

# Nasdaq Calypso Data Uploader Integration Guide

Version 15 – Version 18

Revision 39.0 May 2025 Approved



#### Copyright © 2025, Nasdaq, Inc. All rights reserved.

All content in this document is owned, or licensed, by Nasdaq, Inc. or its affiliates ('Nasdaq'). Unauthorized use is prohibited without written permission of Nasdaq.

While reasonable efforts have been made to ensure that the contents of this document are accurate, the document is provided strictly "as is", and no warranties of accuracy are given concerning the contents of the information contained in this document, including any warranty that the document will be kept up to date. Nasdaq reserves the right to change details in this document without notice. To the extent permitted by law no liability (including liability to any person by reason of negligence) will be accepted by Nasdaq or its employees for any direct or indirect loss or damage caused by omissions from or inaccuracies in this document.

# **Document History**

| Revision | Published      | Summary of Changes   |
|----------|----------------|--|
| 1.0      | October 2013   | First edition for version 3.1.19.                                      |
| 2.0      | March 2014     | Second edition for version 3.1.23.                                     |
| 3.0      | September 2014 | Third edition – Added information for Data Uploader engine.            |
| 4.0      | October 2014   | Fourth edition – Updates for version 14.1 of Core Calypso.             |
| 5.0      | July 2015      | Fifth edition – Updates for version 4.0.0.                             |
| 6.0      | October 2016   | Sixth edition – Updates for version 4.13.0.                            |
| 7.0      | April 2017     | Seventh edition – Updates for version 5.2.0.                           |
| 8.0      | May 2017       | Eighth edition – Updates for version 5.3.0.                            |
| 9.0      | July 2017      | Ninth edition – Added information for Uploader Import Message engine.  |
| 10.0     | February 2018  | Tenth edition – Added Web Service information.                         |
| 11.0     | November 2018  | Eleventh edition – Updates for version 7.3.1.                          |
| 12.0     | January 2019   | Edition 12 – Updated for version 7.5.0.                                |
| 13.0     | May 2019       | Edition 13 – Updated for version 7.6.7.                                |
| 14.0     | June 2019      | Edition 14 – Added supported IBM MQ version.                           |
| 15.0     | January 2020   | Edition 15 – Added Mapping of Interfaces with Calypso Upload Document. |
| 16.0     | February 2020  | Edition 16 – Added lost messages when persistMessages=None.            |
| 17.0     | February 2020  | Edition 17 – Updated for version 8.1.1.                                |



| Revision | Published      | Summary of Changes  |
|----------|----------------|---|
| 18.0     | May 2020       | Edition 18 – Added information about running multiple instances of the Import Message engine.                                 |
| 19.0     | May 2020       | Edition 19 – Updated for version 8.4.0.   |
| 20.0     | September 2020 | Edition 20 – Updated for version 9.0.3. Added Data Uploader Web Portal.   |
| 21.0     | September 2020 | Edition 21 – Updated for version 9.1.7.   |
| 22.0     | November 2020  | Edition 22 – Updated for version 9.2.3.   |
|          |                | Documentation on Data Uploader Web Services is now in the Calypso Data Uploader Developer's Guide.                            |
| 23.0     | December 2020  | Edition 23 – Updated for version 9.3.6.   |
|          |                | Introduction of calypsoServicesInterfacesMs.bat for Data Uploader REST API and Web Portal.                                    |
| 24.0     | May 2021       | Edition 24 – Updated for version 9.8.0.   |
|          |                | Updated Web Portal Login window.  |
| 25.0     | October 2021   | Edition 25 – Documentation on Data Uploader Web Portal is now in the Calypso Data<br>Uploader Web Portal Guide.               |
| 26.0     | February 2022  | Edition 26 – Version 17.0 – Technical release only.   |
|          |                | As of version 17.0, REST APIs documentation is provided under the REST APIs page on the Documentation Portal and under /docs. |
| 27.0     | April 2022     | Edition 27 – Added support for CCDT feature for IBM MQ.   |
| 28.0     | September 2022 | Edition 28 – Added support for Kafka  |
| 29.0     | November 2022  | Edition 29 – Added Reprocessing Bloomberg VCON Messages for Missing Bonds   |
| 30.0     | March 2023     | Edition 30  |
| 31.0     | May 2023       | Edition 31 – Added email generation to scheduled task EXPORT_QUOTES.  |
| 32.0     | June 2023      | Edition 32 – Added support for SavePriceFromTradePriceReport Server Side Access<br>Permissions                                |
| 33.0     | July 2023      | Edition 33 - Added Calypso Mapping Service Support for data translation.  |
| 34.0     | August 2023    | Edition 34 - Enhanced JMS Sender not set MQ header properties on Messages   |
| 35.0     | February 2024  | Edition 35 – Compatibility with version 18.   |



| Revision | Published     | Summary of Changes   |
|----------|---------------|--|
| 36.0     | April 2024    | Edition 36 – Added UploadPoolSequencePolicy instead of UploaderImportMessageEngine<br>as Event pool policy.<br>Added support for CoppClark Scheduled Task. |
| 37.0     | August 2024   | Added support for popup window for Data Uploader window<br>Added support for search function in Calypso Mapping Window.                                    |
| 38.0     | February 2025 | Added support for Multiple instances of Upload Import Message Engine.  |
| 39.0     | May 2025      | Added domain "SecFinance.EventTypeActionExternalRefName".  |

# This document describes how to install, configure, and run the Calypso Data Uploader to import various types of data into Calypso.

NOTE: The Calypso License to use this Calypso Integration Module does not include a license for any third-party data services to which this module can interface. Clients are responsible for contracting with the appropriate third-party data service(s) prior to using this Calypso Integration Module.



# Table of Contents

| 1 | Data  | Uploade                          | er Overview   | 7                        |  |  |  |  |  |  |  |
|---|-------|----------------------------------|---|--------------------------|--|--|--|--|--|--|--|
|   | 1.1   | Feature                          | S   | 7                        |  |  |  |  |  |  |  |
|   | 1.2   | Input Sources                    |   |                          |  |  |  |  |  |  |  |
|   | 1.3   | Supported Data                   |   |                          |  |  |  |  |  |  |  |
|   | 1.4   | Operations Overview              |   |                          |  |  |  |  |  |  |  |
|   |       | 1.4.1<br>1.4.2<br>1.4.3<br>1.4.4 | Manual Uploading<br>Handling of Lifecycle Events<br>General Notes<br>Known Issues | .10<br>.10<br>.11<br>.11 |  |  |  |  |  |  |  |
|   | 1.5   | Custom                           | izing the Data Uploader   | . 11                     |  |  |  |  |  |  |  |
| 2 | Insta | llation a                        | nd Configuration  | 12                       |  |  |  |  |  |  |  |
|   | 2.1   | Installat                        | ion   | .12                      |  |  |  |  |  |  |  |
|   |       | 2.1.1                            | Property Files  | .13                      |  |  |  |  |  |  |  |
|   |       | 2.1.2                            | Database Upgrade  | .13                      |  |  |  |  |  |  |  |
|   |       | 2.1.3                            | SWAPSWIRE_MAPPING Table Migration   | .13                      |  |  |  |  |  |  |  |
|   |       | 2.1.4                            | Sec Finance I ifecvele Events   | . 14<br>14               |  |  |  |  |  |  |  |
|   |       | 2.1.6                            | Access Permissions  | .14                      |  |  |  |  |  |  |  |
|   |       | 2.1.7                            | Data Authorization  | .14                      |  |  |  |  |  |  |  |
|   | 2.2   | Calypso                          | Mapping Window  | .16                      |  |  |  |  |  |  |  |
|   |       | 2.2.1                            | Reverse Default Functionality   | . 17                     |  |  |  |  |  |  |  |
|   |       | 2.2.2                            | Access Permissions  | . 17                     |  |  |  |  |  |  |  |
|   |       | 2.2.1                            | Calypso Mapping Audit   | . 17                     |  |  |  |  |  |  |  |
|   |       | 2.2.2                            | Mapping of Interfaces with Calypso Upload Document                                | .18                      |  |  |  |  |  |  |  |
|   | 2.3   | Workflo                          | w Configuration   | 23                       |  |  |  |  |  |  |  |
|   |       | 2.3.1                            | GATEWAYMSG Message Workflow   | 23                       |  |  |  |  |  |  |  |
|   |       | 2.3.2                            | KickOff/CutOff Config (for Local Mode Only)                                       | 27                       |  |  |  |  |  |  |  |
|   |       | 2.3.3                            |   | 29                       |  |  |  |  |  |  |  |
|   | 2.4   | .4 Task Station Setup            |   |                          |  |  |  |  |  |  |  |
|   | 2.5   | 5 Pricing Environment            |   |                          |  |  |  |  |  |  |  |
|   | 2.6   | Uploade                          | er Import Message Engine Configuration  | 33                       |  |  |  |  |  |  |  |
|   |       | 2.6.1                            | Engine Configuration  | 33                       |  |  |  |  |  |  |  |
|   |       | 2.6.2                            | Domain values   | 34                       |  |  |  |  |  |  |  |
|   |       | 2.0.3                            | Engine Palameters   | 34                       |  |  |  |  |  |  |  |
|   |       | 2.6.5                            | Property File   | 35                       |  |  |  |  |  |  |  |
|   |       | 2.6.6                            | Sender Engine Configuration   | 37                       |  |  |  |  |  |  |  |
|   |       | 2.6.7                            | Running Multiple Instances of the Import Message Engine                           | 39                       |  |  |  |  |  |  |  |
|   |       | 2.6.8                            | Using the Update Manager Engine   | 43                       |  |  |  |  |  |  |  |
|   | 2.7   | Using th                         | ne Data Uploader Engine   | 46                       |  |  |  |  |  |  |  |
|   |       | 2.7.1                            | Property File   | 46                       |  |  |  |  |  |  |  |
|   |       | 2.7.2                            | Launching the Data Uploader Engine  | 47                       |  |  |  |  |  |  |  |
|   | 2.8   | Using th                         | ne Scheduled Task DATA_UPLOADER   | 49                       |  |  |  |  |  |  |  |



