ASX Trade OI
Application Conformance Process

Version 1.3 | July 2023

Table of Contents

1. Introduction

The ASX application conformance test process is designed to help protect market participants from any erroneous application behaviour that may be disruptive to the market due to non-conformance.

The conformance process applies to any application that connects to the ASX production environment for trading or market data services.

All customer applications must pass the ASX conformance process prior to accessing the production market for trading or receipt of market data.

Except for certain mandatory functions, customers are required to complete only the conformance test scenarios which relate to the functionality they will use in the production environment. All other test scenarios may be omitted.

1.1 Purpose

The purpose of this document is to provide an overview of the ASX Trade conformance test and set expectations regarding conduct and expected results.

1.2 Readership

This document outlines the test phases customers need to pass to certify their trading and market data related software applications.

Technical staff within ASX Trade participants and information vendors and other market participants that connect directly to the ASX Trade Platform are expected to read this document and understand the requirements of this process.

1.3 Document History

Issue	Date	Description
1.0	Jan 2018	First release of document.
1.1	Mar 2020	Updated for the ASX Trade refresh
1.2	Jun 2020	Categorization of test cases modified
1.3	Jul 2023	Updated section 2.1 When to Repeat Conformance Testing

1.4 Enquiries

Please contact CTS @asx.com.au or your Technical Account Manager if you have any questions relating to this document.

2. ASX Conformance Process

The ASX conformance process provides procedures that customers can use to ensure that their application software conforms to ASX Trade operating rules and technical specifications.

ASX expects customers to become conversant with the ASX development platform and develop their applications to a state of readiness for conformance testing. At this point they can contact ASX to schedule a date and time to undertake the conformance test. ASX requires customers to have finalised their software, meaning that said software is in a production-ready state, before booking the test.

Customers must submit an application conformance checklist and indicate within, the test scenarios their application will not complete when arranging with CTS to undertake the conformance test.

Customers perform the majority of the conformance test unassisted and complete the process by advising CTS about start and end time of the self-test and the Instruments used for the test. CTS will then check the available logs to verify the result and produce a pass or fail report.

Once a pass is verified by CTS, customers can go on to complete the ASX-assisted test scenarios assisted by a member of the CTS team. CTS assisted test slots are of 30 minutes duration. If testing is not completed within a test slot, customers will need to book an additional slot.

2.1 When to Repeat Conformance Testing?

While it is encouraged to perform regular testing, customers are required to perform software conformance with the ASX when:

- Software is modified in any way that may affect or directly impact ASX connectivity or messaging.
- Additional ASX facing functionality is added to an already conformed application.
- Software is recompiled for a new operating system.
- ASX upgrades or changes its production environment and deems the change mandatory.
- During extended periods of absence when the software is not connected to ASX.
- Erroneous or disruptive behaviour is identified in the Production Environment.
- Upon specific request from the ASX.

ASX software conformance testing is optional but recommended when:

- Customer software is recompiled on the same operating system with minor changes, or new builds which have no assumed impact to ASX connectivity or messaging.

- Independent Software Vendor (ISV) applications have passed conformance testing and the same software and version is to be used by new or existing customers of the ISV on new customer infrastructure.

2.2 Non-Compliant software

Warning:

ASX reserves the right to block access to the production system by non-compliant software. ASX will require successful completion of a conformance test prior to re-connection.

2.3 Application Conformance Testing

To successfully complete Application Conformance Testing the customer's software application needs to have established a connection to the ASX Trade Testing Environment (FTE/ETE). These platforms simulate the ASX production environment and is designed to give a "Production-like" platform to develop and test against.

2.4 Prior to Conformance

Customers should ensure they have read and understand the specification documents pertaining to the interfaces they are developing against.

Customers should then discuss the intended functionality of their applications with their TAMs to ensure all steps were followed and the process is fully understood.

The CTS team will be available to assist customers in completing their tests, if required.

Customers must submit the application conformance checklist as provided by the CTS team with all functions not supported by the software clearly marked.

The checklist must be sent to CTS@asx.com.au prior to the test to validate the successfully attempted test scenarios.

2.5 Expectations during Conformance

- Customers are responsible for ensuring prices and statistics relating to market data and order entry are calculated correctly and displayed according to best market practices.
- Customer must maintain a stable connection.
- Customers testing market data applications must ensure that securities with frequently updating information are thoroughly tested.
- Customers should refer to the ASX Operating Rules for guidelines on acceptable market behaviour. Adherence to these guidelines is required throughout all customer application conformance testing. Failure to comply with these guidelines will result in failure of the conformance test.

2.6 Categories

Mandatory – these test cases must be attempted and passed.

Supported Functionality – these cases are conditionally mandatory if you intend to support this functionality in production.

Optional - can be skipped if the application does not support this function. The function may not be used in production.

Not Supported – on the checklist, “Optional” and “Supported Functionality” cases can be set to Not Supported only if this functionality will not be used in production.

3. Conformance Test Scenario

The customer is responsible for completing this section of the conformance test. All activities performed during this segment of the conformance test are logged. On completion of this segment of the conformance test, ASX will review logs to ensure that applicable test scenarios were successfully completed by the customer application.

3.1 Connection

The following set of test scenarios define how applications must connect and disconnect from the environment to meet ASX conformance requirements.

3.1.1 Log on

Category	Mandatory for OI applications
Description	The application must perform a standard logon. The application must sustain connectivity and receive a successful connection acknowledgement.
	The application must pass this test to progress to the next phase of functionality testing.
Notes	
Requirements	It is expected that the tester will have confirmed network connectivity to the testbed prior to commencement of accreditation.
	If concurrent connection type is used to read broadcast a separate connection is made prior to the logon connection.

