ASX CHESS Replacement

FIX Specification

NOVEMBER 2024

Table of Contents

1 Introduction

1.1 FIX Introduction

The Financial Information Exchange ("FIX") protocol is a standardised messaging protocol used in the finance industry for securities related transactions. It provides a common language for communication between different entities involved in trading, such as buy-side firms, sell-side firms, and ASX Clear.

The new Clearing Service will utilise the FIX protocol between Approved Market Operators (AMOs) and the new clearing service during the Trade Capture and Pricing processes.

The new Clearing Service FIX Gateway supports FIX version 5.0 SP2.

Information relating to FIX is available from the FIX Trading Community and includes all the specifications. Unless specifically stated, field numbers, names, and data types are as published by this FIX specification. A full explanation of the FIX protocol is out of scope for this document and customers should refer to the information published by the FIX Trading Community website for a full understanding of the protocol prior to using this guide.

Link: https://www.fixtrading.org/

Any questions with relation to the FIX Specification document should be directed to chessreplacement@asx.com.au

2 Document Information

The document describes:

- FIX overview
- FIX messaging specifications
- Rejection scenarios
- Recovery and failover processes
- Common message structures, including:
    - Standard header
    - Trailer definitions
- FIX Session Layer
    - **Logon** (A)
    - **Heartbeat** (0)
    - **TestRequest** (1)
    - **ResendRequest** (2)
    - **Reject** (3)
    - **SequenceReset** (4)
    - **Logout** (5)
    - **BusinessMessageReject** (j)
- FIX Application Messages
    - **TradeCaptureReport** (AE)
    - **TradeCaptureReportAck** (AR)
    - **MarketDataSnapshotFullRefresh** (W)
    - **ConfirmationAck** (AU)
    - **SecurityDefinition** (d)

This document is written from ASX Clear's perspective. Inbound messages refer to messages that ASX receives, and outbound messages refers to messages sent from ASX Clear.

2.1 Version History

This document will be iteratively updated as the relevant information becomes available and has been revised according to the table below:

Version	Date	Comments / Change Log
1.0	12 June 2024	Initial release to Approved Market Operators

1.1	12 Nov 2024	• Added *TradeReportTransType* (487) tag to **TradeCaptureReportAck** (AR) message indicating acknowledgements for new or trade cancellations
		• Updated *Text* (58) description of the **TradeCaptureReportAck** (AR) message pertaining to business validation failure examples to include buyer and sellers to pending messages
		• Updated **TradeCaptureReportAck** (AR) text field message in the case of cancellation scenario
		• Removed **TradeCaptureReport** (AR) Rejection Messages for:
		  ○ non-active trading or clearing partipants
		  ○ circumstances where certain link conditions are not met
		• added non allowable values for PartyRole to include values 21 or 119
		• Clarified valid and required values for *NoPartyIDs* (453) repeating group description
		• Updated *PartyRole* (452) from int (2) to int (3) to cater for 119 – Central Securities Depository (CSD)
		• Updated **TradeCaptureReport** (AE) where field name *Side* (54) to only have value 2 present (as opposed to both 2 or 5)
		• Updated ResetSeqNumFlag (141) requirement to 'Optional'
		• Addition of the new **SecurityDefinition** (d) Message
		• Additional New Sections:
		  ○ 4.3 CHESS FIX Gateway Failover
		  ○ 4.4 FIX Client Session Recover Process4.6 Onboarding
		  ○ 4.7 Encryption
		  ○ 4.8 Alternate File Interface
		• Removed section 4.5 Possible Resends and Duplicated Messages
		• Updated legal entity description when referring to ASX Clear and clarified Introduction to FIX Gateway
		• Uplifted Fig 3.5.1 Communication Workflow

2.2 Glossary

Glossary	Description
AMO	Approved Market Operator. A company that is an approved trading venue in Australia
ASCII	American Standard Code for Information Interchange. This is a character encoding standard for electronic communication
CCP	Central Clearing Participant
CSP	Clearing & Settlement Participant
FIX	Financial Information eXchange Protocol
TLS	Transport Level Security. This is the encryption method that is supported by FIX
UTC	Coordinated Universal Time is the primary 24-hour time standard that the world regulates time

3 FIX Overview

3.1 Formatting

A FIX message is composed of a collection of "(Field tag) = (Field value)" pairs. The 'tag' of each FIX field determines the field name and its associated data type that defines and constrains the allowable values for the field.

Each Tag/Value pair is separated by the FIX Delimiter character 0x01. In the sample messages provided, this delimiter is represented by the pipe character '|', e.g.

```
8=FIXT.1.1|9=376|35=AE|49=CHIX|56=ASX-CSP|34=80238|1128=9|52=20240214-
00:56:27.916|
```

The order of the tags must have the first tag as *BeginString* (8), the second tag as *BodyLength* (9), the third tag as *MsgType* (35), and the final tag must be *CheckSum* (10).

In this document, messages are indicated in **bold** whereas tags are indicated in *italic*, e.g. **Logon** (A), *TargetCompID* (56).

3.2 Definition of Required Column Values

Values	Comment
M	• Mandatory field • This field is required for this message type
O	• Optional field • Optional fields not specified in this ASX Clear FIX Specification will be ignored if sentOptional fields specified in this ASX Clear FIX Specification will be validated
C	• Conditional field • This field is dependent on the field / data type

3.3 Data Types

Data Type Name	Description
String	• Alpha-numeric free format strings, can include any character or punctuation except the delimiter (0x01), carriage return (0x0D) and linefeed (0x0A) characters • All char fields are case sensitive (i.e. morstatt != Morstatt)
Float	• Sequence of digits with optional decimal point and sign character (ASCII characters "-", "0" - "9" and "."); the absence of the decimal point within the string will be interpreted as the float representation of an integer value • All float fields must accommodate up to fifteen significant digits. The number of decimal places used should be a factor of business/market needs and mutual agreement between counterparties. Note that float values may contain leading zeros (e.g. "00023.23" = "23.23") and may contain or omit trailing zeros after the decimal point (e.g. "23.0" = "23.0000" = "23" = "23.") • Note that fields which are derived from float may contain negative values unless explicitly specified otherwise

Data Type Name	Description
int	• Sequence of digits without commas or decimals and optional sign character (ASCII characters "-" and "0" - "9"). The sign character utilizes one byte (i.e. positive int is "99999" while negative int is "-99999") • Note that int values may contain leading zeros (e.g. “00023” = “23”). Examples: 723 in field 21 would be mapped int as |21=723|. -723 in field 12 would be mapped int as |12=-723|
Boolean	• Char field containing one of two values: 'Y' = True/Yes 'N' = False/No
Length	• Representing the length in bytes • Value must be positive
SeqNum	• Representing a message sequence number • Value must be positive

UTCTimestamp	• Representing time/date combination represented in UTC (Universal Time Coordinated, also known as “GMT”) in either YYYYMMDD-HH:MM:SS (whole seconds) or YYYYMMDD-HH:MM:SS.sss (milliseconds) format, colons, dash, and period required
	• Valid values:
	  ○ YYYY = 0000-9999, MM = 01-12, DD = 01-31, HH = 00-23, MM = 00-59, SS = 00-60 (60 only if UTC leap second) (without milliseconds)
	  ○ YYYY = 0000-9999, MM = 01-12, DD = 01-31, HH = 00-23, MM = 00-59, SS = 00-60 (60 only if UTC leap second), sss=000-999 (indicating milliseconds)
	  ○ Leap seconds: Note that UTC includes corrections for leap seconds, which are inserted to account for slowing of the rotation of the Earth
	• Leap second insertion is declared by the International Earth Rotation Service (IERS) and has, since 1972, only occurred on the night of Dec 31 or Jun 30
	• The IERS considers March 31 and September 30 as secondary dates for leap second insertion, but has never utilized these dates
	• During a leap second insertion, a UTCTimestamp field may read "19981231-23:59:59", "19981231-23:59:60", "19990101-00:00:00" (see https://maia.usno.navy.mil/information/what-is-a-leap-second)

LocalMktDate	• Representing a Date of Local Market (vs. UTC) in YYYYMMDD format • This is the “normal” date field used by the FIX protocol • Valid values: YYYY = 0000-9999, MM = 01-12, DD = 01-31
Char	• Char value, can include any alphanumeric character or punctuation except the delimiter • All char fields are case sensitive (i.e. m != M)
Amt	• Value typically representing the Price multiplied by Qty
Price	• Value representing a price • Note the number of decimal places may vary • For certain asset classes prices may be negative values (for example, prices for options strategies can be negative under certain market conditions)
Qty	• Value capable of storing either a whole number (no decimal places) of “shares” (securities denominated in whole units) or a decimal value containing decimal places for non-share quantity asset classes (securities denominated in fractional units)
Currency	• Representing a currency type using ISO 4217 Currency code (3 character) values
Exchange	• Representing a market or exchange - ISO 10383 Market Identifier Code (MIC)
NumInGroup	• Value that represents the number of repeating values in a group

Length: when there is a limit in the length, a value is provided in the FIX Message Specification.

3.4 Handling of Unsupported Messages and Tags

- Any message that is not listed in this specification will be rejected with a **BusinessMessageReject** (j)
- When a message is rejected at a session level, a **Reject** (3) message will be sent from the FIX Gateway
- Unless otherwise specified, if a message listed in this specification is received with a tag that is not in this specification, then the message will be rejected with a **Reject** (3) message
- Fields or tags that are indicated as "Mandatory" or "Conditional" under the Req column, can lead to a rejection if not provided
- All fields or tags can lead to a rejection if the format is different from the format required as indicated in this specifications document or by the standard FIX version 5.0 SP2
- Fields with a null value are considered as not provided.

3.5 FIX Communication Workflow

The following message flow diagram describes the processing steps for FIX messaging in terms of:

1. **Logon** (A) Requests and Acknowledgements
2. **SecurityDefinition** (d)
3. **TradeCaptureReport** (AE)
4. Session Level rejections
    - **Reject** (3)
    - **Business Message Rejections** (j)
5. Business Message Validation
    - **TradeCaptureReportAck** (AR) – pending (optional)
    - **TradeCaptureReportAck** (AR) – approval
    - **TradeCaptureReportAck (**AR) – cancel
    - **TradeCaptureReportAck** (AR) – rejection
6. **Logout** (5) Requests and Acknowledgements