|   | 2.9    | Resourc                    | ces Files                           | 50       |  |  |  |  |  |
|---|--------|----------------------------|-------------------------------------|----------|--|--|--|--|--|
|   | 2.10   | Dates F                    | Format                              | 50       |  |  |  |  |  |
|   | 2.11   | Fees Ma                    | anagement                           | 50       |  |  |  |  |  |
| 3 | Opera  | ations                     |                                     | 51       |  |  |  |  |  |
|   | 3.1    | Require                    | ments                               | 51       |  |  |  |  |  |
|   | 3.2    | Handlin                    | g Blocked Messages                  | 51       |  |  |  |  |  |
|   | 3.3    | Task Sta                   | ation Errors                        | 51       |  |  |  |  |  |
|   | 3.4    | Trade A                    | Allocations                         | 51       |  |  |  |  |  |
| 4 | IBM N  | IQ and                     | JMS Support                         | 56       |  |  |  |  |  |
| 5 | Sourc  | e Mess                     | age Workflow                        | 62       |  |  |  |  |  |
|   | 5.1    | Impleme                    | entation                            | 62       |  |  |  |  |  |
|   | 5.2    | Benefits                   |                                     |          |  |  |  |  |  |
|   | 5.3    | Task Station Configuration |                                     |          |  |  |  |  |  |
| 6 | Expor  | ting Qu                    | lotes                               | 71       |  |  |  |  |  |
|   | 6.1    | Configu                    | Jration                             | 71       |  |  |  |  |  |
|   |        | 6.1.1<br>6.1.2             | Domain Value<br>Scheduled Task      | 71<br>71 |  |  |  |  |  |
|   | 6.2    | Limitatio                  | on and Alerts                       | 73       |  |  |  |  |  |
|   |        | 6.2.1                      | Quotes Records                      | 73       |  |  |  |  |  |
|   | 6.3    | File Split                 |                                     |          |  |  |  |  |  |
|   | 6.4    | Typical                    | Use Cases                           | 74       |  |  |  |  |  |
|   |        | 6.4.1                      | Examples and Notes                  | 74       |  |  |  |  |  |
| 7 | Client | : Chann                    | el Definition Table (CCDT) - IBM MQ | 75       |  |  |  |  |  |
|   | 7.1    | Queue I                    | Manager Groups                      | 77       |  |  |  |  |  |
| 8 | Kafka  | Suppor                     | rt                                  | 82       |  |  |  |  |  |



# 1 Data Uploader Overview

The Calypso Data Upload Framework (CDUF) simplifies importing data into Calypso. It supports many product types as well as certain reference data.

CDUF supports the following trade lifecycle events:

- New
- Amend
- Terminate (Partial and Full Termination)
- Novate (PO, CounterParty, and Incoming Party. Partial and Full Novation)
- Cancel
- Undo Terminate
- Exercise

In addition to Trades, CDUF also supports uploading:

- Books
- Legal Entities
- Corporate Actions

## 1.1 Features

- Full Lifecycle support This supports the full lifecycle for trades, from New to Terminated. It also supports custom actions defined in workflows.
- Calypso BO Message-based framework This allows for extensibility by allowing customers to add their own validations via rules. This can also be customized to meet customer needs.
- Trade Lifecycle Ordering Trade lifecycle actions are played back in the same order as they are received. If one of the earlier actions cannot be applied due to a problem, all subsequent actions on the trade are blocked as well.
- Error Reporting All errors are available in the Task Station for review by your operations staff, or by Calypso operations staff on the SaaS implementations.
- Replay Messages Trades in errors are blocked as a message and can be played back when the underlying system error is corrected.
- Native XML format The Data Uploader has a native XML format. It supports multiple types of Keywords and Fees.
- Acknowledgement Framework In addition to Task Station integration, there is a publisher framework to interface with electronic Trade Capture systems, such as MarkitWire, to send acknowledgements and trade capture status to these systems.



- Calypso Termination API The Data Uploader uses the Calypso Termination Extensions. As a result, it supports any extensions created by users and allows for configurable actions for Termination and Novation.
- Validation Framework All validations are done via Calypso BO Message rules. This allows a single code-base
  to be used for Validations and Actions across different incoming formats. For example, the Data Uploader
  supports inputs from MarkitWire, ICELink, CSV, as well as the Calypso Upload Document format (XML). All
  business validations are performed in one place.
  - The required effort to interface with a new trade capture is reduced to translating the new format to the Calypso Upload Document format (assuming the native XML supports those features).
- Web Services Please refer to the Calypso Data Uploader Developer's Guide.
- Extensions The Framework supports extensions as follows:
  - New rules to handle customer-specific business rules.
  - Implementation teams can supply their own CSV formats convertors to create CSV upload files.

Please refer to the Calypso Data Uploader Developer's Guide.

As of version 17.0, REST APIs documentation is provided under the REST APIs page on the Documentation Portal and under /docs.

## 1.2 Input Sources

The following input sources are currently supported by the Calypso Uploader Framework

• XML - Standardized Calypso Upload XML, also known as the Calypso Upload Document (CUD) format.

#### () [NOTE: The Calypso Upload Document Format differs from Calypso ML format.]

- MarkitWire A new MarkitWire interface for IRD Swaps trades. Refer to the Calypso MarkitWire Integration Guide.
- ICELink Credit Derivatives Affirmation Interfaces.
- CSV Support for limited products in CSV format. Extendable by the implementation team.

() [NOTE: The CSV format supports most common use cases. For complex objects and full coverage, please use the XML format. The XML format should be used for high performance integration]

## 1.3 Supported Data

The list of supported XML-format data can be found in "<calypso home>/docs/calypso-datauploader/Calypso DataUploader Coverage.xls".

INOTE: All features may not be covered for all data. Please review the formats and samples for supported features, and raise enhancement requests for any gaps]



| ۲ | docs                                |  |  |  |  |  |  |  |
|---|-------------------------------------|--|--|--|--|--|--|--|
|   |                                     |  |  |  |  |  |  |  |
|   |                                     |  |  |  |  |  |  |  |
|   | Name                                |  |  |  |  |  |  |  |
|   | 鷆 format                            |  |  |  |  |  |  |  |
|   | 📗 samples                           |  |  |  |  |  |  |  |
|   | \mu schema                          |  |  |  |  |  |  |  |
|   | 🕙 Calypso DataUploader Coverage.xls |  |  |  |  |  |  |  |
|   | 📄 ReadMe.txt                        |  |  |  |  |  |  |  |
|   |                                     |  |  |  |  |  |  |  |

- The "format" folder contains a description of the format for all supported data
- The "sample" folder contains sample import files for all supported data

## 1.4 **Operations Overview**

The Data Uploader can be invoked by the File Watcher application (through the Data Uploader engine), by a Calypso engine, or manually using the scheduled task DATA\_UPLOADER.