#	Test Scenarios	Expected results
1.	Establish a connection and log on to the gateway.	Participant receive a cstatus as OMNIAPI_SUCESS or a positive value confirming a successful logon.
		Participant need to read the BI9 heartbeats off their queue.

3.1.2 Logout

Category	Mandatory for OI applications
Description	The purpose of this test is to ensure that applications log out gracefully.
Notes	
Requirements	

#	Test Scenarios	Expected results
1.	Log out gracefully and disconnect from the session.	Participant receive a cstatus as OMNIAPI_SUCESS or a positive value confirming a graceful log off.

3.1.3 Password Change

Category	Mandatory for OI applications
Description	The purpose of this test is to ensure that the application can perform a password change as part of a logon to the environment to demonstrate to ASX that the application can manage its own password in day to day operations
	The application must pass this test to progress to the next phase of functionality testing.
Notes	OMNET application passwords expire every 90 days, and must be changed by the application prior to this or it will not be able to logon to the Exchange.
Requirements	Applications must adhere to the ASX Password policy, outlined in ASX Trade OI manual.

#	Test Scenarios	Expected results
1.	Establish a connection and send new password to the exchange.	Password successfully changed and application logging in with new password.

3.2 Download and Subscription

3.2.1 Broadcast Subscription

Category	Mandatory for OI applications
Description	The purpose of this test is to ensure that the application can request and receive market and segment information.
	After login users must subscribe to applicable broadcasts, retrieve and hold them in sequential order in an internal queue while their associated queries are processed.
Notes	Each different type of user is allocated a certain set of broadcasts. Typically users would retrieve the broadcasts to which they are allowed to subscribe. Having done this user would then iterate through those broadcasts, and subscribe to each.
	If concurrent connection type is used to read broadcast a separate connection is made prior to the logon connection.
Requirements	

#	Test Scenarios	Expected results
1.	The omniapi_set_event_ex() routine is used to set up a subscription to broadcasts	Application will receive cstatus for the successful subscription.
2	Concurrent broadcast type to be used to ensure the queue is being cleared	A separate connection is made prior to the logon connection to process the broadcast messages.

3.2.2 Download required market information utilising delta queries

Category	Mandatory for OI applications
Description	Application downloads all required markets, instrument types, class and series information utilising delta queries
	Users can download all required instruments from the central system. Delta queries ought to be utilised to request instruments, class and underlying. The system must be able to handle intraday security updates (underlying, class and series).
Notes	
Requirements	

#	Test Scenarios	Expected results
1.	Application to send delta queries DQ126, DQ124, DQ122 and DQ120	The user needs to call the function omniapi_query_ex(…), providing various messages structures as a parameter, and then parse the response.

3.2.3 Application downloads available Partition information and validate trades in applicable market

Category	Optional for all OI applications
Description	The purpose of this test is to ensure that OI applications can send through a request to receive all partition and trade information.
Notes	
Requirements	

#	Test Scenarios	Expected results
1.	Application downloads available Partition information via UQ1 or is mapping series to Partitions via CQ112	Confirm number of partitions received.
2.	Application view trades correctly in different markets and partitions including trades carrying 4 decimal places trade price such as trades from Centre Point orders.	Application confirm the Trade details and can track Trade Slip Number.

3.2.4 Sending a UI1 message

Category	Mandatory for OI applications
Description	The purpose of this test is to ensure that the application can send the ready state by sending UI1 transaction.
Notes	After subscribing to broadcasts, retrieving them, holding them in an internal queue, querying the system for a market snapshot, then applying the held broadcasts, the user is now ready to begin trading or gathering information on the market. At this stage, ASX Trade requires the user to inform the system their ready state by issuing a UI1 transaction.
Requirements	

#	Test Scenarios	Expected results
1.	Application to inform the system their ready state by issuing a UI1 transaction.	Presence of UI1 message after all queries in log. UI1 = Ready to trade

3.2.5 Viewing Trading Session States

Category	Mandatory for OI applications
Description	The purpose of this test is to ensure that the application can monitor the current session status for the markets, which it is intending to subscribe to. This dictates the valid list of actions an application can perform in that session.
Notes	The UQ15 Instrument Status query can be used to recover the information in this broadcast if the user missed it.
Requirements	

#	Test Scenarios	Expected results
1.	Application sends UQ15 and receives BI41 for Trading Session State	Verify application's ability to view different TSS of different series in different markets correctly.

3.3 Equity Order Management

3.3.1 Equity Order Entry

Category	Mandatory for Order Entry applications
Description	The purpose of this test is to ensure that the application can successfully place a new order into equity market with a required details and receive an order identifier that can be used to track the order.
Notes	Participant can pick either order type and choose any equity stock.
Requirements	

#	Test Scenarios	Expected results
1.	The application will need to send a MO1 transaction for entering orders.	The transaction returns a successful *cstatus* and the order number of the order in *ordidt.* A BO5 broadcast is received with the full order details.

3.3.2 Equity Order Modification

Category	Mandatory for Order Entry applications
Description	The purpose of this test is to ensure that the application can successfully modify the previously placed order with a new unique order number, series field and the bid/ask flag in the previously used equity stock, and receive a valid confirmation message.
	For the Purpose of this test, participant may modify either price or quantity used in the previous order.
Notes	
Requirements	To complete this test scenario, an order will need to have been placed in the market by the OI user who is connected for the Conformance test.