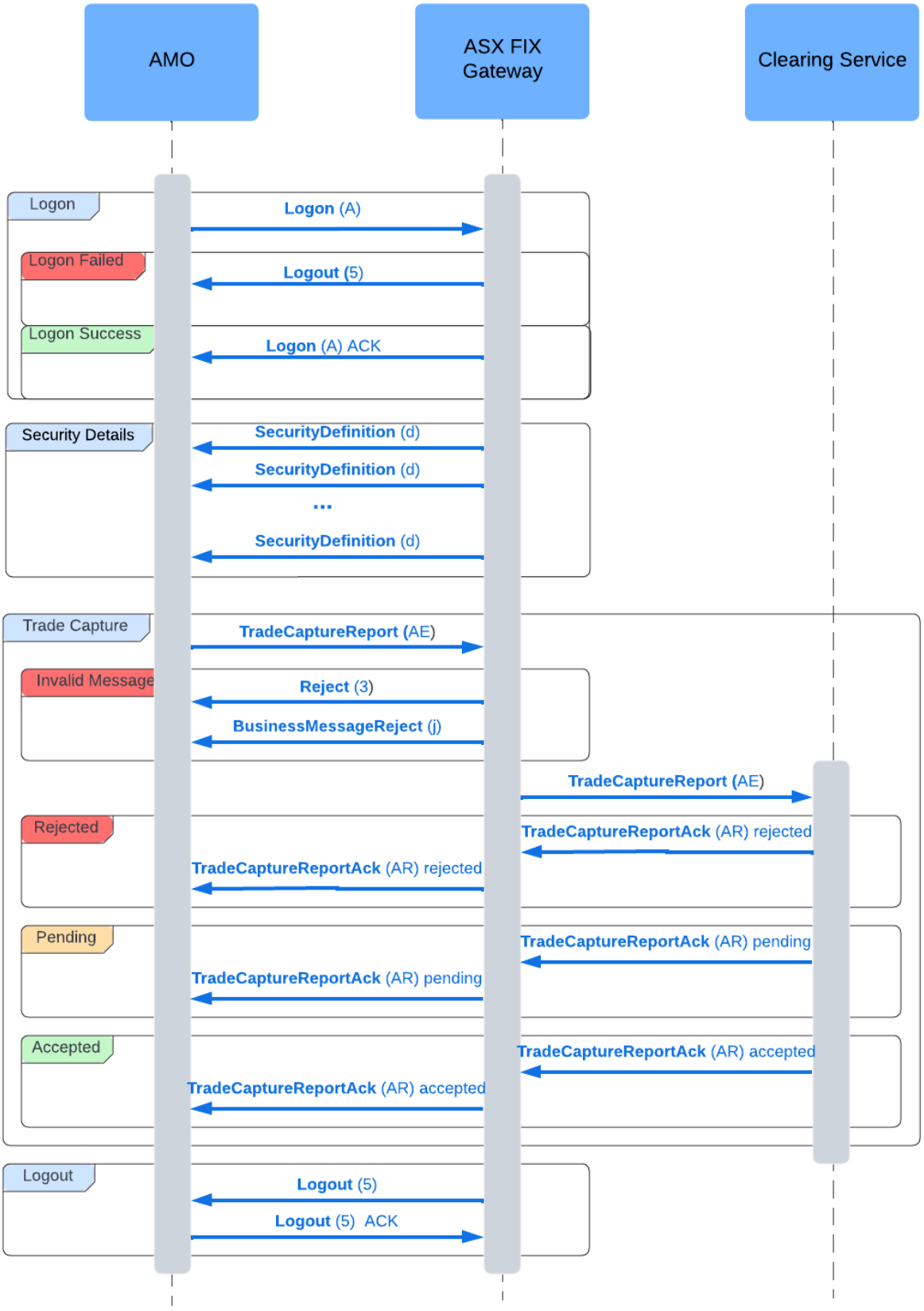


Fig 3.5.1 Communication Workflow

3.6 Sequence Number Gap

The FIX session protocol uses message sequence numbers to maintain an orderly exchange of messages between ASX Clear and AMOs.

ASX Clear and AMOs will each maintain their own set of inbound and outbound message sequence numbers.

Sequence numbers should reset at the start of the day and increment throughout the session. Any message sent by one side of a FIX session will increment the sequence number. A FIX session will not continue to the next trading day so both sides are expected to initialise the sequence numbers at the start of each trading day.

A gap between the message sequence numbers may indicate the possibility of missed messages which usually implies the need for resynchronisation of message sequence numbers between ASX Clear and the AMO. Re-synchronisation needs to be completed to ensure an orderly state of information in each party's systems and ASX Clear during day-to-day runs, as well as in cases of recovery after disruptive incidents.

3.6.1 Upon a Logon Request validating the *MsgSeqNum* (34) from the header

The FIX Gateway should validate the values of *MsgSeqNum* (34) from the header when receiving a **Logon** (A) request. This validation is necessary to identify any sequence number gaps and enable the AMOs and ASX Clear FIX Gateway to initiate the process of recovering lost messages.

When the ASX Clear FIX Gateway receives a **Logon** (A) request, it is responsible for identifying any sequence number gaps in the messages exchanged between the AMO and ASX Clear. To accomplish this, the FIX Gateway will utilize the *MsgSeqNum* (34) field from the header.

- If the *MsgSeqNum* (34) value is as expected, the transmission will be normal
- If the *MsgSeqNum* (34) value is lower than expected, the ASX Clear FIX Gateway closes the session with a **Logout** (5) message
- If the *MsgSeqNum* (34) value is higher than expected, then Message Recovery Process must be performed under the following conditions:
    - ASX Clear will acknowledge the logon request and respond with a **Logon** (A) Acknowledge
    - ASX Clear will send a **ResendRequest** (2) message to request for the missing messages
    - AMO will resend the missing messages.

Message and Behaviours

The table represents the scenarios where values are received when a gap is detected upon **Logon** (A) and the expected behaviours.

(Note: This is from ASX Clear's perspective.)

#	MsgSeqNum (34)	Behaviour
1	Value received is expected	• AMO will send **Logon** (A) • Normal message transmission
2	Value received is lower than expected	• AMO will send **Logon** (A) • ASX Clear will send **Logout** (5) with *SessionStatus* (1409) = 9 (sequence number too low)
3	Value received is higher than expected	• AMO will send **Logon** (A) • ASX Clear will acknowledge the logon request and respond with a **Logon** (A) Acknowledge • ASX Clear will send a **ResendRequest** (2) message to request for the missing messages • AMO will resend the missing messages

3.6.2 Upon a Logon Request using the tag NextExpectedMsgSeqNum (789)

The FIX Gateway should validate the values of *MsgSeqNum* (34) from the header and the *NextExpectedMsgSeqNum* (789) fields when receiving a **Logon** (A) request. This validation is necessary to identify any sequence number gaps and enable the AMOs and ASX Clear FIX Gateway to initiate the process of recovering lost messages.

When ASX Clear FIX Gateway receives a **Logon** (A) request, it is responsible for identifying any sequence number gaps in the messages exchanged between the AMO and ASX Clear. To accomplish this, the FIX Gateway will utilize the *MsgSeqNum* (34) from the header and *NextExpectedMsgSeqNum* (789) fields.

- The **Logon** (A) must contain the *NextExpectedMsgSeqNum* (789) field to indicate the sequence number that was last received from ASX Clear incremented by one (+1). This must be set to 1 for the first logon of the day
    - If the *NextExpectedMsgSeqNum* (789) value is as expected, the transmission will be normal
    - If the *NextExpectedMsgSeqNum* (789) value is higher than expected, ASX Clear closes the session with a **Logout** (5) message
    - If the *NextExpectedMsgSeqNum* (789) value is lower than expected, the **Logon** (A) request sent by the AMO proceeds and any sequence number gap is automatically filled before receiving any messages with sequences above the gap

The session initiator must wait until the **Logon** (A) acknowledgement from ASX Clear is received before sending other messages. Once the **Logon** (A) acknowledgement is received, the session initiator must validate that the *MsgSeqNum* (34) and *NextExpectedMsgSeqNum (789*) does not represent a gap

Message and Behaviours

The table represents the scenarios where values are received when a gap is detected upon **Logon** (A) and the expected behaviours.

(Note: This is from ASX Clear's perspective.)

Scenario #	MsgSeqNum (Tag 34)	NextExpectedMsgSeqNum (Tag 789)	Behaviour
1	Value received is expected	Value received is expected	AMO will send **Logon** (A) ASX Clear will acknowledge the logon request and respond with a **Logon** (A) ACK Normal message transmission
2	Value received is expected	Value received is lower than expected	AMO will send **Logon** (A) ASX Clear will acknowledge the logon request and respond with a **Logon** (A) **ACK** The outgoing *MsgSeqNum* (34) value will be higher than the AMO's expected value, however ASX Clear will resend the missing messages automatically Other message transmissions will pause until retransmissions are completed
3	Value received is expected	Value received is higher than expected	AMO will send **Logon** (A) ASX Clear will send Logout (5)
4	Value received is higher than expected	Value received is expected	AMO will send **Logon** (A) ASX Clear will acknowledge the logon request and respond with a **Logon** (A) ACK ASX Clear will send a **ResendRequest** (2) message to request for the missing messages AMO will resend the missing messages
5	Value received is higher than expected	Value received is lower than expected	AMO will send **Logon** (A)

Scenario #	MsgSeqNum (Tag 34)	NextExpectedMsgSeqNum (Tag 789)	Behaviour
			ASX Clear will send a *ResendRequest* (2) message to request for the missing messages (*MsgSeqNum* (34) value higher) After retransmission is complete, ASX Clear will proceed with normal transmission but once ASX Clear sends *MsgSeqNum* (34) = value higher than AMO's expected *MsgSeqNum* (34), ASX Clear will resend the missing messages automatically (*NextExpectedMsgSeqNum* (789) = value lower) Other message transmissions will pause until retransmissions are completed
6	Value received is higher than expected	Value received is higher than expected	AMO will send **Logon** (A) ASX Clear will send a **Logout** (5) (*NextExpectedMsgSeqNum* (789) higher than expected)
7	Value received is lower than expected	Value received is expected	AMO will send **Logon** (A) ASX Clear will send **Logout** (5) with *SessionStatus* (1409) = 9 (sequence number too low) *LastMsgSeqNumProcessed* (369) in the **Logout** (5) will indicate the last message sequence number received from the AMO AMO can use this number to modify its *MsgSeqNum* (34) value in the next **Logon** (A) message to re-establish connection
8	Value received is lower than expected	Value received is lower than expected	AMO will send **Logon** (A) ASX Clear will send **Logout** (5) *(MsgSeqNum* (34) value lower) with *SessionStatus* (1409) = 9 (sequence number too low) *LastMsgSeqNumProcessed* (369) in the **Logout** (5) will indicate the last message sequence number received from the AMO. AMO can use this number to modify its *MsgSeqNum* (34) value in the next **Logon** (A) message to re-establish connection When the **Logon** (A) message *Tag 789* value is lower than expected, ASX Clear will resend the missing messages automatically
9	Value received is lower than expected	Value received is higher than expected	AMO will send **Logon** (A) ASX Clear will send **Logout** (5) with *SessionStatus (Tag 1409)* = 9 (sequence number too low)

3.6.3 During the Trade Capture Session

During the Trade Capture session, the *MsgSeqNum* (34) in the message header must be checked to determine if the value provided in this tag aligns to the expected value. If the value does not align, then this indicates that there is a gap. When the ASX Clear FIX Gateway receives a **TradeCaptureReport** (AE) request, it is responsible for identifying any sequence number gaps in the messages exchanged between the AMO and ASX Clear. To accomplish this, the FIX Gateway will utilize the *MsgSeqNum* (34) field from the header.

If the *MsgSeqNum* (34) value is as expected, the transmission will be normal

If the *MsgSeqNum* (34) value is lower than expected, the ASX Clear FIX Gateway closes the session with a *Logout* (5) message

If the MsgSeqNum (34) value is higher than expected, then Message Recovery Process must be performed

- ASX Clear will send a **ResendRequest** (2) message to request for the missing messages
- AMO will resend the missing messages based on the values provided in the tags *BeginSeqNo* (7) and *EndSeqNo* (18)

Message and Behaviours

The table represents the scenarios where values are received when a gap is detected upon Trade Capture and the expected behaviours.
(Note: This is from ASX Clear's perspective.)

#	MsgSeqNum (34)	Behaviour
1	Value received is expected	AMO will send **TradeCaptureReport** (AE) Normal message transmission
2	Value received is lower than expected	AMO will send **TradeCaptureReport** (AE) ASX Clear will send **Logout** (5) with *SessionStatus* (1409) = 9 (sequence number too low)
3	Value received is higher than expected	AMO will send **TradeCaptureReport** (AE) ASX Clear will send a *ResendRequest* (2) message to request for the missing messages AMO will resend the missing messages

4 FIX Technical Connectivity (New)

4.1 FIX Sessions

AMOs will be able to establish one or more FIX sessions, each identified by a unique *SenderCompID* (49) allocated by ASX Clear.