The File Watcher monitors a specified folder at a specified interval, and when a new file is found, invokes the Data Uploader process.

#### [NOTE: The "watched" file directory and the interval between checks is user-configurable]

Input files must be comma separated values (.CSV) or in the Calypso Upload Document (CUD) XML format. Examples of Calypso Upload Documents for data are available in "<calypso home>/docs/calypsodatauploader/samples".

For CSV- and XML-formatted file, the file name must start with the data name. For example, the name of a CSV or XML file containing FX trades must begin with FX: "FX\_20100226.csv", "FX\_1Q2010.xml", etc. There are no restrictions on the remaining characters of the filename.

The file extension indicates the format of the contents, CSV for a file containing data in comma separated values format, or XML for a file containing data in the Calypso Upload Document format.

When a CSV file is loaded, the application converts it to the CUD using XSLT Transformation. The XML files are next processed into BO Messages, where they are validated before being processed into Trades and uploaded to the database.

For trade data, the **External Ref ID** field must be unique. The **Bundle Type** and **Bundle Name** are generated on the fly. For example, in a file of fills for one order on an ECN, the **Bundle Name** can be the order number; the **External Ref ID** will represent the **ECN Trade ID**.



## 1.4.1 Manual Uploading

To manually upload trades, launch the Data Uploader engine that runs the File Watcher application, and copy an input file into the folder specified in the configuration file. The File Watcher invokes the Calypso Data Uploader at the end of the current wait interval. You can also use the scheduled task DATA\_UPLOADER.

The Data Uploader processes the input file and creates an XML result file having the same base filename (e.g., FX\_1Q2010) in the same directory. The result file contains details on the uploads, and depending on the outcome of that processing, either has a ".finished" or ".failure" file extension, to indicate success or failure, respectively.

# () [NOTE: Certain Calypso engines internally process incoming data into CUD format, making it ready for upload to the database without needing external files]

| Lifecycle Event      | Trade Data   | Keyword Action                                       | Fees Action                                     |
|----------------------|--|--|---|
| NEW                  | Save All   | Save All   | Save All  |
| AMEND                | OVERWRITE  | Overwrite Existing<br>Keywords.<br>Add New Keywords. | Remove existing Fees and replace them from XML  |
| CANCEL               | Perform Basic Validation, do not<br>Overwrite other trade data.  | Overwrite Existing<br>Keywords.<br>Add New Keywords. | Keep existing Fees and add new fees from XML    |
| TERMINATE            | Perform Basic Validation, do not<br>Overwrite other trade data.<br>Refer to the TERMINATION block<br>for termination tags to perform the<br>Termination.         | Overwrite Existing<br>Keywords.<br>Add New Keywords. | Keep existing Fees and add<br>new fees from XML |
| PARTIAL<br>TERMINATE | Perform Basic Validation, do not<br>Overwrite other trade data.<br>Refer to the TERMINATION block<br>for termination tags to perform the<br>Partial Termination. | Overwrite Existing<br>Keywords.<br>Add New Keywords. | Keep existing Fees and add new fees from XML    |
| NOVATE               | Perform Basic Validation, do not<br>Overwrite other trade data.<br>Refer to the NOVATION block for<br>novation tags to perform the<br>Novation.                  | Overwrite Existing<br>Keywords.<br>Add New Keywords. | Keep existing Fees and add<br>new fees from XML |

## 1.4.2 Handling of Lifecycle Events



| Lifecycle Event   | Trade Data   | Keyword Action                                       | Fees Action   |  |  |  |  |
|-------------------|--|--|---|--|--|--|--|
| PARTIAL<br>NOVATE | Perform Basic Validation, do not<br>Overwrite other trade data.                            | Overwrite Existing<br>Keywords.                      | Keep existing Fees and add new fees from XML                            |  |  |  |  |
|                   | Refer to the NOVATION block for novation tags to perform the Partial Novation.             | Add New Keywords.                                    |   |  |  |  |  |
| UNDO<br>TERMINATE | Perform Basic Validation, do not<br>Overwrite other trade data.                            | Overwrite Existing<br>Keywords.<br>Add New Keywords. | Keep existing Fees. Ignore<br>Fees from XML. Remove<br>TERMINATION_FEES |  |  |  |  |
| Basic Validation  | Validate that the book and counterparty of the Action matches those of the existing trade. |  |   |  |  |  |  |

## 1.4.3 General Notes

- Adding an (non-unique) index for the External Reference column of the Trade Table can improve performance.
- Adding an index on the Attribute Type + Attribute Value columns of the Message Attribute table can improve performance.

## 1.4.4 Known Issues

• When a Basis Swap is Partially Terminated in Calypso, the first Rate for the newly-created residual trade (i.e. the child trade) is cleared. This situation also applies to Cash Settlement information for Cap/Floor trades.

This is a core Calypso issue and not a problem specific to other interfaces.

# 1.5 Customizing the Data Uploader

Please refer to the Calypso Data Uploader Developer's Guide for information on customizing the Data Uploader.

Important Note - The Data Uploader XSD is internal to the Data Uploader and cannot be used as an API. Between major versions, we cannot guarantee backward compatibility.

For example, in 7.0.0 onwards we have changed the "trade id" related tags from type "int" to "long" and when migrating to version 7.0.0, you need to make sure that your custom code is changed accordingly.

# 2 Installation and Configuration

# 2.1 Installation

The Data Uploader is installed as part of the Calypso Installer when you select the "Data Uploader" optional module:

| A Setup - Calypso 14.0.0.15.SP1  |
|--|
| Select Components<br>Which components should be installed?   |
| Select the components you want to install; clear the components you do not want to install. Click Next when you are ready to continue. |
| 👘 📝 🚼 Base Installation (software required for all installations, includes Main Entry) 🮯   |
| 😥 🐨 📝 📁 Solutions (pre-packaged options for installing standard configurations) 🧕  |
| 😥 👘 🗐 📁 Interfaces (Please consult with your Calypso Account Executive for licensing questions)  |
| 🚊 🐨 📝 🧔 Optional Modules (Please consult with your Calypso Account Executive for licensing questions)                                  |
| 🔄 📩 Calypso Business Services Layer (CBSL)   |
| 🕅 📩 CCP eClearing Web Portal   |
| 🔄 📩 CMF OTC Clearing   |
|  |
| 🔄 📩 CVA  |
| 🔲 📩 Datasynapse Compute Grid Integration 🧔   |
| 🔽 💦 Data Uploader  |
| 🖂 📩 Liquidity  |

> Please refer to the Calypso Installation Guide for complete details on the Calypso Installer.

If you are installing a Calypso Upgrade package instead, the instructions are also in the Calypso Installation Guide.

The deliverable contains the following directories and files in <calypso home>:

```
client\bin\uploaderMQSeries.bat
client\bin\dbscripts\GatewaySchemaBase.xml
client\bin\dbscripts\GatewaySchemaData.xml
docs\calypso-datauploader\Calypso DataUploader Coverage.xls
docs\calypso-datauploader\format\ - (Spreadsheets containing annotated DUP messages)
docs\calypso-datauploader\samples\ - (CSV and xml messages samples)
docs\calypso-datauploader\schema\ - (XSD files)
client\lib\calypso-cal-upload-x.x.x.jar
client\lib\calypso-datauploader-x.x.x.x.x.x.jar
client\lib\edtftpj-pro-3.1.2.jar
client\lib\jdom-1.0.jar
client\lib\license-files-x.x.x.zip - (Copies of third-party licenses)
```





```
client\resources\calypso_uploader_config.properties.sample
client\resources\datauploader.properties.sample
client\resources\GATEWAYMSG.wf
client\resources\GATEWAYMSG_BACKLOADING.wf
client\resources\gatewayservice.properties.sample
client\resources\PLATFORMMSG.wf
client\resources\sftpuploader.properties.sample
client\resources\UploaderJMSAdmin.config.sample
client\resources\UploADSOURCEMSG.wf
```

## 2.1.1 Property Files

Copy the property and configuration files to <calypso home>/tools/calypso-templates/resources.