#	Test Scenarios	Expected results
1.	The application will need to send a MO3 transaction on the previously entered order, modifying either the price or quantity.	The transaction returns a successful *cstatus* and the order quantity before the amendment in *txstat.* A BO5 broadcast is received with the updated order details.

3.4 Options/Warrants Order Management

3.4.1 Options/Warrants Order Entry

Category	Supported Functionality for Order Entry applications
Description	The purpose of this test is to ensure that the application can successfully place a new order into option series or warrants with the required details and receive an order identifier that can be used to track the order.
Notes	Participant can pick either order type and choose any options/warrants stock.
Requirements	

#	Test Scenarios	Expected results
1.	The application will need to send a MO1 transaction for entering orders.	The transaction returns a successful *cstatus* and the order number of the order in *orddt.* A BO5 broadcast is received with the full order details.

3.4.2 Options/Warrants Order Modification

Category	Supported Functionality for Order Entry applications
Description	The purpose of this test is to ensure that the application can successfully modify the previously placed order with a new unique order number, series field and the bid/ask flag in the previously used options/warrants stock, and receive a valid confirmation message.
	For the purpose of this test, the participant may modify either price or quantity used in the previous order.
Notes	
Requirements	To complete this test scenario, an order will need to have been placed in the market by the OI user who is connected for the conformance test.

#	Test Scenarios	Expected results
1.	The application will need to send a MO3 transaction on the previously entered order, modifying either the price or quantity.	The transaction returns a successful *cstatus* and the order quantity before the amendment in *txstat.* A BO5 broadcast is received with the updated order details.

3.5 Undisclosed Order Management

3.5.1 Undisclosed Order Entry

Category	Supported Functionality for Order Entry applications
Description	The purpose of this test is to ensure that the application can successfully place a new undisclosed order into market with the required details and receive an order identifier that can be used to track the order.
	Participant can choose any stock.
Notes	
Requirements	

#	Test Scenarios	Expected results
1.	The application will need to send a MO1 transaction for entering orders with exch_order_type_n=32.	The transaction returns a successful *cstatus* and the order number of the order in *ordidt.* A BO5 broadcast is received with the full order details.

3.5.2 Undisclosed Order Modification

Category	Supported Functionality for Order Entry applications
Description	The purpose of this test is to ensure that the application can successfully modify the previously placed order with a new unique order number, series field and the bid/ask flag in the previously used stock, and receive a valid confirmation message.
	For the Purpose of this test, participant may modify either price or quantity used in the previous Undisclosed order.
Notes	When amending undisclosed orders, exch_order_type_n must be set to 32 unless the order is to be amended to become disclosed.
Requirements	To complete this test scenario, an order will need to have been placed in the market by the OI user who is connected for the conformance test.

#	Test Scenarios	Expected results
1.	The application will need to send a MO3 transaction on the previously entered order, modifying either the price or quantity.	The transaction returns a successful *cstatus* and the order quantity before the amendment in *txstat.* A BO5 broadcast is received with the updated order details.

3.6 Auction Imbalance Order Management

3.6.1 Auction Imbalance Order Entry

Category	Supported Functionality for Order Entry applications
Description	The purpose of this test is to ensure that the application can successfully place a new Auction Imbalance order into any stock with the required details and receive an order identifier that can be used to track the order.
	Auction Imbalance orders are entered with a limit price and are only accepted with time validity Fill and Kill.
Notes	
Requirements	

#	Test Scenarios	Expected results
1.	The application will need to send a MO1 transaction for entering orders with order_type_c=65.	The transaction returns a successful *cstatus* and the order number of the order in *ordidt.* A BO5 broadcast is received with the full order details.

3.7 Iceberg Order Management

3.7.1 Iceberg Order Entry

Category	Supported Functionality for Order Entry applications
Description	The purpose of this test is to ensure that the application can successfully place a new Iceberg order into stock with a visible quantity and a total quantity and receive an order identifier that can be used to track the order.
	Participant can pick either order type and choose any stock.
Notes	Iceberg orders require a visible quantity rather than a visible value
Requirements	

#	Test Scenarios	Expected results
1.	The application will need to send an Iceberg order using MO1 (minimum visible quantity = 500; total order quantity must not exceed 1,000 times the visible quantity)	The transaction returns a successful *cstatus* and the order number of the order in *orddt.* A BO5 broadcast is received with the full order details, including visible and total quantity.

3.7.2 Iceberg Order Modification

Category	Supported Functionality for Order Entry applications
Description	The purpose of this test is to ensure that the application can successfully modify the previously placed order with a new unique order number, series field, the bid/ask flag and minimum visible quantity in the previously used stock, and receive a valid confirmation message.
	For the purpose of this test, participant may modify either price or visible quantity used in the previous order.
Notes	The total order quantity may not be increased through an amendment.
Requirements	To complete this test scenario, an Iceberg order will need to have been placed in the market by the OI user who is connected for the conformance test.

#	Test Scenarios	Expected results
1.	The application will need to send a MO3 transaction on the previously entered order, modifying either the price or quantity	The transaction returns a successful *cstatus* and the order quantity before the amendment in *txstat.* A BO5 broadcast is received with the updated order details.

3.8 Centre Point Order Management

3.8.1 Centre Point Order Entry

Category	Supported Functionality for Order Entry applications
Description	The purpose of this test is to ensure that the application can successfully place a new Centre Point order into stock with the required details and receive an order identifier that can be used to track the order.
	Participant can pick either order type and choose any stock.
Notes	
Requirements	

#	Test Scenarios	Expected results
1.	The application will need to send a MO1 transaction for entering CP orders with exch_order_type_n=64.	The transaction returns a successful *cstatus* and the order number of the order in *ordidt.* A BO5 broadcast is received with the full order details.

