EMAPI Certification

Version 1.0
December 2009

Overview

The Certification tests are intended to cover a range of functional elements in order to demonstrate that a participant can successfully interact and perform as specified with the core Burgundy architecture. The mandatory testing is designed to ensure that a robust platform is in operation between participants and Burgundy.

The purpose of this document is to provide detailed test cases for the Burgundy EMAPI Certification. The Burgundy Certification package consists of several parts; Session Management, Information Certification, Trading Certification, Market Maker Certification and Technical Certification.

Session Management defines the initial interaction between a member firm's application and the Burgundy Trading System. The objective of this test is to establish that the connectivity and logon requirements have been implemented correctly.

Information Certification verifies the clients handling of subscriptions, reference data and that the full range of messages generated during a daily cycle of the Burgundy Trading system can be handled.

Trading Certification tests all order types supported by Burgundy against price, time and size conditions and state changes.

Market Maker Certification tests the full range of messages for quote management.

Technical Certification verifies the clients' ability to cope with technical requirements, as failover and load.

Testing Arrangements

Prior to undertaking Certification, all participants are required to have submitted a completed and signed Testing Agreement to Burgundy. Furthermore the participants are required to have confirmed connectivity to the Test environment. Participants requiring exemptions from certain test cases should have confirmed these with the Market Operation department in advance.

Certification Testing is arranged between the participant and Burgundy Market Operations. Whilst participants should provide three working days notice for test, short notice booking may be available, please contact support@burgundy.se.

Full Certification Testing is expected to take approximately x hours. The testing session may need to be extended where there is a requirement to re-run test cases. Testing will be available between 8 and 16.30 Monday to Friday.

Session Management

The Session Management Test assesses the ability to establish a connection and login. The Session Management Test forms the initial phase of the Certifications Tests. Connection to the system must be maintained throughout the full testing schedule. A failure to maintain connection at any point during the Certification Tests will constitute a failure of the Certification Test. Please note that if at any point a `heartbeat` message is not received after the `heartbeatInterval` periods have elapsed, the Customer Test environment will logoff the user and disconnect the session. This will also constitute a failure of the Certification Test.

Test Case	Activity	Participant Input	Burgundy Output	Mandatory / Optional
1	A TCP/IP connection should be established to the Customer Test environment using the connect method of the EMAPI.			M
2	The application should send a `TaxLoginReq` using the previously established connection.	`TaxLoginReq` Username – The User ID allocated by Technical Account Management. Member – The Trading Member the User belongs to. Password – This will be provided prior to the start of the test.	`TaxLoginRsp` logonAccepted = True clientHbtInterval = 15 maxLostHeartbeats = 4	M
3	Once a successful login has been established the `TaxHeartbeatReq` messages should be sent at the intervals defined within the `TaxLoginRsp`. These will be responded to with a `TaxHeartbeatReq` message.	`TaxHeartbeatReq` userData = Test	`TaxHeartbeatRsp` userData = Test	M
4	The user is required to change their password using the `ChangePasswordReq` message.	`ChangePasswordReq` MemberId – The Trading Member the User belong to. UserId - The User ID. OldPassword – This will be provided prior to the test. NewPassword – This will be provided prior to the test.		O
5	The user can query own Member.	`QueryOwnMemberReq`	`QueryOwnMemberRsp`	O
6	The user can query own Member users.	`QueryUsersReq`	`QueryUsersRsp`	O
7	The user can query all Members.	`QueryMemberReq`	`QueryMemberRsp`	O

		memberId = null		
8	The user must correctly logout of the system.	`TaxLogoutReq`	`SimpleRsp`	M

Information Certification

During the Information Certification the clients' ability to setup subscriptions and receive reference data is verified. Furthermore the full range of messages that can be generated during a daily cycle is generated and disseminated.

Order books will be provided at the start of the tests.

Test Case	Activity	Participant input	Burgundy Output
1	The user sets up a subscription for reference data.	`TaxSnapshotSubscribeReq` flow = 11 requestType = 1	`TaxSnapshotSubscribeRsp` + loads of reference data
2	The user subscribes to the System Status flow.	`TaxSnapshotSubscribeReq` flow = 20 requestType = 2	`TaxSnapshotSubscribeRsp`
3	The user subscribes to the System Events flow.	`TaxSnapshotSubscribeReq` flow = 18 requestType = 2	`TaxSnapshotSubscribeRsp`
4	The user subscribes to the Private Events flow.	`TaxSnapshotSubscribeReq` flow = 19 requestType = 2	`TaxSnapshotSubscribeRsp`
5	A message is sent by Market Operations to all.		`MarketMessage`
6	A message is sent to Testing Member from Market Operations.		`MarketMessage`
7	The user subscribes to public orders for OB1.	`TaxSnapshotSubscribeReq` flow = 4 requestType = 2 orderbookFilter = OB1	`TaxSnapshotSubscribeRsp`
8	The user subscribes to public trades for OB1.	`TaxSnapshotSubscribeReq` flow = 5 requestType = 2 orderbookFilter = OB1	`TaxSnapshotSubscribeRsp`
9	The order book enters Pre-trading.		`OrderbookStateChangeEvent`
10	Market Operations creates a trade.		`TradeEvent`