You will then need to deploy the files to your applications servers.

Please refer to the Calypso Installation Guide for details.

You can modify "gatewayservice.properties" as needed (Optional).

The application converts imported data into BOMessages, which require a Legal Entity for the Sender and Receiver.

By default, "gatewayservice.properties" sets the Sender to CLIENT and the RECEIVER to CALYPSO. If these entities are not present as Legal Entities, Calypso will create them.

To use a different Sender (i.e., Client) or Receiver (Host), modify the following lines in the file "gatewayservice.properties" as desired:

### 2.1.2 Database Upgrade

When you run Execute SQL as part of your installation, the Data Uploader files will be already loaded. You just need to check the "Gateway" checkbox.

## 2.1.3 SWAPSWIRE\_MAPPING Table Migration

[NOTE: Customers who are upgrading from older versions of SwapsWire/MarkitWire (prior to Calypso version 11.0) that made use of the SWAPSWIRE\_MAPPING table must perform the steps in this section]

Prior to using the Calypso Mapping window for MarkitWire, you must migrate any existing Swapwire mappings to the Calypso Mapping table.

#### (I) [NOTE: Please follow the sequence below. Data loss can occur if the steps are not precisely followed]

**Step 1** – Contact Calypso Product Support to obtain the following migration scripts:

- Oracle: sw\_mapping\_migration\_oracle.sql
- Sybase: sw\_mapping\_migration\_sybase.sql

The migration script copies all existing mapping data from the swapswire\_mapping table to the calypso\_mapping table.



Step 2 - Run Execute SQL with "SwapswireSchemaData.xml".

### 2.1.4 FRA Rounding

You can set the default rounding method for uploaded FRA data by setting the value to UP, DOWN, or NEAREST in the domain "UploadDefaultRoundingMethod". By default, uploaded data is rounded to NEAREST.

## 2.1.5 Sec Finance Lifecycle Events

The trade keyword EventTypeActionExternalRef is propagated to the transfer attribute defined in domain "SecFinance.EventTypeActionExternalRefName", or to the transfer attribute EventTypeActionExternalRef if the domain is not set.

## 2.1.6 Access Permissions

The Data Uploader enforces access permissions on imported data.

Please refer to Calypso Access Permissions documentation "Server Side Access Permissions" for complete details.

Calypso has added support for SavePriceFromTradePriceReport Server Side Access Permissions, thereby allowing them to be enforced when using the Data Uploader:

- SavePriceFromTradePriceReport
- DeletePiceFromTradePriceReport

### 2.1.7 Data Authorization

#### This only applies to Core Calypso version 16.1 and above.

The Data Uploader enforces data authorization (if data authorization is enabled on the system) for the objects defined in the domain "classServerAuthMode", except for Is Admin users.

The data upload will be successful but there will be a warning message as well as pop-up message stating that there are Pending Modifications. The user can then access the "Data Authorization" window to either accept or reject the Pending modifications. Once accepted, the respective objects will be saved into the database.

For example:

Here is after uploading Quote Object and if the authorization is enabled then, user get the following pop-up window:



| Source:       | Uploader              | ✓ Format | : Up       | loaderX $\vee$ | Select File / Fo | Ider C:\work\w | vorkspace\v17 | -sonar-day\cal     | ypso\module       | iles\datauploa 🔄 Refresh 🛛 Uplo   |   |
|---------------|-----------------------|----------|------------|----------------|------------------|----------------|---------------|--------------------|-------------------|---|---|
| Persist Messa | ge: None              | $\sim$   |            |                |                  |                |               |                    |                   |   |   |
| Quote_N       | Quote_NewSample.xml × |          |            |                |                  |                |               |                    |                   |   |   |
| ۹- Filter Dat | a                     |          |            |                |                  |                |               |                    |                   |   |   |
| UploadSta     | CalypsoQ              | CalypsoQ | CalypsoQ   | CalypsoQ       | . CalypsoQ       | CalypsoQ       | CalypsoQ      | CalypsoQ           | Calyp             | <sup>1</sup> , Upload Field   | Value   |
| SUCCESS       | NEW                   | default  | 2004-01-01 | FX.USD.BWP     | Price            | 0.0            |               |                    | 3.0               | CalypsoQuote, Action<br>CalypsoQuote, QuoteSet<br>CalypsoQuote, QuoteSet<br>CalypsoQuote, QuoteName<br>CalypsoQuote, QuoteName<br>CalypsoQuote, Bid<br>CalypsoQuote, Ask<br>CalypsoQuote, Ask<br>CalypsoQuote, Close<br>CalypsoQuote, Close<br>CalypsoQuote, Low<br>CalypsoQuote, Low<br>CalypsoQuote, Last<br>CalypsoQuote, EstimatedB | NEW<br>default<br>2004-01-01<br>FLUSD_BWP<br>Price<br>0.0<br>3.0<br>3.0 |
|               |                       |          |            |                |                  |                |               | Warning<br>Data Au | ithorization Pend | X Iding for the Item Quotes   | Value   |

#### Supported Objects

Account

AccountingRuleAccountConfig

AccountingRule

AccountingBookRuleLink

AccountingEventConfig

AccountInterestConfig

AccountSweepingConfig

Basket

Bond

Book

Commodity

CommodityReset

Country

CurrencyDefault

CurrencyPair

Fund

FutureContract

FutureOptionContract

FXReset

HolidayRule



LegalEntity ManualSDI PortfolioSwapContract StaticDataFilter

SettleDeliveryInstruction

QuoteValue

Equity

RateIndexDefaults

LEContact

CreditRating

HolidayCode

HedgeRelationshipDefinition

MatchingAlias

# 2.2 Calypso Mapping Window

You can access the Calypso Mapping Window using Processing > Tools > Calypso Mapping (menu action mapping.CalypsoMappingWindow) from the Calypso Navigator.

The Calypso Mapping window allows mapping data for various interfaces by providing the interface values and the corresponding Calypso values.



Calypso has added new REST end point to export calypso mappings in CUD format based on query criteria interfaceName, Type, and interfaceValue. Following 3 query parameters can be used to filter out the results:



interfaceName - to fetch the mappings related to specific interface

type - to fetch the mappings only for the specific type

interfaceValue - to fetch mappings only for the specific interface value

Users can search for various mapping results using the "Search" field.

## 2.2.1 Reverse Default Functionality

Calypso uses the table for reverse mapping values from Calypso to external interfaces (e.g., MarkitWire, etc.). In particular scenarios, multiple Interface (MarkitWire) values can have the same Calypso value. To map the default Interface value for a particular Calypso value, use the "Reverse Default" checkbox in the Calypso Mapping window.

For example, for DayCount, the MarkitWire values, ACT/365.FIXED and ACT/365.ISDA both have the Calypso value, ACT/365. In the Markitwire Backloading process, the Calypso value is mapped to a single Markitwire value. To permit the user to choose one of these MarkitWire values as the default mapping, check the "Reverse Default" checkbox when mapping that value. With "Reverse Default" checked on the ACT/365.FIXED mapping, the BackLoading report for MarkitWire shows ACT/365.FIXED if the Calypso value is ACT/365.

