Confidential

Business Requirements

OCCTR – MQ Ops DB Monitoring

Issue 1.0

Issue date: May 22, 2013

Produced by:
International Securities Exchange, Inc.
60 Broad Street, New York NY 10004
www.iseoptions.com

Table of Contents

About this document

This document provides the business requirements monitoring OCCTR – MQ connection status in the Ops Dashboard.

Users of this document

The audience for this document includes ISE employees who are responsible for delivering trading applications and supporting the day to day trading and surveillance:

- Development
- Market Operations
- Software Quality and Management
- Business Development
- Product Management
- Technology Member Services

Summary of changes

Issue	Date	Author	Change
0.1	May 22, 2013	M.DiBenedetto	First draft
1.0	May 30, 2013	M. DiBenedetto	Reviewed with Development and App Support.

Reviewers and approvers

This document was subject to review and approval.

The records of reviews and approvals are stored on file with the ISE Product Management team.

Reviewers

This document was reviewed by:

Name	Title	Date
Rob Klotz		5/24/2013
Christian Velez		5/30/2013

Approvers

This document was approved by:

Name	Title	Date

Chapter 1. Introduction

1.1. Scope

This document covers the software requirements for monitoring the connection status between OCCTR and MQ using the Operations Dashboard. This document will not cover the requirements for monitoring connection and queue status for MQ to OCC; this will be covered in a phase 2 requirements document.

1.2. Planned Benefits

Application support will have near real-time monitoring of the connection status between OCCTR and MQ. ASG will no longer need to look for an email to be alerted to connection or data flow issues. OCCTR application status is currently monitored on the Ops DB, adding the connection status to MQ will give App Support a consolidated view of health of the application in a single place.

1.3. Delivery Requirements

Phase 1 of the OCCTR – MQ monitoring enhancement is being requested for Q4 2013, sometime after the 7.0 release but before 8.0.

1.4. Priority

The priority is high.

1.5. Related Documents

Chapter 2. Overview

2.1. Current behavior

OCCTR uses MQ to send trades to the OCC. MQ is a third party application provided by IBM that the OCC uses to receive trade data from each exchange. MQ consists of two applications, the MQ Manager and the MQ Channels. MQ Manager must be up and running before the MQ channels are brought up by UC4 to connect to the OCC.

Throughout the day, when trades are executed, OCCTR sends the trade data to MQ, where it is queued up and eventually sent to the OCC via the MQ channels. MQ is constantly receiving trade data from OCCTR while sending queued up trades to the OCC.

When the system is rebooted, both MQ components (MQ Manager and MQ Channels) are brought down. Application support has a scheduled UC4 job to start the MQ channels; however the MQ channels require the MQ manager to be running before they are started. Unfortunately there is no automated process to bring up the MQ Manager (this should be addressed in a separate requirement and is out of scope for this BRD), this may result in MQ not being restarted properly leading to issues with OCCTR connecting to MQ to send trade data to the OCC.

The OCCTR to MQ connection has experienced several issues in the past that have resulted in disruptions to the flow of trade data from ISE to the OCC. The monitoring of the connection status between OCCTR and MQ must be improved to assist App Support in situations like those described above.

2.2. Current Problem

Application support uses the operations dashboard to monitor the health of various applications both connected to the Core and residing outside of the core. Currently, the only method of monitoring the status of the MQ Manager and MQ Channels is using the MQ Explorer application. In addition to using MQ Explorer, a UC4 job exists that runs a command each morning to check the IBM MQ log for any errors, if an error is found then an email is sent to App Support. There are two issues with the current monitoring solution:

1. Application status monitoring should be consolidated to the ops dashboard. App support should not need to look for a morning email or launch a separate application to actively check the status of the MQ Manager. Adding the status to the ops dashboard would allow for passive monitoring and alerting for interruption in connectivity or data flow.
2. A morning check of the MQ log for errors is sufficient for ensuring there are no issues at the start of day, however the MQ log and OCCTR to MQ connectivity status should continue to be checked throughout the day for any errors or warnings that may occur.

2.3. Proposed Solution

OCCTR to MQ connection status and data flow statistics should be available on the Ops Dashboard.

Report OCCTR connection status to MQ on the Ops Dashboard, and change status if there is an issue connecting at start up, or throughout the day if no new messages are sent in a configurable timeframe. The connection status should initially be checked prior to start of trading, as early as possible, preferably around 7:30 AM at the latest.

Transmitted message count should be available on the Ops DB as well.

Chapter 3. Functional Requirements

3.1. Interfaces

None

3.2. Core

None

3.3. Trading Tools

Ops Dashboard

Connection Status from OCCTR to MQ should factor into displayed status of OCCTR application. I.e. if OCCTR is having issues sending data to MQ, OCCTR should turn red to alert the App Support user.

Sent message count should be visible on the Ops Dashboard under OCCTR application.

3.4. OCCTR

OCCTR will need to be updated to send connection status information to the dashboard. This status should be sent well before trading opens for the day. Two proposed solutions for this included using FX products to ascertain OCCTR to MQ status (since FX products open before equity options), or sending a test message to MQ. Additionally, at startup, OCCTR should check the MQ log to ensure that the MQ channel is up. If the MQ channel is not up then the dashboard status should go red.

Comment [MD1]: This will be included if the command to read the MQ log is on the same box as OCCTR, otherwise this will fall into a future enhancement.

OCCTR should also be updated to send a status change to the dashboard if a message has not been successfully sent to MQ for a configurable amount of time. This configured interval should be specified by application support at time of deployment.

Chapter 4. Non-functional Requirements

4.1. Operational Deployment

There are no explicit operational deployment requirements for this function.

4.2. Maintainability

The functionality must be built with an on/off configurability.

4.3. Reliability

There are no explicit reliability requirements for this function.

4.4. Availability

There are no explicit availability requirements for this function.

4.5. Auditability

None

4.6. Documentation

There are no explicit documentation requirements for this function.

4.7. Acceptance Criteria

No specific acceptance criteria

4.8. Rules

No rule change required

4.9. Future Enhancements

Phase 2 of this enhancement will include MQ monitoring. This includes queue status and connection to OCC status.

Phase 2 will include functionality to start MQ Manager and MQ Channels from the Ops DB.