|9=0000138|35=h|49=ASX|56=ABCM1|34=763|52=20161128-03:34:35.498|369=76|335=ABC-g-20161128-03:34:35|
1301=XSFE|1300=WM_OPTIONS|75=20161128|336=1|340=2|10=201|
```

6. General Messages

6.1. News (B)

The News (B) message is used to disseminate text information to subscribers.

Tag	Name	Data Type	Reqd	Comment
	StandardHeader		Y	MsgType = B
148	Headline	String [4]	Y	Description of information being transmitted
42	Orig Time	UTCTimestamp	Y	Time of message origination.
33	NoLinesOfText	NumInGroup	Y	Number of Test (58) repeating group instances.
> 58	Text	String [see comment]	Y	Detailed description, or repetition of Headline (148), if no additional information to be transmitted. Outgoing messages may exceed 128 characters.
	StandardTrailer		Y	

6.1.1. News (B) Example

```
8=FIXT.1.1|9=0000128|35=B|49=ASX|56=ABCM4|34=12137|52=20161104-01:31:01.495|369=14034|42=20161104-01:31:01.494|
148=News|33=1|58=IR VOLS X6 0.120 1.0|10=040|
```

6.2. BusinessMessageReject (j)

The BusinessMessageReject message can reject an application-level message, which fulfills session-level rules and cannot be rejected via any other means. Typically, these are unsupported application messages or application messages lacking a specific reject message. If the message fails a session-level rule (for example, incorrect body length), a session-level Reject message should be issued.

Tag	Name	Data Type	Reqd	Comment
	StandardHeader		Y	MsgType = j (lowercase)
45	RefSeqNum	SeqNum		MsgSeqNum of rejected message.
372	RefMsgType	String	Y	The MsgType of the FIX message being referenced.
380	BusinessRejectReason	int	Y	Code to identify reason for a Business Message Reject message. Valid values: 0 = Other – see Text (58) for further details 3 = Unsupported Message Type 5 = Conditionally required field missing.
379	BusinessRejectRefID	String [128]		The value of the business-level "ID" field on the message being referenced. Required unless the corresponding ID field was not specified. The list of ID fields is available in the description of the BusinessMessageReject (j) message in Volume 1 of the FIX 5.0 SP2 specification.

Tag	Name	Data Type	Reqd	Comment
58	Text	String [see comment]		Where possible, a text to explain the reason for rejection. Outgoing messages may exceed 128 characters.
	StandardTrailer		Y	

6.2.1. BusinessMessageReject (j) Example

```
8=FIXT.1.1|9=0000106|35=j|49=ASX|56=ABCM1|34=578|52=20161128-03:33:34.544|369=71|45=72|372=D|380=3|58=Unsupported
Message Type|10=109|
```

Disclaimer

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®