## 2.2.2 Access Permissions

The following access permissions apply to the Calypso Mapping window:

- ViewCalypsoMapping Permission to view mappings.
- ModifyCalypsoMapping Permission to modify mappings.

| Select Functions      |      |                       | ×        |
|-----------------------|------|-----------------------|----------|
| Filter calypsomapping |      | Filter calypsomapping |          |
| Permissions           |      |                       |          |
| ModifyCalypsoMapping  | >>   | AllowERSMakeOfficial  |          |
| ViewCalypsoMapping    |      | AllowExecuteSQLDelete |          |
|                       | <<   | AllowExecuteSQLInsert |          |
|                       |      | AllowExecuteSQLSelect |          |
|                       | Up   | ConfigPairingMatching |          |
|                       | _    | Connect               |          |
|                       | Down | CreateAuthConfig      |          |
|                       |      | CreateCurve           | <b>–</b> |
|                       |      | <b>▲</b>              | ·        |

## 2.2.1 Calypso Mapping Audit

(Optional) – You can enable Audit Reporting for Calypso Mapping by adding "CalypsoMapping" to the domain "classAuditMode".



| Ľ | Domain Values Window (User: calypso_user) |      |         |                                       |  |  |  |  |  |
|---|---|------|---------|---------------------------------------|--|--|--|--|--|
|   | Search: classaud                          | Find | 🗖 Value |                                       |  |  |  |  |  |
|   | 🕀 🖽 CFDRiskArbitrage.Pricer               |      |         | Name: classAuditMode                  |  |  |  |  |  |
|   |   |      |         | · · · · · · · · · · · · · · · · · · · |  |  |  |  |  |
|   | 🗄 📲 CheckUnauthorizedSDI                  |      |         | Value: CalypsoMapping                 |  |  |  |  |  |
|   | ⊞ <u>III</u> city                         |      |         |                                       |  |  |  |  |  |
|   | 🗄 📖 classAuditMode                        |      |         | Comment:                              |  |  |  |  |  |
|   |   |      |         | ,                                     |  |  |  |  |  |
|   |   |      |         |                                       |  |  |  |  |  |
|   |   |      |         | << A00                                |  |  |  |  |  |

▶ Refer to Calypso Data Audit documentation for information on using the Audit Report.

## 2.2.2 Mapping of Interfaces with Calypso Upload Document

The Data Uploader provides two types of mapping:

• Fix tag number to CUD xpath

Interface value should be the Fix tag number and Calypso value should be CalypsoTrade xpath.

There is a special case for NotesGroup since this is a repeating group. For NotesGroup mapping, the interface value should be "NotesGroup.(labelname)" and Calypso value should be the CalypsoTrade xpath.

The mapping type name is "TradeMappings".

• Fix tag number to Keyword name in CUD.

Interface value should be Fix tag and Calypso value should be the keyword name.

The mapping type name is "TradeKeywordMappings".

To enable this functionality, you need to add the mappings translator.

| Zalypso Mapping Window |                  |                             |
|------------------------|------------------|-----------------------------|
| Interface Mappings     |                  |                             |
| BloombergFIT           | Name:            | Uploader/MappingsTranslator |
| E CME<br>E FpML        | Interface Value: | BloombergFIT                |
|                        | Calypso Value:   | Fix                         |
|                        | Reverse Default: |                             |
| Uploader               | << Add           |                             |
| BloombergFIT           | >> Remove        |                             |

In this example, the functionality is enabled for "BloombergFIT" (source).

The calypso value Fix represents the default translator to be used in the absence of source and format specific translator.



The mappings translator class is "<source> + <format> + MappingsTranslator". If not found, then the class "<source> + MappingsTranslator" is used. And if not found "<format> + MappingsTranslator" is used.

Currently the supported formats are Fix and Xml. Interfaces with XML data type need to put Calypso Value =Xml. Example of TradeMappings:





Example of NotesGroup Mapping:

Tag 9613 of NotesGroup is populated in the TradeBundleName of CUD as per the mapping provided in the mapping table.









Example of TradeKeywordMappings:

The TestKeywordName trade keyword is created in CUD with value same as tag 37 of the Fix message as per the mappings provided in mapping table for TradeKeywordMappings.





## 2.3 Workflow Configuration

## 2.3.1 GATEWAYMSG Message Workflow

"GATEWAYMSG.wf" provides a standard message workflow for GATEWAYMSG messages.

- () NOTE: Prior to importing the workflows, it is recommended that you create a backup of your existing workflows using File > Export all in the Workflow Configuration window.
- **I** NOTE: Please use the settings below or you will OVERWRITE your existing workflows.

**Step 1** – From the Calypso Navigator, navigate to **Configuration > Workflow > Workflow**.

Step 2 - In the Workflow Configuration window, choose File > Import current.

#### Step 3 - Select the following settings:

- Processing Org = ALL
- Event Class = **PSEventMessage**
- Subtype = GATEWAYMSG
- Product = ALL

| Select Workflow To Process |                                    | × |
|----------------------------|------------------------------------|---|
| Processing Org             | ALL                                |   |
| Event Class PSEventMessage | ▼ Subtype GATEWAYMSG ▼ Product ALL | • |
| Apply                      | Cancel                             |   |

**Step 4** - Click **Apply**. The confirmation window appears:





**Step 5** - Click **Yes** to import or **No** to abandon the operation.

If you click Yes, use the file browser to select the file "GATEWAYMSG.wf".



If you are updating an existing Data Uploader instance, the application displays a confirmation message:

| Questi | on                            | ×         |
|--------|-------------------------------|-----------|
| 2      | Do you want to remove current | workflow? |
|        | Yes <u>N</u> o                |           |

Click **Yes** to continue the import operation. The application imports the workflows and then displays a message showing what actions were accomplished:



| Message |  |
|---------|--|
| 1       | 15 tasks imported,<br>0 KickOff Configs imported,<br>0 Custom Rule Definitions imported,<br>0 SD filters imported (3 not saved, probably because definition in calypso and in the .wf file are the same),<br>0 Audit filters imported,<br>0 Mapping Status imported. |
|         | ΟΚ   |

The Gateway Message workflow is now available:

| Id    | Orig Status   | Action    | Resulting Status | Different User | Use STP  | Priority | Log | Subtype    | Product Type | Rules              |
|-------|---------------|-----------|------------------|----------------|----------|----------|-----|------------|--------------|--------------------|
| 77712 | NONE          | NEW       | PENDING          |                | <b>V</b> | 0        |     | GATEWAYMSG | ALL          |                    |
| 77713 | NONE          | NEW       | PENDING_VALID    |                | <b>V</b> | 0        |     | GATEWAYMSG | ALL          |                    |
| 77714 | PENDING       | CANCEL    | CANCELED         |                |          | 0        |     | GATEWAYMSG | ALL          |                    |
| 77715 | PENDING       | COMPLETE  | COMPLETED        |                | 1        | 0        |     | GATEWAYMSG | ALL          |                    |
| 77716 | PENDING       | REPROCESS | REPROCESSING     |                |          | 0        |     | GATEWAYMSG | ALL          | UploadReprocess    |
| 77717 | PENDING       | UPDATE    | PENDING          |                |          | 0        |     | GATEWAYMSG | ALL          |                    |
| 77718 | PENDING_TRADE | CANCEL    | CANCELED         |                |          | 0        |     | GATEWAYMSG | ALL          | CancelCleanUp      |
| 77719 | PENDING_TRADE | LOAD      | COMPLETED        |                | 1        | 0        |     | GATEWAYMSG | ALL          | CheckLink,Loader   |
| 77720 | PENDING_TRADE | REPROCESS | PENDING_TRADE    |                |          | 0        |     | GATEWAYMSG | ALL          | ReMap              |
| 77721 | PENDING_VALID | CANCEL    | CANCELED         |                |          | 0        |     | GATEWAYMSG | ALL          | CancelCleanUp      |
| 77722 | PENDING_VALID | REPROCESS | PENDING_VALID    |                |          | 0        |     | GATEWAYMSG | ALL          | ReMap              |
| 77723 | PENDING_VALID | VALIDATE  | PENDING_TRADE    |                | 1        | 0        |     | GATEWAYMSG | ALL          | CheckLink,Validate |
| 77724 | REPROCESSING  | CANCEL    | CANCELED         |                |          | 0        |     | GATEWAYMSG | ALL          |                    |
| 77725 | REPROCESSING  | COMPLETE  | COMPLETED        |                |          | 0        |     | GATEWAYMSG | ALL          |                    |
| 77726 | REPROCESSING  | REJECT    | PENDING          |                |          | 0        |     | GATEWAYMSG | ALL          |                    |



**Step 6** - You need to add the FORCE\_NEW action manually to the GATEWAYMSG workflow:

PENDING\_VALID - FORCE\_NEW - PENDING\_VALID with the workflow rule "ForceNew"

The FORCE\_NEW action is a manual action that can convert the AMEND subaction to the NEW subaction.



|                  | Processi         | ng Org ALL    |                     |                                       |
|------------------|------------------|---------------|---------------------|---------------------------------------|
| Event Class      | PSEventMessage   | ▼ Subtype     | GATEWAYMSG 🔹        | Product ALL                           |
| Orig Status      | PENDING_VALID    | ▼ Action      | FORCE_NEW           | Result Status PENDING_VALID           |
| 🔲 Different User | 📝 Create Task    | Use STP       | Use KickOff/Cut Off | 0 Priority                            |
| Log Completed    | Preferred Action | 📃 Update Only | Generate Intermedia | ry Event 🔄 Needs manual Authorization |
| Rules            | ForceNew         |               |                     | Help                                  |
| Filter           |                  |               |                     | Custom Rules Definition               |
| Comment          |                  |               |                     |                                       |