4.2 FIX session scheduler events

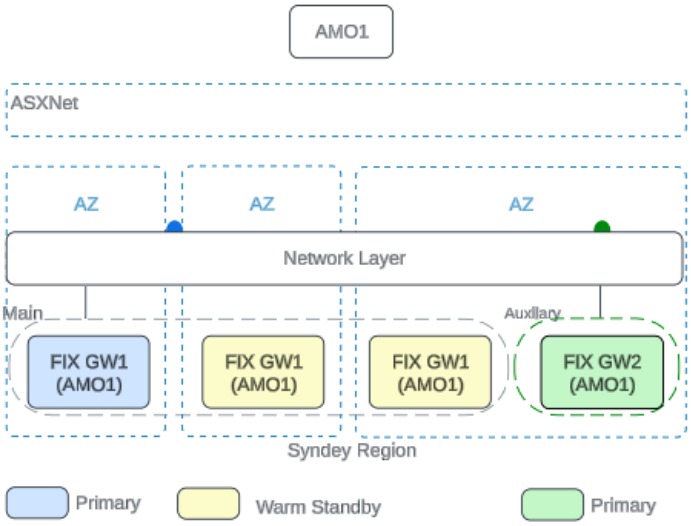
Event	Time	Description
Initiation of FIX session	07:00am AEST	• AMOs will be able to establish a FIX session with sequence numbers starting from 1, for each FIX Gateway • Where configured for the CompID of the FIX Session, a batch **SecurityDefinition** (d) messages will be disseminated
End of Trade Capture (cut-off time)	07:00pm AEST	• Trades and price messages received after the cut-off time will be rejected (received time and not the transaction time) • Post the cut-off time, any incoming messages from AMO will be rejected by the FIX Gateway • The **BusinessMessageReject** (j) message will be sent to AMO with the *business rejection reason* (380) set to 0 (other), and the *text* (58) set to "FIX message not accepted after cut-off time
Pending Messages Clearance	07:05pm AEST	• If there are remaining messages in a pending status, ASX Clear will reject these messages through a **TradeCaptureReportAck** (AR) rejection
End FIX Session		• All trade and price messages received before the cut-off time will be processed by ASX Clear and corresponding responses will be sent to AMOs • Subsequently, the FIX Gateway will initiate a session logout by sending a **Logout** (5) message

4.3 CHESS FIX Gateway Failover (New)

The ASX Clear FIX Gateway implementation is based upon a fault-tolerant design that ensures message transmissions will resume as normal in the event of a hardware, network or software component outage.

Each AMO will be provided with two end-points:

Main Endpoint (blue)
Auxiliary Endpoint (green)



Characteristics of the Main Endpoint:

- The Main Endpoint is backed by the ASX FIX Gateway Group 1 that is deployed across 3 Amazon Web Service (AWS) availability zones in the Sydney region
- The ASX FIX Gateway Group 1 will have one Primary instance and one or more Warm Standby instances
- The Primary instance will accept connections whilst the Warm Standby instance(s) will not, until and only when, it becomes Primary
- The promotion from Standby to Primary is automated and will take under 60 seconds
- AMOs will have one or more pre-configured CompIDs (FIX sessions) in Group 1
- A reconnection of the FIX session from the primary instance to a warm standby will be transparent to the FIX client with no loss of FIX session state (e.g. message sequence numbers) or FIX messages

Characteristics of the Auxiliary Endpoint

- The Auxiliary Endpoint is backed by ASX FIX Gateway Group 2 that is deployed in one availability zone in the Sydney region
- ASX FIX Gateway Group 2 will only have one primary FIX Gateway instance
- AMOs will have one or more pre-configured CompIDs (FIX sessions) in Group 2

Note:

The CompIDs allocated by ASX Clear will be specific to a FIX Gateway Group. A CompID that is supported in one Group will not be supported in the other group. FIX Session information is not shared across Group 1 and Group 2.

It is recommended that AMOs use the Main Endpoint to submit trades and prices as it is designed with a higher availability Service Level Objective (SLO). In the case of component failure within the Group 1, the AMO will experience a connection disconnect and should retry connection as described in section *4.4.2 Recovery on Disconnect (New)* to resume FIX Session (recovery).

AMO's may switch to the Auxiliary Endpoint at their own discretion e.g. for a Main Endpoint disconnect or failure. When switching to the Auxiliary Endpoint, it is important to recognise that the FIX Session context from the Main Endpoint is lost, and a new FIX session is established on the Auxiliary Endpoint with a different FIX CompID.

If an AMO switches from the Main Endpoint to the Auxiliary Endpoint, it is the AMO's responsibility to ensure any messages which were not acknowledged on the terminated Main Endpoint FIX session have been received by the FIX Server. This requires that all unacknowledged messages are resent on the new FIX Session on the Auxiliary Endpoint.

Resent trade messages will not be processed if they are already registered.

Any pending trades originally sent on the Main Endpoint session will not receive AR messages on the Auxiliary Endpoint session, as these will be routed to the CompID with which the trade was first registered.

4.4 FIX Client Session Recovery Process (New)

4.4.1 Lost Connection

As per the standard FIX session protocol, Heartbeat messages are used to validate FIX connections remain active. ASX requires Heartbeat intervals to be values between 10 and 60 seconds (as specified in the **HeartBtInt** (108) tag of the **Logon** (A) message).

If there are no business messages (e.g. **TradeCaptureReport** (AE) messages) to send within the heartbeat interval, a Heartbeat message is sent to keep the FIX connection active. ASX recommends a period of 30 seconds for both incoming and outgoing messages to signify an active connection.

When either side of the connection has not received any messages for a period equal to (*HeartBtInt* + "a reasonable transmission time in seconds") a **TestRequest** (1) message will be transmitted. If there is no response to the **TestRequest** (1) message, the sender of the **TestRequest** (1) will disconnect by sending a **Logout** (5) message.

4.4.2 Recovery on Disconnect (New)

In the case of disconnections, the AMO should retry connecting to the same end point at 10 second intervals up to 10 times. Should this fail, AMOs are asked to contact the ASX Operations team.

Once successful connection is established, the AMO should send a **Logon** (A) message to re-establish the FIX Session.

If the *MsgSeqNum* (34) in the **Logon** (A) message header matches the expected AMO outbound sequence number (being the ASX's inbound sequence number), the AMO can continue sending and receiving messages from where it left off.

If the *MsgSeqNum* (34) does not match the expected sequence number, refer to section 3.6 "Sequence Number Gap" to identify lost messages and manage message sequence number gaps.

4.5 Onboarding (New)

As part of the onboarding process, ASX will share an initial FIX username and password credential. ASX recommends that AMOs change their password when they first connect.

ASX will also provide AMOs with a self-service client-side portal to manage their Transport Layer Security (TLS) certificates. When requesting a client-side certificate AMOs must provide a list of IPs to whitelist.

4.6 Encryption (New)

Mutual TLS Encryption

The ASX FIX channel will use mutual TLS encryption. AMOs will be required to provide a valid TLS certificate to initiate a connection to their endpoint.

When an AMO initiates a Mutual TLS connection to an ASX endpoint, ASX will validate the TLS certificate and incoming IP addresses to allow the connection to the ASX FIX Gateway. If either validation fails, the connection will be rejected. If the problem persists AMOs should troubleshoot the connection with ASX Customer Support team.

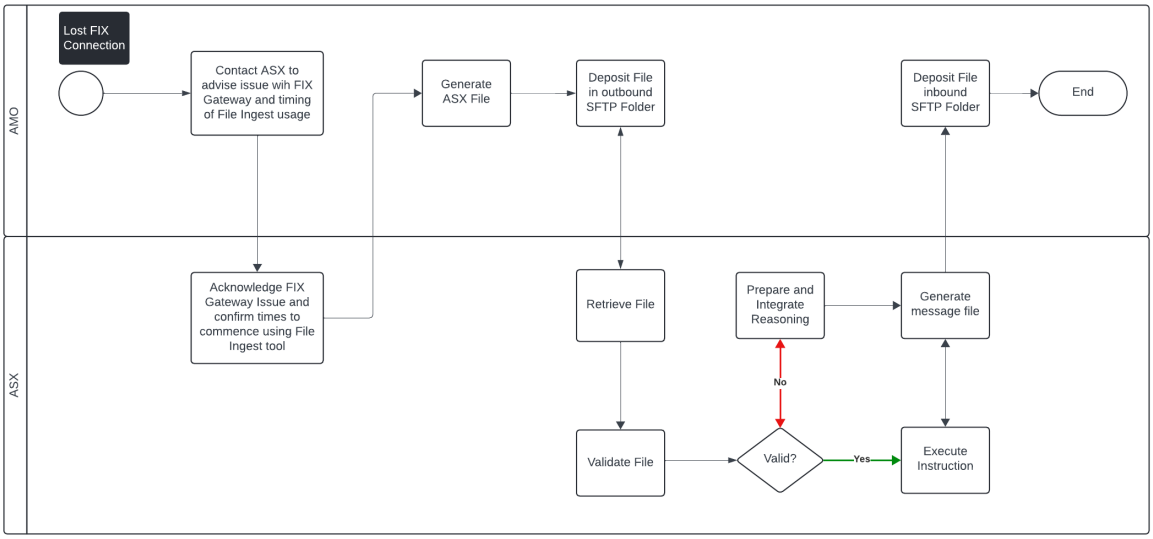
After a successful network connection, standard authentication will follow with FIX username and password credentials.

4.7 Alternate File Interface (BCP) (New)

The ASX provides an alternate file interface to support AMO's in the event of connectivity failures to the ASX FIX Gateway. The alternate file interface is designed to provide the following:

- AMOs can submit files containing transaction batches to the Clearing Service via a secure file transfer protocol (sFTP) connection
- Each file must contain transactions formatted as FIX messages (One FIX message per line)
- Messages must be the same format as was submitted via the ASX FIX Gateway
- AMO can submit instructions for the Clearing Service to:
  - Process and execute new or cancelled market trades
  - Process and update new security price data
  - Process “Accept”, “Reject” or “Pend” transactions, similar to those messages submitted via ASX FIX Gateway
- The tool will generate a response file containing the FIX response messages, i.e. **'TradeCaptureReportAck** (AR)' and ‘**PriceConfirmationAck** (AU)’
- AMO will be able to retrieve the response file via the sFTP connection

BCP Process Flow



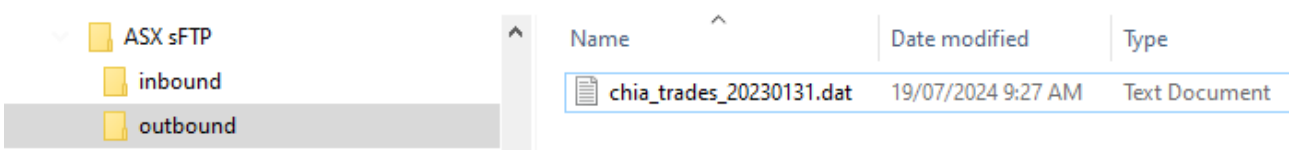
4.7.1 Alternate File Ingestion Interface sFTP Folder Structures

AMOs will be provided a unique, secure folder structure held by the ASX. These folders will contain ‘outbound’ and ‘inbound’ subdirectories.

AMOs will be able to place files containing transactions into their ‘outbound’ folder for processing by the ASX Clear Service.

AMOs will retrieve files containing transaction responses from their ‘inbound’ folder once the file from the ‘outbound’ folder has been processed by the Clearing Service.