3.8.2 Centre Point Order Modification

Category	Supported Functionality for Order Entry applications

Description	The purpose of this test is to ensure that the application can successfully modify the previously placed order with a new unique order number, series field and the bid/ask flag in the previously used stock, and receive a valid confirmation message.
	For the purpose of this test, participant may modify either price or quantity used in the previous order.
Notes	It is not possible to amend a Centre Point order into a non-Centre Point (e.g. Limit) order or to amend a non-Centre Point order into a Centre Point order.
Requirements	To complete this test scenario, a CP order will need to have been placed in the market by the OI user who is connected for the conformance test.

#	Test Scenarios	Expected results
1.	The application will need to send a MO3 transaction on the previously entered CP order, modifying either the price or quantity.	The transaction returns a successful *cstatus* and the order quantity before the amendment in *txstat.* A BO5 broadcast is received with the updated order details.

3.9 Centre Point Block Order Management

3.9.1 Centre Point Block Order Entry

Category	Supported Functionality for Order Entry applications
Description	The purpose of this test is to ensure that the application can successfully place a new CP Block order into a stock with the required details and receive an order identifier that can be used to track the order.
	Participant can pick either order type and choose any stock.
Notes	
Requirements	

#		Expected results

1.	The application will need to send a MO1 transaction for entering CP Block orders with exch_order_type_n=4096.	The transaction returns a successful *cstatus* and the order number of the order in *ordidt.* A BO5 broadcast is received with the full order details.

3.9.2 Centre Point Block Order Modification

Category	Supported Functionality for Order Entry applications
Description	The purpose of this test is to ensure that the application can successfully modify the previously placed CP Block order with a new unique order number, series field and the bid/ask flag in the previously used stock, and receive a valid confirmation message.
	For the purpose of this test, participant may modify either price or quantity used in the previous order.
Notes	It is not possible to amend a CP Block order into a non-CP Block (e.g. Limit or Centre Point) order or to amend a non-CP Block order into a CP Block order.
Requirements	To complete this test scenario, a CP Block order will need to have been placed in the market by the OI user who is connected for the conformance test.

#	Test Scenarios	Expected results
1.	The application will need to send a MO3 transaction on the previously CP Block entered order, modifying either the price or quantity.	The transaction returns a successful *cstatus* and the order quantity before the amendment in *txstat.* A BO5 broadcast is received with the updated order details.

3.10 Centre Point Sweep Order Management

3.10.1 Centre Point Sweep Order Entry

Category	Supported Functionality for Order Entry applications
Description	The purpose of this test is to ensure that the application can successfully place a new CP Sweep order into any stock with a required details and receive an order identifier that can be used to track the order.
	Participant can pick either order type and choose any stock.
Notes	
Requirements	

#	Test Scenarios	Expected results
1.	The application will need to send a MO1 transaction for entering CP limit Sweep orders with exch_order_type_n=2048.	The transaction returns a successful *cstatus* and the order number of the order in *orddt.* A BO5 broadcast is received with the full order details.

3.10.2 Centre Point Sweep Order Modification

Category	Supported Functionality for Order Entry applications
Description	The purpose of this test is to ensure that the application can successfully modify the previously placed CP Limit Sweep order with a new unique order number, series field and the bid/ask flag in the previously used stock, and receive a valid confirmation message.
	For the purpose of this test, participant may modify either price or quantity used in the previous order.
Notes	
Requirements	To complete this test scenario, an order will need to have been placed in the market by the OI user who is connected for the conformance test.

#	Test Scenarios	Expected results
1.	The application will need to send a MO3 transaction on the previously entered CP Sweep order, modifying either the price or quantity.	The transaction returns a successful *cstatus* and the order quantity before the amendment in *txstat.* A BO5 broadcast is received with the updated order details.

3.11 Short Sell Order Management

3.11.1 Short Sell Order Entry with SHL condition code

Category	Supported Functionality for Order Entry applications
Description	The purpose of this test is to ensure that the application can successfully place a new Short Sell order into a stock which has a SHL condition code with a required details and receive an order identifier that can be used to track the order.
	Participant can pick either order type and choose a stock with no SH status note attached.
Notes	
Requirements	

#	Test Scenarios	Expected results
1.	The application will need to send a MO1 transaction for entering Short Sell orders with exch_order_type_n=2 in to the specific stock which has no SH status note	The transaction returns a successful *cstatus* and the order number of the order in *orddt.* A BO5 broadcast is received with the full order details.

3.11.2 Short Sell Order Modification

Category	Supported Functionality for Order Entry applications
Description	The purpose of this test is to ensure that the application can successfully modify the previously placed Short Sell order with a new unique order number, series field and the bid/ask flag in the previously used stock, and receive a valid confirmation message.
	For the purpose of this test, participant may modify quantity used in the previous order.
Notes	
Requirements	To complete this test scenario, a short sell order will need to have been placed in the market by the OI user who is connected for the conformance test.

#	Test Scenarios	Expected results
1.	The application will need to send a MO3 transaction on the previously entered Short Sell order, with modified quantity.	The transaction returns a successful *cstatus* and the order quantity before the amendment in *txstat.* A BO5 broadcast is received with the updated order details.