**Step 7** – (Optional) You can add the FORCE\_AMEND action as needed to the GATEWAYMSG workflow:

PENDING\_VALID - FORCE\_AMEND - PENDING\_VALID with the workflow rule "ForceAmend"

The FORCE\_AMEND actions forces an amend on a trade for instances when the interface is down and a trade is entered using a manual link to an existing MarkitWire trade in Calypso.

| Image: State Sta | 56               | COMPLETED            |                          | TRADE CANCEL  |      |
|--|------------------|----------------------|--------------------------|---------------|------|
| Id 4301  | _                | Action FORCE_        | AMEND                    | VANDATE CA    | NCEL |
| Orig Status PENDING_VALID  |                  | Result Status PENDIN | G_VALID                  |               |      |
| Event Class PSEventMessage   |                  | Subtype GATEW/       | AYMSG                    | FORCE_AMEND   |      |
| Product ALL  |                  | Processing Org ALL   |                          | PENDING_VALID |      |
| Different User   | Create Task      | Use STP              | Use KickOff/Cut Off      | REPROCESS     |      |
| Log Completed  | Preferred Action | Update Only          | Generate Intermediary Ev |               |      |
|  |                  |                      | Needs man. Auth.         | NEW           |      |
| Rules ForceAmend   |                  |                      | Help                     |               |      |
| Filter   |                  | Custom Rules Def     | inition                  |               |      |
| Comment  |                  |                      |                          |               |      |
|  |                  | Save Delete          | Close                    | NONE          |      |

If a trade with given External Reference already exists in Calypso, then a new MarkitWire trade will not be uploaded to Calypso. Such a trade will fail with exception task "Trade Already exists with External Reference: MW\_CALYPXXXX\_1234567" and the GATEWAYMSG remains in PENDING\_VALID state.

To replace the existing trade with the new MarkitWire trade, manually apply the FORCE\_AMEND action on the GATEWAYMSG. The action results in the Calypso trade be forced from the NEW subaction to the AMEND subaction, and the MarkitWire trade will then be uploaded into Calypso with the AMEND action.



## 2.3.2 KickOff/CutOff Config (for Local Mode Only)

Users should configure KickOff/CutOff for GATEWAYMSG workflow for use in Local mode. For setting up kickoff/cutoff, please refer to below steps and sample snapshot for configurations.

| Section WorkFlow Action    |                       |                     |                            |  |  |  |
|----------------------------|-----------------------|---------------------|----------------------------|--|--|--|
| Id 37700                   |                       | Action REPRO        | DCESS                      |  |  |  |
| Orig Status PENDING        |                       | Result Status REPRO | DCESSING                   |  |  |  |
| Event Class PSEventMessage |                       | Subtype GATEWAYMSG  |                            |  |  |  |
| Product ALL                |                       | Processing Org ALL  |                            |  |  |  |
| Different User             | Create Task           | Use STP             | Use KickOff/Cut Off        |  |  |  |
| Log Completed              | Preferred Action      | Update Only         | 🔲 Generate Intermediary Ev |  |  |  |
| 🔽 Needs man, Auth.         |                       |                     | 2 Priority                 |  |  |  |
| Rules UploadReprocess      | ,CheckKickOff,CheckCu | tOff                | Help                       |  |  |  |
| Filter                     |                       | Custom Rules D      | efinition                  |  |  |  |
| Comment                    |                       |                     |                            |  |  |  |
|                            |                       | Save Delete         | Close                      |  |  |  |

From **PENDING** status have a **STP** transition for kickoff/cutoff workflow. The 'Use Kick Off / Cut Off' must be enabled on this transition and it **must** have following message rules:

- CheckKickOff
- CheckCutOff (if cut-off enabled)
- UploadReprocess This is specific to Uploader in Local mode and should only be used if you are using Kick-Off

It is advisable to keep this (kickoff/cutoff) transition of low priority (shown as 2 above), so that other STP transition passing from this status are given first preference.

#### 1. Kick Off /Cut Off Config:

For any kickoff/cutoff, user needs to specify its kick-off and cut-off time. For set-up and to know more about these configurations, please refer to the Calypso workflow setup guide.

For Uploader kick-off/cut-off, the configuration **must** have

- a. Method as 'ALL'
- b. Date Calculator as 'Uploader'



It is recommended to have a kick-off time of more than 2 minutes.

#### Sample snapshot for Kickoff/Cutoff config:

| Edit Browse |               |                             |         |                    |          |               |          |          |      |
|-------------|---------------|-----------------------------|---------|--------------------|----------|---------------|----------|----------|------|
| Wor         | kflow ALL/PS  | EventMessage/GATEWAYMSG,    | /ALL/PE | NDING/REPROCESS/R  | EPROCES  | SING          |          |          |      |
| Rec         | eiver         |                             |         | Method             | ALL      |               | -        | ·        |      |
| Curr        | rency ANY     | <b>~</b>                    |         | Date Calculator    | Uploader |               |          | ·        |      |
| SD          | Filter        |                             |         |                    |          |               | HH       | MM       |      |
| Dat         | te Roll NO_CH | IANGE                       |         | KickOff Days Lag   | 0        | KickOff Time  | e 0 :    | 2        |      |
| Holi        | idays         |                             |         | No CutOff          |          |               |          |          |      |
| Time        | Zone Americ   | a/New_York                  | _       | CutOff Days Lag    | 0        | CutOff Time   | e 0 :    | 5        |      |
|             | Che           | ck Holiday                  |         |                    |          |               |          |          |      |
|             |               | olute Time                  |         |                    | S        | can Frequency | 0 :      | 0        |      |
|             | <u> </u>      | Jude Time                   |         |                    |          |               |          |          |      |
|             |               | New                         |         | Delete Sa          | ve       | SaveAsNe      | w        |          |      |
| Kick0 =     | Workflow Id   |                             | Workf   | low Description    |          |               | Receiver | Currency | Meth |
| 38196       | 37700         | ALL /PSEventMessage/GATEW   | AYMSG   | ALL PENDING REPRO  | CESS/REP | ROCESSING     | ALL      | ANY      | ALL  |
| 27477       | 27412         | ALL/TradeBundle/ALL/ALL/PEN | DING/E  | XECUTE/VERIFIED    |          |               | ALL      | ANY      | ALL  |
| 12984       | 12981         | ALL/PSEventMessage/RECEIP   | TMSG/A  | LL/VERIFIED/TO SEN | ID/TO_BE | SENT          | ALL      | USD      | SWIF |
| 12983       | 12981         | ALL/PSEventMessage/RECEIP   | TMSG/A  | LL/VERIFIED/TO_SEN | ID/TO_BE | SENT          | ALL      | CAD      | SWIF |
| 12982       | 12981         | ALL/PSEventMessage/RECEIP   | TMSG/A  | LL/VERIFIED/TO_SEN | ID/TO_BE | SENT          | ALL      | ANY      | SWIF |
| 12940       | 12910         | ALL/PSEventMessage/PAYME    | TMSG/   | ALL/VERIFIED/TO_SE | ND/TO_BE | SENT          | ALL      | USD      | SWIF |
| 12939       | 12910         | ALL/PSEventMessage/PAYME    | TMSG/   | ALL/VERIFIED/TO_SE | ND/TO_BE | SENT          | ALL      | CAD      | SWIF |
| 12938       | 12910         | ALL/PSEventMessage/PAYME    | TMSG/   | ALL/VERIFIED/TO_SE | ND/TO_BE | _SENT         | ALL      | ANY      | SWIF |
| 4           |               |                             |         |                    |          |               |          |          | •    |

Note: In the case of local mode, new tasks are created for each KickOff, marking the previous tasks as complete. As new tasks are created, Scan Frequency parameter is not used. Similar behavior is achieved by setting KickOff and CutOff.

For example: - to reprocess every 2 minutes and to stop the reprocessing after 10 minutes, please configure the following-

#### KickOff=2, CutOff=10

Scan Frequency can be left 0 or any other value. It is not used.