Example SFTP Trade Ingestion Folder Structure



4.7.2 Interface File Format

File naming convention:

File names should be in lowercase
File names should start with "z_trades_" prefix where 'z' is the Operating MIC of the Market Operator:
MIC Prefix Definitions (must be lower case in the interface file name):

- xasx – ASX Trade
- chia – CBOE
- xnec – National Stock Exchange

Must end with "yyyymmdd_hhmm.dat", where:

- yyyymmdd = current date for which trades are uploaded where this part of the filename remains the same if there are multiple files to be processed in a single day
- hhmm = current hour and minute of the trade file upload where this part of the file name will be different if there are multiple files to be processed intra-day (hhmm is to be entered in 24-hour format e.g. 9:30PM would be entered as 2130)
- .dat as the file extension

Note:

The outbound (AMO to ASX interface) file can:

Handle multiple messages within the one file (bulked)
Comprise of multiple message types i.e. pricing updates, new trades and trade cancellations within the one interface file but must be formatted in the defined FIX messaging standards
In the instance of 'cancelled trades', the original trade must already exist or be listed before the cancelled trade sharing the same SecondaryTradeID (i.e. to cancel a trade, the original trade must already be registered or listed before the trade cancellation for it to be cancelled)

Example File Name: chia_trades_202401312130.dat

4.7.3 Acknowledgement File Name Format (Received from ASX)

File naming convention:

- File name should be in lowercase
- File prefix will start with "z_ack_" where 'z' is the Operating MIC of the Market Operator (displayed in the MIC prefix definitions above)
- File name should end with "yyyymmdd_hhmm.dat" where "yyyymmdd_hhmm" must match yyyymmdd_hhmm part of the Market Operator's trade filename
  Note: hhmm to be entered in 24-hour format e.g. 9:30PM would be entered as 2130
- .dat as the file extension

Note:

The inbound (ASX to AMO interface) acknowledgement file can, depending on the file submitted to the ASX, contain multiple message responses within the one file.

Example File Name: xnec_ack_202401312130.dat

4.7.4 Messages Accepted via BCP File Ingestion

Allowed Interface Messages	Outbound (AMO) File Message	Inbound (AMO) File Message
New Trades – Accepted	Successful bulk new trades can be submitted in the **TradeCaptureReport** (AE) messaging format	All new market trades successfully registered in the Clearing Service in the

Allowed Interface Messages	Outbound (AMO) File Message	Inbound (AMO) File Message
		format of the **TradeCaptureReportAck** (AR) FIX message
Trade Cancellations - Accepted	Bulk trade cancellations can be submitted in the **TradeCaptureReport** (AE) format	Registered trades successfully cancelled in the Clearing Service in the format of the **TradeCaptureReportAck** (AR) FIX message
Pricing Update – Accepted	Price updates can be submitted to the ASX in the **MarketDataSnapshotFullRefresh** (W) format	All accepted price updates successfully registered in the Clearing Service in the format of the **ConfirmationAck** (AU) message
Trade Cancellations – Rejected	Bulk trade cancellations can be submitted in the **TradeCaptureReport** (AE) format	All failed cancellations will be submitted in the format of the **TradeCaptureReportAck** (AR) FIX message including the reject reason
New Trades - Pending	New bulk trades can be submitted via the **TradeCaptureReport** (AE) format	All pending market trades in the format of the **TradeCaptureReportPEND** (AR) FIX message including the pending reason
New Trades – Schema Validation Rejected	New bulk trades can be submitted via the **TradeCaptureReport** (AE) format	All schema rejections will be submitted in the format of the **Reject** (3) message within the inbound file submitted to the AMO
New Trades – Business Validation Rejected	New bulk trades can be submitted in the **TradeCaptureReport** (AE) format	All business invalid session messages in the format of the **BusinessMessageReject** (j)
Pricing Updates - Rejected	Price updates can be submitted to the ASX in the **MarketDataSnapshotFullRefresh** (W) format	All failed price updates registered in the Clearing Service will be submitted in the format of the **ConfirmationAck** (AU) FIX message

4.7.5 Delivery Timing

It is expected that the interface file is received by the ASX during the Trade Capture session (or the end time of Trade Capture Session if it is to be extended for that business day).

AMOs should contact ASX Technical Support prior to utilising the BCP file ingest tool as the process requires upfront co-ordination.

5 Rejection Scenarios

The rejection levels are:

FIX session level rejection: related to the way an AMO's FIX session is communicating with the ASX Clear FIX Gateway
FIX application-level rejection (business rejection): related to the data contained in the requests sent by the AMO's FIX session

5.1 Rejection or disconnection messages from the ASX FIX Gateway Scenarios

Rejection Scenario	ASX FIX Gateway Behaviour
AMO sends a FIX message and *MarketID* (1301) "Market Identifier Code (MIC)" is invalid	ASX Clear sends a **BusinessMessageReject** (j) message with *BusinessRejectReason* (380) = 0 (other)
AMO sends a FIX message after cut-off time	ASX Clear sends a **BusinessMessageReject** (j) message with *BusinessRejectReason* (380) = 0 (other) with Text (58) set to "FIX Message not accepted after cut-off time"
AMO sends a message type that is unsupported by the ASX FIX Gateway	ASX Clear sends a **BusinessMessageReject** (j) message with *BusinessRejectReason* (380) = 3 (unsupported message type)
AMO sends a message and conditional field is missing	ASX Clear sends a **BusinessMessageReject** (j) message with *BusinessRejectReason* (380) = 5 (Conditionally Required Field Missing)
AMO sends a message that contains an invalid tag number	ASX Clear sends a **Reject** (3) with *SessionRejectReason* (373) = 0 (invalid tag number)
AMO sends a message where a mandatory tag or an applicable conditional tag is missing	ASX Clear sends a **Reject** (3) with *SessionRejectReason* (373) = 1 (required tag missing)
AMO sends a message where a tag is not defined for the message type	ASX Clear sends a **Reject** (3) with *SessionRejectReason* (373) = 2 (tag not defined for this message type)
AMO sends a message where a tag specified is missing a value	ASX Clear sends a **Reject** (3) with *SessionRejectReason* (373) = 4 (tag specified without a value)
AMO sends a message where a value is out of range	ASX Clear sends a **Reject** (3) with *SessionRejectReason* (373) = 5 (tag specified without a value)
AMO sends a message where the data format (syntax) is incorrect	ASX Clear sends a **Reject** (3) with *SessionRejectReason* (373) = 6(Incorrect data format for value)
AMO sends a **ResendRequest** (2) message with *BeginSeqNo* (7) or *EndSeqNo* (16) = a number that is out of range	ASX Clear sends a **Reject (**3) with *SessionRejectReason* (373) = 5 (value is incorrect (out of range) for this tag)

Rejection Scenario	ASX FIX Gateway Behaviour
AMO sends a **ResendRequest** (2) message with *BeginSeqNo* (7) or *EndSeqNo* (16) is an invalid format such as an alphabet (e.g. "a")	ASX Clear first sends a **Reject** (3) with *SessionRejectReason* (373) = 6 (incorrect data format for value)
AMO sends a FIX message and the SenderCompID (49) and *TargetCompID* (56) values received did not match values expected	ASX Clear sends a **Reject** (3) with *SessionRejectReason* (373) = 9 (CompID problem)
AMO sends a **ResendRequest** (2) message with *PossDupFlag* (43) = Y, *OrigSendingTime* (122) is greater than *SendingTime* (52), and *MsgSeqNum* (34) is as expected	ASX Clear sends a **Reject** (3) with *SessionRejectReason* (373) = 10 (*SendingTime* (52) accuracy problem)
AMO sends a FIX message and the value of MsgType (35) is not valid (not defined in the message specification)	ASX Clear sends a **Reject** (3) with *SessionRejectReason* (373) = 11 (invalid MsgType)
AMO sends a FIX message and the tag number is duplicated	ASX Clear sends a **Reject** (3) with *SessionRejectReason* (373) = 13 (tag appears more than once)
AMO sends a FIX message with a message header with *BeginString* (8), *BodyLength* (9), and *MsgType* (35) are not in its respective order at the start of the message and/or a message trailer without tag 10 as the last tag in the message	ASX Clear sends a **Reject** (3) with *SessionRejectReason* (373) = 14 (tag specified out of required order)
AMO sends a FIX message with repeating groups in which the order of repeating group fields does not match the specification	ASX Clear sends a **Reject** (3) with *SessionRejectReason* (373) = 15 (Repeating group fields out of order)
AMO sends a FIX message with repeating groups count field value for a repeating group is incorrect	ASX Clear sends a **Reject** (3) with *SessionRejectReason* (373) = 16 (Incorrect NumInGroup count for repeating group)
AMO sends a **Logon** (A) message and a *NewPassword* (925) value that does not meet the password policy	ASX Clear sends a **Logout** (5) with *SessionStatus* (1409) = 3 (New session password does not comply with policy)
AMO sends a **Logon** (A) message with incorrect *SenderCompID* (49)	ASX Clear sends a **Logout** (5) with *SessionStatus* (1409) = 4 (session logout complete) with reason in *Text* (58) = "Incorrect Comp ID when Logon"
For the first **Logon** (A) message of the day, AMO sends a **Logon** (A) message with the *ResetSeqNumFlag* disabled (141=N)	ASX Clear sends a **Logout** (5) with *SessionStatus* (1409) = 4 (session logout complete) with reason in *Text* (58) as "Logout complete - Reset flag need to be enabled for the first logon request"
AMO sends a **Logon** (A) message with invalid username or password	ASX Clear sends a **Logout** (5) with *SessionStatus* (1409) = 5 (invalid username or password)
AMO sends a **Logon** (A) message with incorrect password for more than the maximum number of times allowed	ASX Clear sends a **Logout** (5) with *SessionStatus* (1409) = 6 (account locked)

Rejection Scenario	ASX FIX Gateway Behaviour
AMO sends a **Logon** (A) message with a sequence number that is lower than the expected sequence number	ASX Clear sends a **Logout** (5) with *SessionStatus* (1409) = 9 (sequence number too low) with reason in *Text* (58) = "Sequence number too low. Expected sequence number is <expected inbound sequence number>. Received <actual sequence number> instead"
AMO sends a **Logon** (A) message with a *HeartBtInt* (108) value that is not between **10 and 60** (seconds)	ASX Clear sends a **Logout** (5) with *SessionStatus* (1409) =104 (*HeartBtInt* (108) must be between 10 and 60 (seconds)
AMO sends **Logon** (A) message that could not be recognised (containing missing or invalid values)	ASX Clear sends a **Logout** (5) with *SessionStatus* (1409) = 106 (**Logon** (A) message could not be recognised) with reason in *Text* (58) = "Logon message invalid"

5.2 TradeCaptureReportAck (AR) Pending New Messages

The AMO has the choice to opt-in or opt-out for receiving **TradeCaptureReportAck** (AR) for pending trades.