3.12 Unintentional Crossing Prevention (UCP) Order Management

3.12.1 UCP Buy Order Entry

Category	Supported Functionality for Order Entry applications
Description	The purpose of this test is to ensure that the application can successfully place a new UCP buy order into a stock with a required details and receive an order identifier that can be used to track the order.
	Participant can pick either order type and choose any stock.
Notes	Participant must provide the unique value in crossing_key_i field.
Requirements	

#	Test Scenarios	Expected results
1.	The application will need to send a MO1 transaction for entering buy order with unique crossing_key_i value.	The transaction returns a successful *cstatus* and the order number of the order in *orddidt.* A BO5 broadcast is received with the full order details.

3.12.2 UCP Sell Order Entry

Category	Supported Functionality for Order Entry applications
Description	The purpose of this test is to ensure that the application can successfully place a new UCP sell order into a same stock with a required details and receive an order identifier that can be used to track the order.
	For the Purpose of this test, participant can send the opposite side order with the same UCP key value.
Notes	Participant must provide the same crossing_key_i value provided in the previously placed buy order.
Requirements	

#	Test Scenarios	Expected results
1.	The application will need to send a MO1 transaction for entering sell order with the same crossing_key_i value provided in the previously placed buy order.	The transaction returns a successful *cstatus* and the order number of the order in *orddidt.* A BO5 broadcast is received with the full order details.

3.12.3 UCP Trade Confirmation

Category	Supported Functionality for Order Entry applications
Description	The purpose of this test is to ensure that the application can successfully execute the previously placed UCP Buy and Sell order and receive a valid confirmation that the order was cancelled.
	For the purpose of this test, participant can see their own orders.
Notes	Orders from the same participant with the same Crossing Key will result in a booked transaction that is not disseminated to the market as a trade.
Requirements	To complete this test scenario, a UCP order will need to have been placed in the market by the participant who is connected for the conformance test.

#	Test Scenarios	Expected results
1.	Participant will confirm the full details of the booking report.	Participant will receive a CB15 Dedicated Trade broadcast contains the full details of the booking report, including a deal source to indicate that the trade resulted from Unintentional Crossing Prevention in continuous matching or an auction.

3.13 Tailor Made Combination (TMC) Order Management

3.13.1 TMC Order Creation

Category	Supported Functionality for Order Entry and Market Maker applications
Description	Creation of TMC allow to entre and continuous integrated matching of combination orders with other single series orders and/or other combination orders involving the same components for the specified net price.
Notes	The series in the combination cannot straddle a Market Place (MP) partition. Each series in the combination must exist in the same MP partition.
Requirements	

#	Test Scenarios	Expected results
1.	TMCs can be created using the DC3 transaction.	Application will receive a TMC identifier upon the TMC creation request.
2.	CTS creates a TMC combinations of TMCs – i.e. Buy-Buy, Sell-Sell and Buy-Sell TMCs.	User will receive a BU126 with the new TMC details.

3.13.2 TMC Order Entry

Category	Supported Functionality for Order Entry and Market Maker applications
Description	The purpose of this test is to ensure that the application can successfully place a new TMC order into previously created TMC with required details and receive an order identifier that can be used to track the order.
Notes	
Requirements	

#	Test Scenarios	Expected results
1.	The application will need to send a MO1 transaction for entering orders.	The transaction returns a successful *cstatus* and the order number of the order in *ordidt.* A BO5 broadcast is received with the full order details.

3.13.3 TMC Order Modification

Category	Supported Functionality for Order Entry and Market Maker applications
Description	The purpose of this test is to ensure that the application can successfully modify the previously placed TMC order with a new unique order number, series field and the bid/ask flag in the previously used equity stock, and receive a valid confirmation message.
	For the purpose of this test, participant may modify either price or quantity used in the previous sent TMC order.
Notes	
Requirements	To complete this test scenario, TMC order will need to have been placed in the market by the OI user who is connected for the conformance test.

#	Test Scenarios	Expected results
1.	The application will need to send a MO3 transaction on the previously entered TMC order, modifying either the price or quantity.	The transaction returns a successful *cstatus* and the order quantity before the amendment in *txstat.* A BO5 broadcast is received with the updated order details.

3.14 Order Cancellation

3.14.1 Single Order Cancellation

Category	Mandatory for Order Entry applications
Description	The purpose of this test is to ensure that the application can successfully cancel the previously placed order with a new unique order number, series field and the bid/ask flag in the previously used equity stock, and receive a valid confirmation that the order was cancelled.
	For the purpose of this test, participant can cancel their own orders.
Notes	
Requirements	To complete this test scenario, an order will need to have been placed in the market by the OI user who is connected for the conformance test.

#	Test Scenarios	Expected results
1.	The application will need to send a MO4 transaction on the previously entered orders. If one specific order is to be deleted, the following fields must be specified: • series (must be fully completed) • order_number_u • bid_or_ask_c.	The order is removed from the market.

3.14.2 Mass Order Cancellation

Category	Mandatory for Order Entry and Market Maker applications
Description	The purpose of this test is to ensure that the application can successfully cancel a group of previously placed orders and receive a valid confirmation that the orders were cancelled.
	For the purpose of this test, participants can cancel their own orders.
Notes	
Requirements	To complete this test scenario, several orders will need to have been placed in the market by the OI user who is connected for the conformance test.

#	Test Scenarios	Expected results
1.	The application will need to send a MO4 transaction to delete all or a subset of previously entered orders. The MO4 must not specify a specific order number.	The intended orders will be removed from the market.