11	The order book enters Opening Call		OrderbookStateChangeEvent
12	The user subscribes to the MBL L2 flow.	TaxSnapshotSubscribeReq flow = 22 requestType = 2 orderbookFilter = OB1	TaxSnapshotSubscribeRsp
13	Market Operations enter an Order into the order book.		OrderEvent
14	Updates on the status of the auction are published.		AuctionEvent Imbalance = ? calculatedAuctionPrice = 0 matchedQuantity = 0
15	Market Operations enter another crossing Order into the order book.		OrderEvent
16	Updates on the status of the auction are published. MBL(5) is published. Aggregated to Equilibrium Price.		Imbalance = ? calculatedAuctionPrice = ? matchedQuantity = ? MarketByLevelEvent
18	The Auction uncrosses.	300@50	TradeEvent
19	The order book enters continuous trading with orders carried over from the auction call.		OrderbookStateChangeEvent OrderEvent MarketByLevelEvent
20	The user cancels all subscriptions for order book OB1.	TaxRemoveSubscriptionReq	
21	The user re-subscribes to public orders and requests a snapshot.	TaxSnapshotSubscribeReq flow = 4 requestType = 3 orderbookFilter = OB1	TaxSnapshotSubscribeRsp TaxStartSnapshot TaxEndSnapshot
22	Market Operations halts the trade in order book OB1.		OrderbookStateChangeEvent OrderbookTradeHaltEvent
23	Market Operations lift the trade halt.		OrderbookStateChangeEvent OrderbookTradeHaltEvent AuctionEvent
24	A replay of order book events is performed.	TaxReplayReq flow = 4	TaxReplayRsp TaxReplayStartEvent

		orderbookFilter = OB1	TaxReplayEndEvent
25	A replay of trade events is performed.	TaxReplayReq flow = 5 orderbookFilter = OB1	TaxReplayRsp TaxReplayStartEvent TaxReplayEndEvent
26	The user subscribes to public orders for order OB2.	TaxSnapshotSubscribeReq flow = 4 requestType = 3 orderbookFilter = OB2	TaxSnapshotSubscribeRsp
27	Market Operations enters some quotes in the order book.		QuoteEvent
28	The order books are closed.		OrderbookStateChangeEvent

Trading Certification

Trading certification tests the trading User functionality. The tests focus on management of trade reports and the different types of orders available on Burgundy.

Order books will be provided at the start of the tests.

Test Case	Activity	Participant input	Burgundy Output
	Pre-Trading		
1	Enter a Limit Order. Sell 10000@10	`OrderInsertReq`	`OrderInsertRsp`
2	Enter a Trade Report.	`TradeReportReq`	`TradeReportRsp`
3	Cancel Trade Report 2	`TradeCancelReq`	`TradeCancelRsp`
	Opening Call		
4	Enter a Dark Limit Order with minimum volume. Sell 500000@10 min. volume 1000	`OrderInsertReq`	`OrderInsertRsp` `OrderEventPrivate`
5	Enter a Limit Order valid for the opening call. Buy 500000@10	`OrderInsertReq`	`OrderInsertRsp` `OrderEventPrivate`
	Continuous Trading		`OrderEventPrivate` `TradeEventPrivate`
6	Enter a Limit Order which is valid for the session. Buy 100000@9	`OrderInsertReq`	`OrderInsertRsp` `OrderEventPrivate`
7	Enter an Iceberg Limit Order. Sell 10000@11 Open qty 1000	`OrderInsertReq`	`OrderInsertRsp` `OrderEventPrivate`
8	Enter an Order pegged to best bid. Buy 10000 Peg offset +2 Peg cap 11	`OrderInsertReq`	`OrderInsertRsp` `OrderEventPrivate`

9	Enter a Dark Order pegged to midpoint.
	Buy 500000
	Peg offset 0
	Peg cap 12
		`OrderInsertReq`	`OrderInsertRsp`
			`OrderEventPrivate`
			`TradeEventPrivate`