The table below outlines the error codes and associated text descriptions that can be provided in the **TradeCaptureReportAck** (AR) message where a trade is accepted as a pending new trade. Trades may be captured and held in a pending state for further manual correction. If Pending New trades are resubmitted for capture processing in ASX Clear subsequent **TradeCaptureReportAck** (AR) messages will be sent indicating the outcome of the reprocessing:

Conditions:

- *TrdRptStatus* (939)=4 (Pending New)

Business Validation Failure	*Text* (58)
Buyer Trading Participant does not have an active Trading role in the Clearing Service	Pending – Buyer Trading Party is not in 'Active role' as Trading member
Trading Participant ID exists in the Clearing Service for buy side of the market trade but is not in an active state	Pending – Buyer Trading party is not in 'Active' status
Seller Trading Participant does not have an active Trading role in the Clearing Service	Pending – Seller Trading Party is not in 'Active' role as 'Trading member'
Trading Participant ID exists in the Clearing Service for sell side of the market trade but is not in an active state	Pending – Seller Trading party is not in 'Active' status
Trading Participant is not linked to an active Clearing Participant	Pending – Trading member and Clearing member association is not defined for the Buyer or Seller
Clearing Participant of the market trade is not linked to a CCP	Pending – CP not linked with CCP
Trade Date is not valid Clearing Service Business Date	Pending – Trade Date is not valid Clearing Service Business Date
Buyer Clearing Participant does not have an active Clearing role in the Clearing Service	Pending – Buyer Clearing Participant is not in 'Active role' as Clearing member
Buyer Clearing Participant is not in Active Status	Pending – Buyer Clearing Member is not in Active Status
Seller Clearing Participant does not have an active Clearing role in the Clearing Service	Pending – Seller Clearing Participant is not in 'Active role' as Clearing member
Seller Clearing Participant is not in 'Active' Status	Pending – Seller Clearing Member is not in 'Active' Status

5.3 TradeCaptureReportAck (AR) Rejection Messages

The table below outlines the error codes and associated text descriptions that can be provided in the **TradeCaptureReportAck** (AR) message where a trade is rejected:

Conditions:

- *TrdRptStatus* (939) = 1 (Rejected)
- TradeReportRejectReason (751) = 99 (Other)

Business Validation Failure	*RejectText* (1328)
Invalid Approved Market Operator	<TAGID><TAG NAME>: [SUPPLIED_VALUE] is invalid For example: <1301><MarketID>: [MAQU] is invalid <1040><SecondaryTradeId>: [SUPPLIED_VALUE] is invalid <1003><TradeId>: [SUPPLIED_VALUE] is invalid
Non-active Market Operator	<TAGID><TAGNAME>: [SUPPLIED_VALUE] is not active For example: <1301><MarketID>: [CHIA] is not active
Participant does not exist in Clearing Service (Execution Firm, Clearing Firm)	<TAGID><TAGNAME>: [SUPPLIED_VALUE] is invalid for <TAGID><TAGNAME>: [SUPPLIED_VALUE] For example: <448><PartyId>: [1234] is invalid for <452><PartyRole>: [1] <448><PartyId>: [12345] is invalid for <452><PartyRole>: [4]
Invalid Data Type	<TAGID><TAG NAME>: [SUPPLIED_VALUE] Data Type is invalid
Non-Allowable Value	<TAGID><TAG NAME>: [SUPPLIED_VALUE] is not an allowable value For example: PartyRole can only be '1' or '4' or '45' or '21' or '119' <452><PartyRole>: [2] is not an allowable value
Conditional Field not populated	<TAGID><TAGNAME>: is Mandatory when <TAGID><TAGNAME>: is present For example: The As At Date is a conditional field that must be populated when the AsOfIndicator is equal to 1 (True) <1125><OrigTradeDate> is Mandatory when <1015> <AsOfIndicator> =1

Business Validation Failure	*RejectText* (1328)
Invalid Conditional Field	<TAGID><TAGNAME>: [SUPPLIED VALUE] cannot be supplied when <TAGID><TAGNAME>: [SUPPLIED VALUE] For example: The As At Date is a conditional field and must not be populated when the AsOfIndicator is equal to 0 (False) <1125><OrigTradeDate>: [SUPPLIED VALUE] cannot be supplied when <1015><AsOfIndicator>: [0]
End of Day batch process to reject remaining pending trades	Rejected by system
AMO sends a Trade Capture Report (AE) to cancel a market data that was previously cancelled	<75><TradeDate>:[SUPPLIED_VALUE]<1040><SecondaryTradeID>:[SUPPLIED_VALUE] already cancelled
AMO sends and Trade Capture Report (AE)to cancel a market data that was previously rejected	<75><TradeDate>:[SUPPLIED_VALUE]<1040><SecondaryTradeID>:[SUPPLIED_VALUE] already rejected
AMO sends and Trade Capture Report (AE) to cancel with SecondaryTradeId does not exist	No trade found for <75><TradeDate>:[SUPPLIED_VALUE]<1040><SecondaryTradeID>:[SUPPLIED_VALUE]
AMO sends Trade Capture (AE) to cancel with data in the relevant tag do not match the value stored for the market trade in the Clearing Service.	<TAGID><TAGNAME>:[SUPPLIED_VALUE] is invalid

5.4 MarketDataSnapshotFullRefresh (W) Rejection Messages

The table below outlines the error codes and associated text descriptions that can be provided in the **ConfirmationAck** (AU) message where *AffirmStatus* (940)=2 (Confirmed Rejected):

ConfirmRejReason (774)	Business Validation Failure	Text (58)
99	Invalid Approved Market Operator	<TAGID><TAG NAME>: [SUPPLIED_VALUE] is invalid For example: <1301><MarketID>: [MAQU] is invalid
99	Non-active Market Operator	<TAGID><TAGNAME>: [SUPPLIED_VALUE] is not active For example: <1301><MarketID>: [CHIA] is not active
99	Invalid Data	<TAGID><TAGNAME>: [SUPPLIED_VALUE] is invalid For example: <75><TradeDate>: [02072018] is invalid
99	Invalid Data Type	<TAGID><TAG NAME>:[SUPPLIED_VALUE] Data Type is invalid
99	Non-Allowable Value	<TAGID><TAG NAME>:[SUPPLIED_VALUE] is not an allowable value For example: MDEntry (269) can only be 0, 1, 2, 4, 7, 8 or B <269><PartyRole>: [b] is not an allowable value
99	Where there is missing repeating block for a required pricing field	One or more required repeating group codes is missing from the tag <TAGID><TAG NAME> For example: One or more required repeating group codes is missing from the tag <269><MDEntry>
99	Where more than one repeating block is provided for an MDEntryType	Message contains more than one repeating group for [SUPPLIED VALUE] in <269><MDEntryType> For example: Message contains more than one repeating group for [B] in <269><MDEntryType>
99	Missing Tag in Repeating Block	<TAGID><TAGName> Required for <TAGID><TAGNAME>: [SUPPLIED VALUE] For example: <270><MDEntryPx> Required for <269><MDEntryType>: [4]
99	Missing currency	< TAGID><TAGName> is missing. For example:

ConfirmRejReason (774)	Business Validation Failure	Text (58)
		<15><Currency> is missing

6 Messages

6.1 Common message structures

6.1.1 Standard Header

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

- Forms the first part of every message
- Identifies the message
- Provides the information to correctly interpret a message

Field	Field Name	Req	Data Type	Description	Valid Values
8	BeginString	M	String	Identifies the beginning of a new message and protocol version It is always the first field in a message	FIXT.1.1
9	BodyLength	M	Length	Identifies the message length, in bytes, forward to the CheckSum field It is always the second field in a message	
35	MsgType	M	String	Identifies the message type It is always the third field in a message Please refer to individual messages for the value to be used	0 = Heartbeat 1 = TestRequest 2 = ResendRequest 3 = Reject 4 = SequenceReset 5 = Logout A = Logon AE = TradeCaptureReport AR = TradeCaptureReportAck j = BusinessMessageReject W = MarketDataSnapshotFullRefresh AU= Confirmation Ack
49	SenderCompID	M	String	Identifies the Sender of the message	

Field	Field Name	Req	Data Type	Description	Valid Values
56	TargetCompID	M	String	Identifies the Receiver of the message	
34	MsgSeqNum	M	SeqNum	Identifies the message's sequence number	
43	PossDupFlag	O	Boolean	Identifies re-transmitted messages	N - Original Transmission Y - Possible Duplicate
97	PossResend	O	Boolean	Identifies if the message contains information that has been sent under another sequence number	N - Original Transmission Y - Possible Resend
1128	ApplVerID	M	String	Identifies the FIX version used for the message	9 - FIX 5.0 SP2
52	SendingTime	M	UTCTimestamp	Identifies the message's transmission time in UTC	
122	OrigSendingTime	O	UTCTimestamp	Identifies the original message's transmission time in UTC This must be sent for messages sent as a result of a ResendRequest	
369	LastMsgSeqNumProcessed	O	SeqNum	The last MsgSeqNum (34) value received by the FIX engine and processed by downstream applications, such as trading engine or order routing system Can be specified on every message sent Useful for detecting a backlog with a counterparty Tag 369 can only be provided in the Logout (5)	

6.1.2 Standard Header Example

```
8=FIXT.1.1|9=376|35=AE|49=CHIX|56=ASX-CSP|34=80238|1128=9|52=20240214-00:56:27.916|
487=0|1003=Cf2T8nqno0|75=20240214|1015=0|64=20220517|60=20240214-
11:56:11.475|55=JPM|31=2.3|32=69|15=AUD|1301=CHIA|552=2|54=1|453=2|448=0001|447=C|452=1|448=20036|447=C|452=4|54=2|453=2|
448=0002|447=C|452=1|448=20037|447=C|452=4|10=195|
```

6.1.3 Standard Trailer

All FIX messages in this document contain a standard trailer, which is defined below. The Standard Trailer terminates every message.

Field	Field Name	Req	Data Type	Description	Valid Values
10	CheckSum	M	String	Identifies the three-byte checksum defined in the FIX 5.0 SP2 Specification It is always the last field in the message	

6.1.4 Standard Trailer Example

```
8=FIXT.1.1|9=376|35=AE|49=CHIX|56=ASX-CSP|34=80238|1128=9|52=20240214-
00:56:27.916|487=0|1003=Cf2T8nqno0|75=20240214|1015=0|64=20220517|60=20240214-
11:56:11.475|55=JPM|31=2.3|32=69|15=AUD|1301=CHIA|552=2|54=1|453=2|448=0001|447=C|452=1|448=20036|447=C|452=4|54=2|453=2|
448=0002|447=C|452=1|448=20037|447=C|452=4|10=195|
```

6.2 Fix Session Layer

6.2.1 Logon (A)

The Logon message authenticates a Participant connecting via FIX. The Participant is required to initiate the FIX session.
SenderCompID (49) and *TargetCompID* (56) are agreed when the service is set up. *HeartBtInt* (108) is recommended to be set to 30 seconds.

If the Logon message received from the Participant contains *ResetSeqNumFlag* (141)=Y the FIX session sequence numbers for both sides of the connection will be set to 1 and only messages generated for the client after the login message will be sent. If *ResetSeqNumFlag* (141)=N or if the tag is omitted in the logon message then any messages queued for the client will be sent immediately after login.

Upon successfully authenticating the user, ASX Clear will respond to the Participant with a logon message in response. If authentication fails ASX will issue a logout message. Circumstances when a **Logout** (5) message is not sent, include an invalid *SenderCompID* (49) or *TargetCompID* (56).

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

Field	Field Name	Req	Data Type	Description	Valid Values
	Standard Header	M	Header	MsgType = A	
98	EncryptMethod	M	Int	Method of encryption	0 = None / Other
108	HeartBtInt	M	Int	• Heartbeat interval in seconds • The same value is used by both sides • Defines the heartbeat interval in seconds • Values between 10 and 60 are accepted • ASX Clear recommends 30 seconds	Values between 10 and 60 are accepted
789	NextExpectedMsgSeqNum	O	SeqNum	• Next expected message sequence number (MsgSeqNum<34>) value to be received • This tag is optional for Clearing Service's inbound messages but mandatory for Clearing Service's outbound messages (Logon A acknowledge by Clearing Service)	
553	Username	M	String	FIX username This tag is not case sensitive	Inbound: must be the same as the *SenderCompID* (49) in the header

Field	Field Name	Req	Data Type	Description	Valid Values
				For incoming messages into ASX Clear, the Username (553) will be the same as the SenderCompID (49)	Outbound: “ASX Clear”
554	Password	C	String	Password for username AMO to supply when logging in Not returned in response from ASX Clear	
1137	DefaultApplVerID	M	String	The default version of FIX messages used in this session	9 = FIX 5.0 SP2
141	ResetSeqNumFlag	O	Boolean	Indicates that both sides of the FIX session should reset sequence numbers	N = No Y = Yes, reset sequence numbers
925	NewPassword	O	String	Specifies a new password for the FIX Logon The new password is used for subsequent logons	The new password must satisfy the below requirements: Minimum of 15 characters in length Must be unique from the previous 12 passwords used Must contain: At least one English upper case character (A-Z); At least one English lower case character (a-z); At least one number; and At least one of the following non-alphanumeric special characters: !@#$%^&*()_+|~-=\`{}[]:";'<>?,./)
1409	SessionStatus	O	Int	Identifies the status of the FIX session It is sent by ASX Clear	0 = session active 1 = session password changed
	Trailer	M	Trailer	Trailer	

6.2.2 Logon (A) Example

```
8=FIXT.1.1|9=111|35=A|49=ASX|56=ASX-CSP|34=1|52=20220524-
04:17:27.545|1128=9|98=0|108=30|141=N|789=1|553=ASX|554=ASX1234|1137=9|10=222|
```

6.2.3 Heartbeat (0)

Heartbeat messages are sent by counterparties to indicate that a connection is still active as well as a response to **TestRequest** (1) messages.