3.15 Quote Management

3.15.1 Quote Request with Volume

Category	Supported Functionality for Order Entry applications, Optional for Market Maker applications
Description	The purpose of this test is to ensure that the application can successfully enter a quote request in the market place with the required details using the MC4 transaction, and receive a quote request confirmation via MI4 broadcast.
Notes	The participant has the option to use the MO36 or MO37 transaction for this test case.
Requirements	All series used in the transaction must exist in the same ME partition

#	Test Scenarios	Expected results
1.	Enter a quote into applicable markets via MC4	After a successful MC4 transaction, the quote request is sent to connected applications through the MI4 broadcast.

3.15.2 Market Maker response to Quote Request - Two Sided Price Quotation Entry

Category	Mandatory for Market Maker applications
Description	The purpose of this test is to ensure that the Market Maker application can successfully respond to the previously placed quote request.
Notes	The participant has the option to use the MO36 or MO37 transaction for this test case.
Requirements	All series used in the transaction must exist in the same ME partition. If there was no MC4 Quote Request sent, then the participant can enter a quote with MO36 or MO37 on the instrument

#	Test Scenarios	Expected results
1.	The application will need to send a MO36 transaction to respond to the previously placed quote request.	The transaction returns a successful *cstatus* and the order number of the quotes in *ordidt.*

3.15.3 Quote Replacement - Two Sided Price Quotation Entry

Category	Mandatory for Market Maker applications
Description	The purpose of this test is to ensure that the application can successfully replace the previously placed quotes and receive a valid confirmation message.
	For the purpose of this test, participant may modify either price or quantity used in the previous send quotes.
Notes	The participant has the option to use the MO36 or MO37 transaction for this test case.
Requirements	To complete this test scenario, quotes will need to have been placed in the market by the OI user who is connected for the conformance test.

#	Test Scenarios	Expected results
1.	The application will need to send a MO36 transaction to modify the previously placed quotes.	The transaction returns a successful *cstatus* and the new order number of the quotes in *ordidt.*

3.15.4 Quote Cancellation - Two Sided Price Quotation Entry

Category	Mandatory for Market Maker applications
Description	The purpose of this test is to ensure that the application can successfully cancel their quotes, and receive a valid confirmation that the quotes were cancelled.
	For the purpose of this test, participant can cancel their own quotes.
Notes	The participant has the option to use the MO36 or MO37 transaction for this test case.
Requirements	To complete this test scenario, quotes will need to have been placed in the market by the OI user who is connected for the conformance test.

#	Test Scenarios	Expected results
1.	The application will need to send an MO36 or MO37 transaction to cancel the previously modified quotes.	Quotes are removed from the market.

3.16 Market Maker Protection

3.16.1 Set Market Maker Protection Parameters

Category	Supported Functionality for Market Maker applications
Description	The purpose of this test is to ensure that the Market Maker application can set and change Market Maker Protection parameters for an underlying.
	ASX Market Maker (MM) Protection is aimed at preventing too many simultaneous trade executions on quotes provided by MMs, offering additional control of market risk.
Notes	If the counter reaches or exceeds a threshold level defined by the MMs, all remaining quotes in the underlying are automatically pulled from the market.
Requirements	

#	Test Scenarios	Expected results
1.	MM application will set the Exposure Time Interval, Quotation Frozen Time, Quantity Protection, Delta Protection and Include Futures parameters per underlying, using the DC87 transaction	Once the MM Protection Parameters are changed, the new values are disseminated in BU87 broadcast.

3.16.2 Handle Quotes deleted as a result of Market Maker Protection

Category	Supported Functionality for Market Maker applications
Description	The purpose of this test is to ensure that the MM application can handle the quotes deleted when MM protection is triggered by reaching or breaching the Quantity or Delta Protection threshold within the given Time Exposure Interval
	ASX Market Maker (MM) Protection is aimed at preventing too many simultaneous trade executions on quotes provided by MMs, offering additional control of market risk.
Notes	All the MMs quotes in the instrument series belonging to the affected underlying are automatically deleted by the system. Quotes are defined as being entered by the following transactions: MO36 and MO37.
Requirements	MM protection parameter set on the underlying to be used in this test.

#	Test Scenarios	Expected results
1.	Marker Maker application has to enter several quotes for Options over the same underlying.	The transaction returns a successful cstatus and the order number of the quotes in ordidt.
2.	MM trades in one quote, so that the Qty protection is exceeded.	The MM application will receive one BO5 Firm Order Book broadcast per deleted quote. The BO5 includes a change reason showing why the quote was deleted (i.e. either due to quantity protection or due to delta protection
		The MM also receives a BO38 broadcast as a notification that the protection has been triggered. The broadcast contains the calculated Quantity or Delta Protection values.

3.17 Trade Management

3.17.1 Trade Report Entry

Category	Supported Functionality for Order Entry and Market Maker applications
Description	The purpose of this test is to ensure that the OI application can successfully enter the trade report in to the market.
Notes	Application is able to use MO75, MO76 to enter trade reports by confirming the presence of trades with list of trade reports types in “Section 14.2 – Trade Report Types” of ASX Trade Introduction and Business Information. Order entry application and MM application can use this test.
Requirements	

#	Test Scenarios	Expected results
1.	Enter One Sided Trade Report (MO75) to report a trade between two different participants.	Application will receive a BO5 Firm Order Book Update broadcast for their side of the trade.
2.	Enter Two Sided Trade Report (MO76) to report Crossing trade.	Two directed Trade broadcasts (CB15) are sent to the participant showing relevant information of the trade report.
3	Enter an ITN Trade Report (MO76 with initial_trd_repot_c = 1.)	One directed Firm Order Book broadcast (BO5) and two directed Trade broadcasts (CB15) are sent to the reporting participant showing all details of the trade report. The initial trade is kept in ASX Trade and marked as a "Pending Trade".
4.	Enter Combination Trade Report (MO77).	Application will receive CB15 Directed Trade broadcasts for the combination trade report.