10	Enter a Fill And Kill Limit Order.	`OrderInsertReq`	`OrderInsertRsp` `OrderEventPrivate`

	Sell 25000@9		TradeEventPrivate
11	Enter a Fill Or Kill Limit Order. Sell 24000@9 Min volume 24000	OrderInsertReq	OrderInsertRsp OrderEventPrivate TradeEventPrivate
12	Update the price of Order 6. Price 9.5	OrderUpdateReq	OrderUpdateRsp OrderEventPrivate (O4) OrderEventPrivate (O6)
13	Cancel Order 8.	OrderCancelReq	OrderCancelRsp OrderEventPrivate
14	Enter a Limit Order valid for Closing Call. Buy 10000@10	OrderInsertReq	OrderInsertRsp OrderEventPrivate
15	Enter a Limit Order with validity Good Till Date Sell 10000@10 Date Today + 2	OrderInsertReq	OrderInsertRsp OrderEventPrivate
16	Enter a Dark Order with validity Good Till Cancel Buy 500000@10	OrderInsertReq	OrderInsertRsp OrderEventPrivate TradeEventPrivate
17	Enter an Order Pegged to Best Offer. Sell 10000 Peg offset -10 Peg cap 9	OrderInsertReq	OrderInsertRsp OrderEventPrivate
18	Enter a Trade Report.	TradeReportReq	TradeReportRsp
19	Market Operations enter a matching Trade Report.		TradeEventPrivate
20	Enter a Trade Report.	TradeReportReq	TradeReportRsp
	Termination		TradeEventPrivate
29	Update Order 25. Validity Good Till Date Date Today + 2	OrderUpdateReq	OrderUpdateRsp OrderEventPrivate
30	Cancel Trade Report 20	TradeCancelReq	TradeCancelRsp
	Post-Trading		
31	Cancel all remaining Orders	MassUpdateReq	MassUpdateRsp OrderEventPrivate

Market Maker Certification

Market Maker certification tests the management of quotes.

Order books will be provided at the start of the tests.

Test Case	Activity	Participant input	Burgundy Output
	Pre-Trading		
1	Enter a two-sided Quote in OB1. Buy 500000@1.00 Sell 500000@1.10	`QuoteInsertReq`	`QuoteInsertRsp` `QuoteEventPrivate`
2	Replace Quote 1. Buy 500000@1.05 Sell 500000@1.10	`QuoteInsertReq`	`QuoteInsertRsp` `QuoteEventPrivate`
3	Cancel Quote 1	`QuoteInsertReq`	`QuoteInsertRsp` `QuoteEventPrivate`
	Opening Call		
4	Enter two-sided Quotes in OB1 and OB2. OB1 Buy 500000@1.00 Sell 500000@1.10 OB2 Buy 500000@0.01 Sell 500000@0.02	`QuoteInsertReq`	`QuoteInsertRsp` `QuoteEventPrivate`
5	Replace Quote 4. OB1 Buy 500000@0.95 Sell 500000@1.00 OB2 Unchanged	`QuoteInsertReq`	`QuoteInsertRsp` `QuoteEventPrivate`
6	Cancel Quote 4	`QuoteInsertReq`	`QuoteInsertRsp` `QuoteEventPrivate`
	Continuous Trading		
7	Enter multi level two-sided Quotes in OB1 and OB2. OB1 Buy 500000@1.00 Buy 500000@0.95 Sell 500000@1.10 Sell 500000@1.15 OB2 Buy 500000@0.01 Buy 500000@0.02	`QuoteInsertReq`	`QuoteInsertRsp` `QuoteEventPrivate`

	Sell 500000@0.03 Sell 500000@0.04		
8	Replace Quote 7 OB1 Buy 500000@1.00 Buy 500000@0.95 Sell 500000@1.10 Sell 500000@1.15 OB2 Cancel all prices.	`QuoteInsertReq`	`QuoteInsertRsp` `QuoteEventPrivate`
9	Cancel Quote 4	`QuoteInsertReq`	`QuoteInsertRsp` `QuoteEventPrivate`

Technical Certification

The technical certification is aimed at verifying fail-over and performance during load.

Burgundy will have load testing in the Stage environment Monday, Wednesday and Friday at 14-15 CET. It will be in the same order books as used for light background load. The load test will be run in four (4) steps. Starting at a low level and increasing up to 1000 orders. Between each step we will make room for clean up and recovery. First test step will be 15 minutes long, next step will last for 10 minutes and the last 2 steps will be 2 minutes long. A 2 minute period with 1000 orders per second will generate 120 000 orders and 12 000 trades. This test shall simulate a large opening or closing call, or important market news that would make the market move rapidly.

Load test schedule

- 14:00-14:15 (15 min load test) 250 orders per second with a trade ration 1:10, spread out on order books according to separate document; prices should be as realistic as possible.
- 14:15-14:25 (10 min) No load
- 14:25-14:30 (5 min) 500 orders per second (same set up as above)
- 14:30-14:40 (10 min) No load
- 14:40-14:42 (2 min) 750 orders per second (same set up as above)
- 14:42-14:58 (16 min) No load
- 14:58-15:00 (2 min) 1000 orders per second (same set up as above)

When performing the failover test case you should only connect to one site at a time. This is to make it possible for us to verify that you have been able to do reconnect and recovery.