Each party will keep track of its previous message sent and when the interval reaches a period defined by the *HeartBtInt* (108) from its previous message sent (and there are no other messages to send), it will send a **Heartbeat** (0) message to the counterparty to indicate that its connection is still active.

When either side of the connection has not received any message from the counterparty for a period equal to (**HeartBtInt +** "some reasonable transmission time") seconds, a **TestRequest** (1) message will be transmitted. If there is no response to the **TestRequest** (1) message, it indicates a network issue or that the counterparty (either the AMO or ASX Clear FIX Gateway) is no longer available. In such cases, the sender of the **TestRequest** (1) will disconnect by sending a **Logout** (5).

Field	Field Name	Req	Data Type	Description	Valid Values
	Standard Header	M	Header	MsgType = 0	
112	TestReqID	C	String	Unique identifier included in TestRequest (1) message to be returned in resulting Heartbeat (0) message Required when the Heartbeat (0) is the result of a TestRequest (1) message	
	Standard Trailer	M	Trailer	Trailer	

6.2.4 Heartbeat (0) Example

```
8=FIXT.1.1|9=54|35=0|49=ASX-CSP|56=CHIX|34=9|52=20240214-23:30:34.395|10=085|
```


6.2.5 TestRequest (1)

The **TestRequest** (1) verify if a connection is active, a **TestRequest** (1) message is sent to the counterparty. The recipient of the **TestRequest** (1) responds with a **Heartbeat** (0) message. Failure to respond to a **TestRequest** (1) message may trigger a disconnection by the sender **Logout** (5).

Field	Field Name	Req	Data Type	Description	Valid Values
	Standard Header	M	Header	MsgType = 1	
112	TestReqID	M	String	The unique identifier of the Test Request message	
	Standard Trailer	M	Trailer	Trailer	

6.2.6 TestRequest (1) Example

```
8=FIXT.1.1|9=92|35=1|49=ASX|56=ASX-CSP|34=166|52=20220516-06:51:43.250|1128=9|112=20220516-06:51:43|10=249|
```

6.2.7 ResendRequest (2)

The **ResendRequest** (2) message is sent to request the retransmission of messages. This will pause the current flow of trade confirmation streams and will resume once the retransmission request is completed. When trade confirmation messages are lost or not consumed, the ASX Clear FIX Gateway will allow client FIX sessions to recover through a retransmission request.

Users have the option to perform the below message retransmissions:

Request for a retransmission of a specific range of messages – by sending a **ResendRequest** (2) message with *BeginSeqNo* (7) = sequence number of the first message in range to be resent and *EndSeqNo* (16) = sequence number of the last message in range to be resent. The FIX Gateway will pause the stream of messages and retransmit the FIX application messages that are from the *BeginSeqNo* (7) until the *EndSeqNo* (16) values.

Request for a retransmission of all messages sent starting from a specific message sequence number – by sending a *ResendRequest* (2) message with *BeginSeqNo* (7) = sequence number of the first message in range to be resent and *EndSeqNo* (16) = 0. The FIX Gateway will pause the stream of messages and retransmit all FIX application messages. The value "0" in the *EndSeqNo* (16) indicates a request for all messages from *BeginSeqNo* (7) until the last *MsgSeqNum* (34) that was sent.

During a retransmission, the FIX Gateway will resend the application messages [i.e., **TradeCaptureReport** (AE), **MarketDataSnapshotFullRefresh** (W)] and **SequenceReset** (4) message will be sent as a gap fill to replace the administration messages [i.e., **Logon** (A)**, Logout** (5)**, Heartbeat** (0)**, TestRequest** (1)**, ResendRequest** (2)**, Reject** (3)**, SequenceReset** (4)**, BusinessMessageReject** (j)]. All retransmitted messages will contain *PossDupFlag* (43) = Y

Field	Field Name	Req	Data Type	Description	Valid Values
	Standard Header	M	Header	MsgType = 2	
7	BeginSeqNo	M	SeqNum	The Message sequence number of first message in range to be resent	
16	EndSeqNo	M	SeqNum	The Message sequence number of last message in range to be resent If all messages subsequent to *BeginSeqNo* (7) are required set *EndSeqNo* (16)=0	
	Standard Trailer	M	Trailer	Trailer	

6.2.8 ResendRequest (2) Example

```
8=FIXT.1.1|9=69|35=2|49=ASX|56=ASX-CSP|34=8|52=20220524-04:27:59.764|1128=9|7=2|16=0|10=233|
```

6.2.9 Reject (3)

The **Reject** (3) message is sent when a message is received but cannot be processed due to a session-level rule violation.
For example, the receipt of a message that has successfully passed de-encryption, CheckSum and BodyLength checks but cannot be processed due to invalid basic data.

Field	Field Name	Req	Data Type	Description	Valid Values
	Standard Header	M	Header	MsgType = 3	
45	RefSeqNum	M	SeqNum	Identifies the Message Sequence Number (MsgSeqNum) of rejected message	
371	RefTagId	O	Int	The tag number of the FIX field being referenced in the session level error	
372	RefMsgType	O	String	Identifies the MsgType of the FIX message being referenced	AE W A 0 1 2 4 5
373	SessionRejectReason	O	Int	The code that identifies the reason for the rejection	0 - Invalid Tag Number 1 - Required Tag Missing 2 - Tag not defined for this message type 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

Field	Field Name	Req	Data Type	Description	Valid Values
					11 - Invalid MsgType 13 - Tag appears more than once 14 - Tag specified out of required order 15 - Repeating group fields out of order 16 - Incorrect NumInGroup count for repeating group 99 - Other
58	Text	O	String	Identifies the reason for the rejection. Incoming messages can be up to 128 characters For further details refer to the Error Code List	
	Standard Header	M	Trailer	Trailer	

6.2.10 Reject (3) Example

```
8=FIXT.1.1|9=114|35=3|49=ASX-CSP|56=ASX|34=2|52=20220721-01:21:10.202|45=2|372=AE|373=9|58=Illegal SenderCompId 'TEST', expected '[65, 83, 88]'|10=115|
```

6.2.11 Security Definition (d) (New)

The Security Definition (d) message is used to notify an Approved Market Operator of security details held in CHESS. A batch of 'd' messages will be disseminated to the AMO after the completion of the logon for each FIX connection. AMOs will be able to nominate whether to receive this batch download of messages based upon a subscription configuration per CompID.

Field	Field Name	Req	Data Type	Description	Valid Values
	Standard Header	M	Header	MsgType =d	
<Instrument>					
55	Symbol	M	String (6)	Identifies the security.	
225	IssueDate	O	LocalMktDate	Date instrument was issued. The date in which a security starts official quotation.	
965	SecurityStatus	M	String (3)	The current status of the sub-register in the Clearing Service for a security code.	1=Active 2 = Inactive 9= Suspended 100 = Archived (ASX customised code)
<EvntGrp>					
Start of repeating group 864, NoEvents					
865	Event Type	O	int	Code to represent the type of event.	6 = Inactivation to represent the Last Listed Date
866	Event Date	O	LocalMktDate	Date of event of the Last Listed Date	
<MarketSegmentGrp>					
Start of Repeating Group 1310, NoMarketSegments					
1300	MarketID	M	Exchange (4)	The MIC code of the market operator. Only approved AMO codes will be accepted.	XASX CHIA XNEC
End of Repeating Group 1310, NoMarketSegments					
	Trailer	M	Trailer	Trailer	

6.2.12 Security Definition (d) Example

```
8=FIXT.1.1|9=54|35=d|49=ASX-CSP|56=ASX|34=9|52=20240214-
23:30:34.395|55=CBA|225=20220214|965=2|865=6|866=20240210|1300=XASX|10=085|
```

6.2.13 SequenceReset (4)

The Sequence Reset message is used to increase the sequence number of the message. It has 2 modes:

Gap Fill mode: Where *GapFillFlag* (123) field has the value Y. This mode is used a response to the Resend Request when one or more messages must be skipped over during normal resend processing. It is used to replace administrative or application messages that will not be resent.
Reset mode: Where the *GapFillFlag* (123) field has the value N. or is omitted. This mode is used to re-establish a FIX session after an unrecoverable application failure has occurred. When using this, a higher sequence number is nominated by the sender.

Field	Field Name	Req	Data Type	Description	Valid Values
	Standard Header	M	Header	MsgType = 4	
123	GapFillFlag	O	Boolean	Indicates that the Sequence Reset message is replacing administrative or application messages which will not be resent	N = Ignore Msg Seq Num Y = Msg Seq Num Field Valid
36	NewSeqNo	M	SeqNum	New sequence number	
	Standard Trailer	M	Trailer	Trailer	

6.2.14 SequenceReset (4) Example

```
8=FIXT.1.1|9=95|35=4|49=ASX-CSP|56=ASX|34=3|43=Y|52=20220524-04:18:34.583|122=20220524-04:18:34.583|123=Y|36=5|10=044|149
```

6.2.15 Logout (5)

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

Field	Field Name	Req	Data Type	Description	Valid Values
	Standard Header	M	Header	MsgType = 5	
1409	SessionStatus	O	Int	Identifies the status of the FIX session It is returned by ASX Clear if the Logout is sent as a result of authentication failure	3 - New session password does not comply with policy 4 - Session logout complete 5 - Invalid username or password 6 - Account locked 9 - Sequence number too low 104 - *HeartBtInt* (108) must be between 10 and 60 (seconds) 106 - Logon (A) message could not be recognised
58	Text	O	String	Free format text string If a Logout is sent as a result of authentication failure, ASX Clear will provide a reason in this field	
	Standard Trailer	M	Trailer	Trailer	

6.2.16 Logout (5) Example

```
8=FIXT.1.1|9=93|35=5|49=ASX-CSP|56=ASX|34=3|52=20220524-04:17:10.430|369=3|58=Session logout complete|1409=4|10=034
```


6.2.17 BusinessMessageReject (j)

The Business Message Reject message can reject an application-level message which fulfils session-level rules and cannot be rejected via any other means.

Note: If a message fails a session-level rule (e.g. BodyLength is incorrect), a session-level Reject message is issued.