3.17.2 Trade Report Rejection

Category	Supported Functionality for Order Entry and Market Maker applications
Description	The purpose of this test is to ensure that the OI application handle receipt of Trade Report reject messages. Customer applications should be able to pass each of these scenarios. However, the exchange will accept a minimum of any 1 out of 3 scenarios below.
Notes	This case is Mandatory if 3.17.1 cases have been attempted.
Requirements	

#	Test Scenarios	Expected results
1	**Invalid Condition Code:** The application will need to trigger a reject message with Invalid Condition Code, e.g., Send condition code NS1 for NS2 trade report.	The application will receive a rejection message with Invalid condition code used.
2	**Invalid Price:** The application will need to trigger a reject message with Invalid trade report price, e.g., Send value of Ext Price outside of a spread for NX NBBO	The application will receive a Transaction aborted message for trade report with price outside the current spread.
3	**Invalid Instrument:** The application will need to trigger a reject message with Invalid instrument, e.g., Send an invalid Instrument details for the OI session.	The application will receive a rejection message with Invalid instrument.

3.17.3 Trade Report Cancellation

Category	Supported Functionality for Order Entry and Market Maker applications
Description	The purpose of this test is to ensure that the OI application can successfully cancel an Initial Trade Report (ITN) that was previously entered.
	Application should include a series, instance and a trade slip number, uniquely identifying the Initial Trade Report.
Notes	
Requirements	To complete this test scenario, an Initial Trade Report should be placed in the market by the OI user who is connected for the conformance test.

#	Test Scenarios	Expected results
1.	The application will need to send a CC87 transaction on the previously entered Trade report.	Trade Report will be removed from the market.

3.17.4 Short Sell Trade Report Entry

Category	Supported Functionality for Order Entry applications
Description	The purpose of this test is to ensure that the OI application can successfully enter a Short Sell trade report in to the market.
Notes	Application is able to use MO75, MO76 to enter trade reports by confirming the presence of trades with trade reports types in “Table 9.9.4.1 – Trade Report Types” of the OI manual".
Requirements	

#	Test Scenarios	Expected results
1.	Enter a trade report with SHL order condition set.	Expects SHL order condition appears as a condition code on the resultant trade and is visible to the user.

3.18 Reject handling

3.18.1 Order Reject Handling

Category	Mandatory for Order Entry and Market Maker applications
Description	The purpose of this test is that OI application able to handle receipt of reject messages. Customer applications should be able to pass each of these scenarios. However, the exchange will accept a minimum of any 1 out of 3 scenarios below.
Notes	
Requirements	

#	Test Scenarios	Expected results
1.	**Invalid Order Type:** The application will need to trigger a reject message with Invalid order type, e.g., Send an invalid exch_order_type_n for the OI session.	The application will receive a transaction aborted message with Invalid Order type.
2.	**Invalid Price:** The application will need to trigger a reject message with Invalid order price, e.g., Send an invalid value in premium_i for the OI session.	The application will receive a transaction aborted message with Invalid price.
3.	**Invalid Instrument:** The application will need to trigger a reject message with Invalid instrument, e.g., Send an invalid Instrument details for the OI session.	The application will receive a transaction aborted message with Invalid instrument.

4. ASX Assisted Conformance

4.1 Account Disable

Category	Mandatory for all applications
Description	The purpose of this test is to ensure that the application does not attempt multiple login retry attempts once their account is disabled.
Notes	
Requirements	

#	Test Scenarios	Expected results
1.	The application will perform a normal logon.	The participant will receive logon success status.
2.	CTS will disconnect the account which subsequently will logout the account and disconnect the session and then change the password.	The participant will receive a logout message with account disconnected in the central system. If forced to log out, or there is a drop in the connection, the application may make a first log-in try as soon as possible. If the first log-in attempt fails, the interval between the following attempts is greater than 5 seconds.
3.	The application will perform another logon and should ceases retry attempt after 3 invalid login attempt.	The application to cease connection after entering an invalid password 3 times.

4.2 Password Expiry

Category	Mandatory for all applications
Description	The purpose of this test is ensure an application can correctly manage an account that has reached its password expiry date. The participant will receive an error message upon login, relating to an account that requires a change in password, simulating a scenario of an expired login. This test will force the account to an expired state and the application should be able to re-establish connectivity and provide a new logon password thereafter.
Notes	It is expected that the application will refrain from any further connection attempts if it is unable to provide a valid logon message after the password expires.
Requirements	

#	Test Scenarios	Expected results
1.	The application will perform a standard logout. CTS will then set a new password which will be configured to expire immediately, and advise the customer what the new password is.	The participant will login with the new password set by CTS. Application will receive a Logon confirmation with OMNIAPI_INVALID_BEF_CHG_PWD. The application will only get a restricted access to the system until the password has been changed successfully.
2.	The application will then need to send a valid New Password using routine called omniapi_set_newpwd_ex.	The application will receive cstatus OMNIAPI_SUCCESS or positive then the password change was successful.