Effectively, KickOff is the frequency for reprocessing, CutOff is the total time after which the kick offs should be stopped.



### 2.3.3 Trade Workflow

#### Workflow Rules

Termination, Novation, and Cancellation Trade workflow rules must be set in the Trade workflow for the products for which you use the Data Uploader, or for ALL.

Step 1 – On the VERIFIED – TERMINATE - TERMINATED transition, add the UpdateTermination workflow rule.

|                  | Processin                  | ng Org        |                        |                                      |
|------------------|----------------------------|---------------|------------------------|--------------------------------------|
| Event Class      | PSEventTrade               | ▼ Subtype     | ALL 👻                  | Product ALL 👻                        |
| Orig Status      | VERIFIED                   | ▼ Action      | TERMINATE              | Result Status TERMINATED 🔹           |
| 🔲 Different User | 📄 Create Task              | Use STP       | 🔲 Use KickOff/Cut Off  | 0 Priority                           |
| Log Completed    | Preferred Action           | 🔲 Update Only | 🔲 Generate Intermediar | y Event 🔄 Needs manual Authorization |
| Rules            | UpdateTermination          |               |                        | Help                                 |
| Filter           |                            |               |                        | Custom Rules Definition              |
| Comment          | Trade has been terminated. |               |                        |                                      |

Step 2 - On the VERIFIED – CANCEL - CANCELED transition, add the Cancel workflow rule.

|                  | Processin        |               |                       |                |                        |       |
|------------------|------------------|---------------|-----------------------|----------------|------------------------|-------|
| Event Class      | PSEventTrade     | ▼ Subtype     | ALL                   | Product        | ALL                    | •     |
| Orig Status      | VERIFIED         | ▼ Action      | CANCEL 🗸              | Result Status  | CANCELED               | •     |
| 📄 Different User | 🔲 Create Task    | Use STP       | 🔲 Use KickOff/Cut Off | 0 P            | riority                |       |
| Log Completed    | Preferred Action | 📃 Update Only | 🔲 Generate Intermedi  | ary Event 📄 Ne | eds manual Authoriza   | ition |
| Rules            | Cancel           |               |                       |                |                        | Help  |
| Filter           |                  |               |                       | <u> </u>       | istom Rules Definition |       |

Comment This is system action. No user should be allowed access to this action.

**Step 3** – On the TERMINATED – UNDO\_TERMINATE – VERIFIED transition, click **Custom Rules Definition** to create the custom rule CustomUploaderUndoTerminationRule:



| Custom Rule Editor Window       |                       |                                       |                   |
|---------------------------------|-----------------------|---------------------------------------|-------------------|
| Other Rules                     | Available Fields      | Name: :omUploaderUnd                  | loTerminationRule |
| AUndoTermination                | BookName              |                                       |                   |
| AUpdateUndoTermination          | CounterParty          | Type: PSEventTrade                    | -                 |
| ApplyCLS                        | EnteredDate           | , , , , , , , , , , , , , , , , , , , |                   |
| AutomaticFees                   | EnteredTime           | Insert Test                           |                   |
| BrokerFees<br>CATransformation  | Id<br>LegalEntityRole | if                                    |                   |
| Cancel                          | Quantity              |                                       |                   |
| CancelAllocatedTrade            | SettleCurrency        | then                                  |                   |
| CancelRemainderOfPartialE       | SettleDate            |                                       |                   |
| CheckAllocatedTrade             | TradeCurrency         | else                                  |                   |
| CheckAmend                      |                       |                                       |                   |
| CheckAmended                    |                       | Insert Te                             | st                |
| CheckBasicProductExercise 🖵     |                       |                                       |                   |
|                                 |                       |                                       |                   |
| Insert Rule                     | Insert Field          | New                                   | Help              |
|                                 |                       | Load                                  | Save              |
| Rule Expression                 |                       |                                       | Cause As a        |
| UpdateUndoTermination() && Reje | ct()                  |                                       | Save As           |
|                                 |                       |                                       | Remove            |

Set the custom rule as a combination of the rules UpdateUndoTermination and Reject.

Then add the rule CustomUploaderUndoTerminationRule on the TERMINATED – UNDO\_TERMINATE – VERIFIED transition.

|                  | Processi   | ng Org ALL    |                   |         |               |                   |            |
|------------------|--|---------------|-------------------|---------|---------------|-------------------|------------|
| Event Class      | PSEventTrade   | ▼ Subtype     | ALL               | •       | Product       | ALL               | •          |
| Orig Status      | TERMINATED   | ▼ Action      | UNDO_TERMINATE    | •       | Result Status | VERIFIED          | •          |
| 📄 Different User | 🔲 Create Task  | Use STP       | 🔲 Use KickOff/Cut | : Off   | 0 P           | riority           |            |
| Log Completed    | Preferred Action   | 🔲 Update Only | 🔲 Generate Interr | mediary | y Event 📄 Ne  | eds manual Auth   | norization |
| Rules            | CustomUploaderUndoTermin                                     | nationRule    |                   |         |               |                   | Help       |
| Filter           |  |               |                   |         | Cu            | istom Rules Defii | nition     |
| Comment          | Trade was termianted by mistake. This action will be undone. |               |                   |         |               |                   |            |



NOTE: Any changes to the BO Message workflow rules must be reflected in the file "gatewayservice.properties" to indicate which status codes are used for blocking messages. These are used to link together the BO Messages.

#### Domain Values

A number of domain values need to be added.

• Add GATEWAY to the domain "exceptionType"

| Domain Values Window       |         |                                    |                     |  |
|----------------------------|---------|------------------------------------|---------------------|--|
| Search: exceptiontype Find | 🗖 Value |                                    |                     |  |
| EARLY_PAYMENT              |         | Name: exceptionType                |                     |  |
| EOD_VALUATION              |         | Value: GATEWAY                     |                     |  |
| ERRORS_REPORTED            |         |                                    |                     |  |
| EXCEPTION                  |         | Comment: To capture the exceptions | in the task station |  |
|                            |         | ,                                  |                     |  |
|                            |         | << Add                             | Save Above          |  |

- (Optional) Add the termination action to the domain "UploadTerminateAction". The default action is TERMINATE if not set.
- (Optional) Add the novation action to the domain "UploadNovateAction". The default action is TERMINATE if not set.
- Add the action to apply when you must exit from an outside trade (e.g., a MarkitWire trade) to the domain "UploadExitAction". This action will affect the upstream trade outside Calypso.
- (Optional) Set the default termination reason in the domain "UploadTerminationReason". The default is Assigned if not set.
- (Optional) Set the default novation reason in the domain "UploadNovationReason". The default is Assigned if not set.
- () NOTE: If you modify the default actions, you need to define those actions in the Trade workflow as well, with the corresponding workflow rules

## 2.4 Task Station Setup

From the Calypso Navigator, navigate to Processing > Task Station to bring up the Task Station.

Then add reports for GATEWAY exception events and GATEWAY message events.



#### GATEWAY Exception Events

#### Report Configurations $\times$ Task Station Tabs Task Enrichment Filters Q.-Name Value Gateway Exceptions Tab Name Gateway Exceptions Workflow Types Exception Books \_\_ANY\_\_ **Book Attributes** Event Types EX\_GATEWAY, EX\_GATEWAYMSG LOW, NORMAL, HIGH, CRITICA ... Priorities Task Statuses NEW, UNDER\_PROCESSING, C ... Enrichment Columns Filter TaskDatetime Task Date Type From Tenor -1M To Tenor +1M Catalog Auto Count

| Q~ gateway<br>EX_GATEWAY<br>EX_GATEWAYMSG<br>EX_GATEWAYMSG_ERROR<br>EX_GATEWAYMSG_IGNORE<br>EX_GATEWAYMSG_INFORMATION |
|---|
| EX_GATEWAY<br>EX_GATEWAYMSG<br>EX_GATEWAYMSG_ERROR<br>EX_GATEWAYMSG_IGNORE  |
| EX_GATEWAYMSG<br>EX_GATEWAYMSG_ERROR<br>EX_GATEWAYMSG_IGNORE  |
| EX_GATEWAYMSG_ERROR<br>EX_GATEWAYMSG_IGNORE   |
| EX_GATEWAYMSG_IGNORE  |
| EV CATEWAYMEE INFORMATION   |
| EX_GATEWAYMSG_INFORMATION   |
| EX_GATEWAYMSG_REJECT  |
| EX_GATEWAYMSG_WARNING   |