Field	Field Name	Req	Data Type	Description	Valid Values
	Standard Header	M	Header	MsgType = j	
45	RefSeqNum	M	SeqNum	Identifies the Message Sequence Number (MsgSeqNum) of rejected message	
372	RefMsgType	M	String	Identifies the MsgType of the FIX message being referenced	AE W
380	BusinessRejectReason	M	Int	The code that identifies the reason for the rejection	0 - Other 3 - Unsupported message type 5 - Conditionally Required Field Missing
58	Text	O	String	Identifies the reason for the rejection Incoming messages can be up to 128 characters	
	Standard Trailer	M	Trailer	Trailer	

6.2.18 BusinessMessageReject (j) Example

```
8=FIXT.1.1|9=138|35=j|49=ASX-CSP|56=CHIX|34=2|52=20190919-06:27:39.270|1128=9|45=2|372=F|380=3|58=Only AE and W message types are supported by FIX Gateway|10=128|
```

6.3 FIX Application Messages

6.3.1 TradeCaptureReport (AE)

A **TradeCaptureReport** (AE) enables an Approved Market Operator to submit a trade that has been executed on their trading venue or cancel a trade so that the trade (or trade cancellation) can be validated for acceptance into the ASX Clear and Settlement Services.

For trade cancellations, the following tags must match with the details from the original **TradeCaptureReport** (AE) otherwise the cancellation will be rejected. Other fields will be ignored:

- TradeDate (75)
- SecondaryTradeID (1040)
- MarketId (1301)
- Symbol (55)
- LastQty (32)
- LastPx (31)

Trade cancellations should optionally be sent via the same FIX Session as the original trade to avoid the possibility of a race condition, whereby the trade cancellation message is queued and processed before the originating trade message. Should this occur, the trade cancellation would be rejected.

Field	Field Name	Req	Data Type	Description	Valid Values
	Standard Header	M	Header	MsgType = AE	
487	TradeReportTransType	M	Int (1)	Identifies that the AMO is either submitting a new trade to the Clearing and Settlement Service or is cancelling an existing trade that has already been registered within the Clearing and Settlement Service	0 - New 1 - Cancel
856	TradeReportType	O	Int (1)	Identifies the type of Trade Report This tag is for ASX Clear Settlement internal use only This will only be used by ASX Clear Settlement Operations under specific scenarios (e.g. replay previously rejected Trade Capture Report message) This tag must not be populated by any AMO when sending a TradeCaptureReport (AE) message	5 - No/Was (i.e. Used to report a full replacement of a previously submitted trade)

Field	Field Name	Req	Data Type	Description	Valid Values
				If a Trade Capture Report message is received from any AMO containing this tag then this will be rejected	
1125	OrigTradeDate	C	LocalMktDate (8)	Identifies the original trade date of the trade being submitted or cancelled Must be present where the: • TradeReportTransType (487) is provided with the value 0 (New) and the AsOfIndicator (1015) is provided with the value 1 (True) • The value must be a valid business date prior to the current business date • When the TradeReportTransType (487) is provided with the value 0 (New) and the AsOfIndicator (1015) is provided with the value 0 (False) (i.e. a new trade that is not an As Of Trade) then the tag OrigTradeDate (1125) must not be present • Must not be provided when TradeReportTransType (487) is equal to (1) Cancel	
1003	TradeID	M	String (32)	Unique Transaction Identifier	
1040	SecondaryTradeID	M	String (10)	The unique identifier assigned to the trade or trade cancellation by the AMO Must be unique for a single business day The 1st character must indicate the AMO submitting the TradeCaptureReport (AE): • ASX Trade = 1 or 2 • CHI-X = C • NSX = N	10 character alphanumeric value Serial Trade Qualifier + Trade Serial Number
75	TradeDate	M	LocalMktDate (8)	Identifies the date the trade or trade cancellation is submitted to the Clearing and Settlement Service The value provided must always be equal to the current business date	

Field	Field Name	Req	Data Type	Description	Valid Values
1015	AsOfIndicator	O	Char (1)	Indicates the trade was reported prior to the current business date The actual trade date should be reported in OrigTradedate (1125) where the tag AsOfIndicator (1015) is provided with the value 1 (True) Must not be provided when TradeReportTransType (487) is equal to (1) Cancel	0 - False (ie. trade is not an AsOf Trade) 1 - True (ie. trade is an AsOf Trade)
64	SettlDate	O	LocalMktDate (8)	Identifies the Settlement Date for the Trade being captured Where it is provided, the value must be a valid business date greater than the current business date The CSP will adjust this date to the First Settlement Date of the Security where it is greater than the Settlement Date Provided Where a value is not provided then CSP will derive a Settlement Date as the greater of: • 2 business day from the current business date (1 business day from the current business date where the tag AsOfIndicator (1015) is provided with the value 1 (True); or • The First Settlement Date of the Security The Settlement Date applied to the trade that is registered in the CSP will be provided in the AR message response	

Field	Field Name	Req	Data Type	Description	Valid Values
				The Clearing Service will not utilize SettlType (63) to determine a Settlement Date for a Trading	
60	TransactTime	M	UTCTimestamp	Trade or Trade Cancellation agreement date and time The UTCTimestamp must be a UTC (Universal Time Coordinated, also known as "GMT") date and time field formatted as either YYYYMMDD-HH:MM:SS (whole seconds) or YYYYMMDD-HH:MM:SS.sss (milliseconds)	Observe the usage of dashes and colons separating date and time components below. YYYY = 0000-9999, MM = 01-12, DD = 01-31, HH = 00-23, MM = 00-59, SS = 00-60 (60 only if UTC leap second) (without milliseconds) YYYY = 0000-9999, MM = 01-12, DD = 01-31, HH = 00-23, MM = 00-59, SS = 00-60 (60 only if UTC leap second), sss=000-999 (indicating milliseconds) - UTCTimestamp only allows 3 digits of milliseconds
<Instrument>Component Starts					
55	Symbol	M	String (6)	Identifies the security on the trade or the trade that is cancelled The AMO must provide the proprietary security code (ticker symbol) that is maintained by ASX Clear	
<Instrument> Component Ends					
31	LastPx	M	Price	The price of the security bought or sold on the trade being captured or on the trade cancelled. The LastPx (31) should be formatted as follows: • The price should be in dollars and cents; • The precision must not exceed 6 decimal places; and	

Field	Field Name	Req	Data Type	Description	Valid Values
				• The price must be greater than or equal to 0.001000 • Total Digits (td): 15, Fraction Digits (fd): 6 and Max Inclusive: 999999999.999999	
32	LastQty	M	Qty (11)	The quantity of the security bought or sold on the trade being captured or on the trade cancelled The LastQty (32) should be formatted as follows: • The quantity must be a whole number greater than or equal to 1 • Total Digits (td): 11 and Max Inclusive: 99999999999	
15	Currency	M	Currency (3)	Identifies the currency of LastPx (31)	AUD
1301	MarketID	M	Exchange (4)	The MIC code of the market operator Only approved AMO codes will be accepted	• XASX • CHIA • XNEC
20003	TrdConditionCode	O	String (14)	Identifies the condition(s) under which the trade was effected Where applicable for the trade, the AMO can provide up to 5 unique condition code values in a single TrdConditionCode (20003) tag where each code is delimited by space Please note that this tag is an extension to FIX 5.0 SP2	Condition Code values must be provided in upper case The Condition Codes can be found here
20007	CorporateAction	O	String (8)	Identifies the Basis of Quotation (BOQ) if effected outside the default basis of quotation (i.e. where there is special market) Where applicable for the trade, the AMO may optionally provide up to 3 unique Basis of Quotation (BOQ) values in a single CorporateAction (20007) tag where each value is delimited by space	Basis of Quotation values must be provided in upper case The Basis of Quotation codes can be found here

Field	Field Name	Req	Data Type	Description	Valid Values
				Please note that this tag is an extension to FIX 5.0 SP2	
<TrdCapRptSideGrp>Component Starts					
552	NoSides	M	NumInGroup (1)	Identifies the number of sides on the trade or trade cancelled i.e. the number of instances of repeating group 552 “NoSides” included within the message Two instances of repeating group 452 “NoSides” are required One instance to report the buy side details of the trade and the other instance to report the sell side details of the trade The side of the trade is identified in tag Side (54)	2
Start of Repeating Group 552, NoSides					
54	Side	M	Char (1)	Identifies the side of the trade, buy or sell, that is reported in the repeating group Two instances of repeating group 552 “NoSides” are required One instance to report the buy side of the trade (the value 1 present in tag Side (54)) and the other instance to report the sell side of the trade (the value 2 present in tag Side (54))	Valid values: 1 - Buy 2 - Sell
<Parties>Component Starts					
453	NoPartyIDs	M	NumInGroup (1)	Identifies the number of party blocks reported on one side of the trade i.e. the number of instances of repeating group NoPartyIDs (453) The repeating group below should contain unique combinations of PartyID, PartyIDSource and PartyRole An instance of the repeating group with PartyuRole (452) = ‘1’ (Executing Firm) must be provided to	1 to 5 to match the number of instances of repeating group NoPartyIDs (453) for the side of the trade

Field	Field Name	Req	Data Type	Description	Valid Values
				identify the Trading Participant on the buy side and sell side of the trade See the following table for expected/valid values in NoPartyIDs (453) group	
Start of Repeating Group 453, NoPartyIDs					
448	PartyID	M	String	Identifies the party reported in the repeating group	
447	PartyIDSource	M	String (1)	Specifies the type of Identifier used for the Party	C – Market Participant Identifier D – Proprietary / Custom code B – BIC (Bank Identification Code) G – MIC (Market Identifier Code)
452	PartyRole	M	Int (3)	Specifies the role of the party reported in the repeating group	1 - Executing Firm (i.e. Trading Participant) 4 - Clearing Firm (i.e. Clearing Participant) 45 - Secondary Account Number (i.e. Trading account from trading Participant) 21 - Clearing Organisation 119 - Central Securities Depository (CSD)
2376	PartyRoleQualifier	O	Int	Used to further qualify the value of PartyRole (452), specifically for the house/client segregation. Should only be specified where PartyRole (452) = 1 (Executing Firm)	0 - Agency 1 - Principal 100 - Mixed
<Parties>Component Ends					
End of Repeating Group 453, NoPartyIDs					

Field	Field Name	Req	Data Type	Description	Valid Values
End of Repeating Group 452, NoSides					
<TrdCapRptSideGrp>Component Ends					
End					
	Standard Trailer	M	Trailer	Trailer	

6.3.2 TradeCaptureReport (AE) Example

```
8=FIXT.1.1|9=325|35=AE|49=CHIX|56=ASX-CSP|34=80238|1128=9|52=20240214-
00:56:27.916|487=0|1040=Cf2T8nqno0|75=20240214|1015=0|64=20220517|60=20240214-
11:56:11.475|55=JPM|31=2.3|32=69|15=AUD|1301=CHIA|552=2|54=1|453=2|448=0001|447=C|452=1|448=20036|447=C|452=4|54=2|453=2|
448=0002|447=C|452=1|448=20037|447=C|452=4|10=176|
```

6.3.3 Parties Group Valid Values

There are 5 possible PartyRole types as outlined in the following table.

PartyRole (452)	Req	PartyIDSource (447)	PartyID (448)
1 - Executing Firm (i.e. Trading Participant)	M	C – Market Participant Identifier	4 digit ID of Trading Participant (e.g. 1234)
4 - Clearing Firm (i.e. Clearing Participant)	O	C – Market Participant Identifier	5 digit UIC of Clearing Participant (e.g. 01234)
45 - Secondary Account Number	O	D – Proprietary / Custom code	Custom string from order entry data up to 10 characters alpha-numeric
21- Clearing Organisation	O	B – BIC (Bank Identification Code) or G – MIC (Market Identifier Code)	Valid BIC or MIC
119- Central Securities Depository (CSD)	O	B – BIC (Bank Identification Code) or G – MIC (Market Identifier Code)	Valid BIC or MIC

6.3.4 TradeCaptureReportAck (AR)

The **TradeCaptureReportAck** (AR) provides an Approved Market Operator with a success or failure or pending response to the **TradeCaptureReport** (AE) received for either reporting a new trade or cancelling an existing trade.