4.3 News and Text Messages

Category	Optional for all applications

Description	The purpose of this test is to confirm if the participant can receive and process News messages as sent by the ASX. Throughout the trading session, ASX may send messages to inform the market of market announcements and other information that may be relevant to the market participants.
Notes	The participant should receive these messages with their Order Entry and/or Market Data connections.
Requirements	

#	Test Scenarios	Expected results
1.	ASX will enter a text message and send it to the entire market.	The participant will receive a news and provides corresponding details to the ASX.

4.4 Market Data Validation

Category	Mandatory for Market Data applications
Description	The purpose of this test is to ensure that the application can correctly receive market data updates.
Notes	
Requirements	Participant must subscribe to market data

#	Test Scenarios	Expected results
1.	ASX enters orders on the equity market on a stock as agreed with the participant.	The participant will need to confirm the price and size of the orders as received by their feed.
	ASX to then trade existing order and question the participant on the current price and volume of the order.	Participant to answer what volume is left on the order and confirm the order state.

4.5 Processing/handling of price limit updates

Category	Supported Functionality for Order Entry and Market Maker applications
Description	The purpose of this test is to ensure that the participant aware of the price limits and handle any changes to the price limit.
Notes	
Requirements	

#	Test Scenarios	Expected results
1.	Application to check for processing/handling of price limit updates using BL51.	Application should confirm the current price limit on the selected stock.

4.6 Order Recovery

Category	Mandatory for all applications except Market Data
Description	The purpose of this test is to ensure that the application is able to recover orders when connection drops.
Notes	
Requirements	To complete this test scenario, the application will need to be disconnected from the development environment and ASX will enter some orders while the user is offline.

#	Test Scenarios	Expected results
1.	Application sending a MQ151 to recover orders placed while it was disconnected.	Application receives the list of orders placed during the trading day. Confirms details of order(s) placed by ASX while disconnected.

4.7 Handling of Purged Orders

Category	Mandatory for all applications except Market Data
Description	The purpose of this test is to ensure that the OI application can handle purged orders when ASX Series session state set to PURGE_ORDERS.
Notes	Orders meeting the purge criteria (e.g. too far from market, day only, expiry) are centrally inactivated. If the order is not reactivated the next day it is then deleted from the system the following night. In production the PURGE_ORDER session state is at the TSS (i.e. market level), not at the instrument series level.
Requirements	Application should pre-enter Day-only orders to trigger the order purge.

#	Test Scenarios	Expected results
1.	ASX will change the Trading Session state on series to PURGE_ORDERS.	A BO5 is sent to the application owning each order, informing that the order has been centrally inactivated along with the reason.

4.8 Validate Trades in Applicable Markets

Category	Mandatory for all applications except Market Data
Description	The purpose of this test is to ensure that the OI application can view their own trades, and BSP if requested by the participant.
Notes	
Requirements	Trades are entered before the application performs this test.

#	Test Scenarios	Expected results
1.	Application to utilise CB15 to confirm their own trades.	Application will confirm the trade slip number for their own trades. Application to confirm the Passive/Aggressive flag on their trades.

4.9 Trade Recovery

Category	Mandatory for all applications except Market Data
Description	The purpose of this test is to ensure that the OI application can recover trades from various partitions via missing deals query.
Notes	
Requirements	New trades are entered while application is offline to perform this test.

#	Test Scenarios	Expected results
1.	Application to send query CQ27 or CQ110 to retrieve the missing deals while the application logged off.	Application will confirm the trade slip number for the new trades entered during offline.

4.10 Trade Cancellation

Category	Mandatory for Order Entry and Market Maker applications
Description	The purpose of this test is to ensure that the application can receive a Trade Cancellation message from the trading platform. Trade cancellations initiated by the ASX will result in the application receiving a details of the cancellation.
Notes	
Requirements	

#	Test Scenarios	Expected results
1.	The application enters a Limit Buy order in the equity stock e.g., ABC. CTS will execute against your order to generate a trade that will either partially or fully fill the order. CTS will then cancel your trade.	The participant will receive a trade cancel update for their side of the trade.

4.11 Cancel on Disconnect

Category	Supported Functionality for Order Entry and Market Maker applications
Description	The purpose of this test is to ensure that the application can recover orders that have been cancelled due to Cancel on Disconnect functionality being triggered.
Notes	For information relating to the different types of Cancel on Disconnect which ASX Trade supports please refer to section 31 *Inactivate on Disconnect Functionality* within the ASX Trade Introduction and Business Information guide. To confirm your test user Cancel on Disconnect configuration, please contact CTS@asx.com.au.
Requirements	To complete this test scenario, the application will be disconnected and required to reconnect.

#	Test Scenarios	Expected results
1.	Utilising the Inactivate – all Cancel on Disconnect configuration, the application enters an order on a stock agreed with CTS. The application process will then need to be killed in order to trigger a disconnection event. Application reconnects and sends a MQ151 to recover order status of previously placed orders.	Orders placed are inactivated due to the Cancel on Disconnect event being triggered and the application recovers the status of the previously placed orders.
2.	Utilising the Inactivate – day-only Cancel on Disconnect configuration, the application enters a day order on a stock agreed with CTS.	Orders placed are inactivated due to the Cancel on Disconnect event being triggered and the application recovers the status of the previously placed orders.

	The application process will then need to be killed in order to trigger a disconnection event. Application reconnects and sends a MQ151 to recover order status of previously placed orders.	
3.	Utilising the immediate inactive – day-only Cancel on Disconnect configuration, the application enters an order on a stock agreed with CTS. The application is required to logout then reconnect and send a MQ151 to recover order status of previously placed orders.	Orders placed are inactivated due to the Cancel on Disconnect event being triggered and the application recovers the status of the previously placed orders.

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®