Field	Field Name	Req	Data Type	Description	Valid Values
	Standard Header	M	Header	MsgType = AR	
<ApplicationSequenceControl>					
487	TradeReportTransTyp e	M	Int	Identifies that the referenced transaction was either a new trade or the cancellation of an existing trade	0 – New 1 – Cancel
1003	TradeID	M	String (32)	Unique Transaction Identifier	
1040	SecondaryTradeID	M	String (10)	Indicates the unique identifier of the trade or of the trade cancellation Will be present with the SecondaryTradeID (1040) value provided on the TradeCaptureReport (AE) received noting that: • The value will be unique for the business day; and • The 1st character will indicate the AMO in which submitted the TradeCaptureReport (AE) received: ○ ASX Trade = 1 or 2; ○ CHI-X = C; or ○ NSX= N	10 character alphanumeric value Serial Trade Qualifier + Trade Serial Number
939	TrdRptStatus	M	Int (1)	Indicates the TradeCaptureReport (AE) received was accepted, rejected, cancelled or pending by the CSP	0 = Accepted 1 = Rejected 2= Cancelled 4= Pending New
<TrdRegTimestamps>					
64	SettlDate	O	LocalMktDate (8)	Indicates the Settlement Date of the Trade For a trade submitted to the CSP, the settlement date will be derived as follows:	

Field	Field Name	Req	Data Type	Description	Valid Values
				• Where the SettlDate (64) value is provided on the TradeCaptureReport (AE) received this settlement date will be the greater of the SettlDate (64) value provided or the First Settlement Date of the Security; or • Where a value is not provided then CSP will derive a Settlement Date as the greater of 2 business day from the current business date (1 business day from the current business date where the tag AsOfIndicator (1015) is provided with the value 1 (True) or the First Settlement Date of the Security For a trade cancellation submitted to the CSP, the value present will be the SettlDate (64) value, where it is provided, on the TradeCaptureReport (AE)	
60	TransactTime	M	UTCTimestamp	Indicates the Trade or Trade Cancellation agreement date and time Will be present with the TransactTime (60) value provided on the TradeCaptureReport (AE) received	
<Instrument>					
55	Symbol	M	String (6)	Indicates the security on the trade or the trade that is cancelled Will be present with the Symbol (55) (i.e. Security Code) value provided on the TradeCaptureReport (AE) received This value will be the proprietary security code (ticker symbol) that is maintained by ASX Clear	
751	TradeReportRejectReason	O	Int	Indicates the trade rejection reason Where the TradeCaptureReport (AE) received is: • Accepted, then the value 0 (Successful) will be present; or • Rejected, the then value 99 (Other) will be present	0 = Successful (default) 99 = Other

Field	Field Name	Req	Data Type	Description	Valid Values
				This field will not be populated for pending or cancel	
1328	RejectText	O	String	Indicates the reason the TradeCaptureReport (AE) received was rejected This value will be present where TradeReportRejectReason (751) is populated with the value 99	
58	Text	O	String	Indicates the reason the TradeCaptureReport (AE) received was pending	
End					
	Standard Trailer	M	Trailer	Trailer	

6.3.5 TradeCaptureReportAck (AR) Example

```
8=FIXT.1.1|9=107|35=AR|49=ASX-CSP|56=ASX|34=2|52=20220524-
04:17:07.457|1128=9|939=0|751=0|55=CBA|64=20220517|1040=1MAT206smi|10=024|
```

6.3.6 MarketDataSnapshotFullRefresh (W)

The Market Data Snapshot Full Refresh message provides a complete snapshot of price details for a Security.

The snapshot will provide the:

- Opening Price
- Last Traded Price
- Trading Session High
- Trading Session Low
- On-Market Volume
- Off-Market Volume
- Bid Price
- Offer Price

Note: End of day pricing details for all securities for the current business date should be captured by the AMO at a suitable point in time after continuous trading has closed for the current trading day and are expected to be received by CSP prior to 5:15pm AEST.

A subsequent update can be sent before 7:00pm AEST for a security where any of the end of day pricing details provided prior to 5:15pm AEST have changed (due to late reporting of a trade, a trade cancellation, or any other adjustment or administration performed). This update may include either a full retransmission of all securities or only incremental changes.

Field	Field Name	Req	Data Type	Description	Valid Values
	Standard Header	M	Header	MsgType = W	
75	TradeDate	M	LocalMktDate	The business date the pricing details have been captured for the Security This value must be a valid business date equal to the current business date	
60	TransactTime	M	UTCTimestamp	The timestamp the pricing details have been captured for the Security The UTCTimestamp must be a UTC (Universal Time Coordinated, also known as "GMT") date and time field formatted as either YYYYMMDD-HH:MM:SS (whole seconds) or YYYYMMDD-HH:MM:SS.sss (milliseconds)	• Observe the usage of dashes and colons separating date and time components below: • YYYY = 0000-9999, MM = 01-12, DD = 01-31, HH = 00-23, MM = 00-59, SS = 00-60

Field	Field Name	Req	Data Type	Description	Valid Values
					(60 only if UTC leap second) (without milliseconds) • YYYY = 0000-9999, MM = 01-12, DD = 01-31, HH = 00-23, MM = 00-59, SS = 00-60 (60 only if UTC leap second), sss=000-999 (indicating milliseconds) - UTCTimestamp only allows 3 digits of milliseconds
1301	MarketID	M	Exchange	MIC code that identifies the Approved Market Operator (AMO) Only approved AMO codes will be accepted	• XASX • CHIA • XNEC
Instrument					
55	Symbol	M	String	Identifies the security for which the pricing details provided pertain to The AMO must provide the proprietary security code (ticker symbol) that is maintained by ASX Clear	
20021	TradeVolume	M	Qty	Indicates the buy-side off-market volume traded in the Security for the business day reported	
20007	CorporateAction	O	String	Identifies the basis on which the pricing information is quoted (i.e. indicates whether the pricing includes or excludes an entitlement relating to a current corporate actions for the underlying security) The tag accepts up to 3 valid values delimited by space	Basis of Quotation values must be provided in upper case The Basis of Quotation codes can be found here
MDFullGrp					
268	NoMDEntries	M	NumInGroup	Number of MDEntry repeating group instances in this market data message	
Start of Repeating Group 268, NoMDEntries					
269	MDEntryType	M	Char	Identifies the pricing information reported in the repeating group A repeating group must be specified for each of the values	0 - Bid price 1 - Offer price 2 - Trade (Last Traded Price)

Field	Field Name	Req	Data Type	Description	Valid Values
					4 - Opening Price 7 - Trading session high price 8 - Trading session low price B - Trade Volume
270	MDEntryPx	C	Price	Identifies the price reported for the repeating group identified by the value in tag MDEntryType (269) Required where the related MDEntryType (269) value is not B (Trade volume)	
15	Currency	O	Currency	Identifies the currency of the pricing information reported Must be provided with the value AUD Required where the related MDEntryType (269) value is not B (Trade Volume)	AUD
271	MDEntrySize	C	Qty	Indicates the buy-side on-market volume traded in the Security for the business day reported Required for the Repeating Group identified by MDEntryType (269) = B (Trade Volume) else must not be provided	
272	MDEntryDate	O	UTCDateOnly	Identifies the date of the Last Traded Price being reported This is required when MDEntryType (269) = 2 - Trade, else must not be provided	
273	MDEntryTime	O	UTCTimeOnly	Identifies the time of the Last Traded Price, Market Bid price or Market Offer price being reported This is required when MDEntryType (269) = 2 - Trade, b - Market bid, or c - Market offer; else must not be provided	
End of Repeating Group 268, NoMDEntries					
End					
	Standard Trailer	M	Trailer	Trailer	

6.3.7 MarketDataSnapshotFullRefresh (W) Example

```
8=FIXT.1.1|9=0000388|35=W|49=ASX AMO|56=ASX|34=20478|52=20190411-10:11:58.000|75=20190411|60=20190411-
10:11:58.000|1301=XASX|55=WOW|20021=50000|20007=CB|268=7|269=0|270=98.18|15=AUD|273=04:54:48.854|269=1|270=98.18|15=AUD|2
73=04:54:48.854|269=2|270=98.125321|15=AUD|272=20190403|273=04:54:48.854|269=4|270=98.18|15=AUD|269=7|270=98.18|15=AUD|26
9=8|270=98.18|15=AUD|269=B|271=1460500|10=150|
```

6.3.8 ConfirmationAck (AU)

A **ConfirmationAck** (AU) provides an Approved Market Operator (AMO) with a success or failure response to the **MarketDataSnapshotFullRefresh** (W) received for reporting price information.

Field	Field Name	Req	Data Type	Description	Valid Values
	Standard Header	M	Header	MsgType = AU	
664	ConfirmID	M	String	References the MarketDataSnapshotFullRefresh (W) this response pertains to The value present will be the MsgSeqNum (34) provided in the MarketDataSnapshotFullRefresh (W) received	
75	TradeDate	M	LocalMktDate	Indicates the business date the pricing details have been captured for the Security This value present will be the TradeDate (75) provided in the MarketDataSnapshotFullRefresh (W) received	
60	TransactTime	M	UTCTimestamp	Indicates the timestamp the pricing details have been captured for the Security This value present will be the TransactTime (60) provided in the MarketDataSnapshotFullRefresh (W) received	
55	Symbol	M	String	Identifies the security for which the pricing details provided pertain to The AMO must provide the proprietary Security Code (ticker symbol) that is maintained by ASX Clear This value present will be the Symbol (55) provided in the MarketDataSnapshotFullRefresh (W) received	
940	AffirmStatus	M	Int	Indicates MarketDataSnapshotFullRefresh (W) was accepted (i.e. 3 - Affirmed) or rejected (i.e. 2 - Confirm Rejected) by the CSP	2 = Confirm rejected 3 = Affirmed
774	ConfirmRejReason	C	Int	Indicates the rejection reason where the MarketDataSnapshotFullRefresh (W) received is rejected by the CSP This value will be present with the value 99 (Other) where the tag AffirmStatus (940) is present with the value 2 (Confirm rejected)	99 = Other Refer to tag *Text* (58) for the rejection reason
58	Text	O	String	Indicates the rejection reason where the MarketDataSnapshotFullRefresh (W) received is rejected by the CSP	

Field	Field Name	Req	Data Type	Description	Valid Values
				This value will be present where the tag ConfirmRejReason (774) is present with the value 99 (Other)	
End					
	Standard Trailer	M	Trailer	Trailer	

6.3.9 ConfirmationAck (AU) Example

```
8=FIXT.1.1|9=0000150|35=AU|49=ASX|56=ASXAMO|34=20478|52=2019040-10:11:58.000|
664=20478|75=20190403|60=20190421-10:15:58.000|55=WOW|940=2|774=99|58=<55><Symbol>:[WOWI] is invalid|10=194|
```

7 Appendix

7.1 Trade Condition Codes Table

The Trade Condition code is an optional field that is used by the AMO submitted to the CSP via the **TradeCaptureReport** (AE) for the purpose of market trade acceptance and registration. The CSP is then expected to accept the new market trade for registration and generate a confirmation message **TradeCaptureReportAck** (AR) to the AMO.

TBA, details of valid Trade Condition Codes will be added to this section in future versions of this document.

7.2 Corporate Action and Basis of Quotation Codes

The *CorporateAction* (20007) FIX tag is referred to as "Basis of Quotation" in the Legacy Service. This field indicates the status under which a security is quoted. In the case of trades, this field will only contain a value if special permission has been granted by the respective committees to trade outside the currently stated Basis of Quotation.

The table below lists the valid values for the *CorporateAction* (20007) tag.

TBA, details of valid Corporate Action and Basis of Quotation codes will be added to this section in future versions of this document.