#### GATEWAY Message Events

#### A Report Configurations

 $\times$ 

| Task Station Tabs Task Enrichment Filters |                           |                            |  |  |
|---|---------------------------|----------------------------|--|--|
| Q-  | Name                      | Value                      |  |  |
| Gateway Messages                          | Tab Name                  | Gateway Messages           |  |  |
|   | Workflow Types            | Message                    |  |  |
|   | Books                     | ANY                        |  |  |
|   | Book Attributes           |                            |  |  |
|   | Event Types               | ACCEPTED_GATEWAYMSG, ACCEF |  |  |
|   | Priorities                | LOW, NORMAL, HIGH, CRITICA |  |  |
|   | Task Statuses             | NEW, UNDER_PROCESSING, C   |  |  |
|   | Enrichment Columns Filter |                            |  |  |
|   | Task Date Type            | Message Creation Date      |  |  |
|   | From Tenor                | -1M                        |  |  |
|   | To Tenor                  | +1M                        |  |  |
|   | Catalog Auto Count        |                            |  |  |



/

Available Q- gateway ACCEPTED\_GATEWAYMSG ACCEPTED\_MWGATEWAYMSG ACCEPTED\_SAPGATEWAYMSG ACKED UNMATCHED\_GATEWAYMSG ACKED UNMATCHED\_MWGATEWAYMSG ACKED UNMATCHED SAPGATEWAYMSG

# 2.5 Pricing Environment

The Data Uploader engine and the Uploader Import Message engine use the pricing environment from the Engine Config by default. If the pricing environment is not provided in the Engine Config, the pricing environment is picked up from the User Defaults. If the pricing environment is not specified in the User Defaults, the Pricing Environment named "default" is used (if any).

# 2.6 Uploader Import Message Engine Configuration

Incoming messages are automatically imported using the Uploader Import Message engine (an extension of the Import Message engine). It is triggered by an IEAdapterListener listening to an IEAdapter. Each time a new message is detected by the adapter, it calls the engine which saves the incoming message into Calypso as a GATEWAYMSG following the workflow previously defined.

## 2.6.1 Engine Configuration

The Uploader Import Message engine is configured in the Engine Manager of Web Admin: event subscription and engine parameters.

You may need to add this engine if it is not available for configuration: Create a new engine called UploaderImportMessageEngine with class name com.calypso.tk.engine.UploadImportMessageEngine.

Make the UploaderImportMessageEngine a consumer of the events:

PSEventUpload

Select the event class "PSEventUpload" from the 'Persisted Event Configuration' field and click 😏 to add it to the box area.

The "config" engine parameter (configuration attribute) is config = Uploader

UpdateManagerEngine is used for reprocessing messages. In case of Bulk-upload, it will process uploadRequestMessages and it will create uploadResponseMessages.





## 2.6.2 Domain Values

From the Calypso Navigator, navigate to Configuration > System > Domain Values to add "tk.util.UploadPoolSequencePolicy" to the domain "engineEventPoolPolicies".

| Z Domain Values Window (User: calypso_user) |   | ×   |  |
|---|---|-----|--|
| Search: Find Value                          |   |     |  |
| EmissionComplianceType                      | Name: engineEventPoolPolicies                     |     |  |
| EnableExoticBaskets                         |   | - 1 |  |
| engineEventPoolPolicies                     | Value: tk.util.UploadPoolSequencePolicy           |     |  |
| 🚽 🚽 tk.util.FIFOPoolSequencePolicy          |   | = [ |  |
| 🚽 🚽 tk.util.PositionEngineSequencePolic     | Comment: New Policy created for UploaderImportMes | sag |  |
| 🚽 🚽 tk.util.TradeLiquidationEventSequer     |   |     |  |
| 🖖 tk.util.TradeLiquidationSequencePo        |   |     |  |
|   | << Add Save Above                                 |     |  |
| 🚽 😔 tk.util.TransferInventorySequence       | >> Remove   |     |  |
| tk.util.UploadPoolSequencePolicy            | 22 Kenove   |     |  |
| engineEventPoolPolicyAliases                | Constraints                                       |     |  |

You also need to add "UploadPoolSequencePolicy" to the domain "engineEventPoolPolicyAliases", with the Comment = "tk.util.UploadPoolSequencePolicy".

| 🟒 Domain Values Window (User: calypso_u | user)  |          |                                  |
|---|--------|----------|----------------------------------|
| Search: Find Va                         | alue   |          |                                  |
| EmissionComplianceType                  | ^      | Name:    | engineEventPoolPolicyAliases     |
|   |        | Value:   | UploaderImportMessageEngine      |
| engineEventPoolPolicyAliases            |        | Comment: | tk.util.UploadPoolSequencePolicy |
| Inventory                               | _      |          |                                  |
|   |        | << A(    | dd Save Above                    |
| PositionEngine                          | Engine |          | nove                             |
| engineName                              |        | Constra  | aints                            |

### 2.6.3 Engine Parameters

Set the following engine parameters using the Engine Manager in Web Admin:

- EVENT\_POOL\_POLICY = UploadPoolSequencePolicy.
- config = Uploader

For property file "calypso\_uploader\_config.properties".



#### 2.6.4 Environment Properties

When using Upload Import Message Engine to consume messages from queues, the message size is limited by environment property EVENT\_LIMIT\_SIZE. The default value is 1MB so you may need to increase it as needed.

RESTART\_ATTEMPTS controls the number of restart attempts by the Uploader Import Message engine if the connection is lost with MQ (default is 10).

RESTART\_DELAY is the delay between attempts (in seconds, default is 5).

### 2.6.5 **Property File**

The property file "calypso\_uploader\_config.properties" has a number of properties to deal with the Uploader Framework.

```
# Import Message Engine (MQ)
input.queue.name=dynamicQueues/input
dynamicQueues/input.queue.setContext=true
# Sender Engine (JMS)
#For ActiveMO
jms.url=tcp://localhost:61616
jms.modetypeclass=org.apache.activemq.jndi.ActiveMQInitialContextFactory
jms.queue.connectionFactory=ConnectionFactory
#For IBMMO
#jms.url=file://localhost/c:/tools/ibm/mqs/binding
#jms.modetypeclass=com.sun.jndi.fscontext.RefFSContextFactory
#jms.queue.connectionFactory=QueueConnectionFactory
output.queue.name=dynamicQueues/output
dynamicQueues/output.queue.ackType=auto
dynamicQueues/output.queue.persist=true
dynamicQueues/output.queue.transacted=false
#the following property can be used to indicate what mode to be used when using the uploader
framework
uploadMode=Local
```

persistMessages=Failure

The uploadMode property indicates which mode to use when utilizing the Uploader Framework. The possible values are **Local** and **LocalAPI**.

#### INOTE: BOMessage is deprecated and no longer supported.



- Local (recommended mode for best performance) All processing occurs on the client side. The property persistMessages controls whether BO Messages are created or not.
  - persistMessages=All A BO message is created for every external message. Failed Messages are reprocessed by applying REPROCESS action, which generates an event for the Update Manager Engine for reprocessing (see setup details below).
  - persistMessages=None No BO Message is created (same as using uploadMode=LocalAPI). The Import Message Engine sends the Acknowledgment to the "output.queue.name" obtained from its configuration. If "output.queue.name" is not configured, no acknowledgement is sent.
  - persistMessages=Failure A BO Message is created only when there is a validation failure in saving the trade. Failed messages are reprocessed by the Update Manager engine (see setup details below).

In Local mode, the uploader caches the Pricing Environment, and the property 'PricingEnvCacheSize' in 'gatewayservice.properties' is used to determine the size of the cache.

# Determines the size of cache to hold PricingEnv, by default its size is 10.
PricingEnvCacheSize=10

Pricing Environment is required to be loaded by the uploader, when uploading certain trade types like FX and Bonds, and loading the pricing environment for every trade that is uploaded will consume a lot of time and hence uploader framework when run in 'Local' mode caches the pricing environment, so that the trades can be uploaded without consuming lot of time.

This property indicates the number of pricing environments to be loaded in the cache. And when the cache size reaches the value of this property, the least recently used item will be removed from the cache.

• LocalAPI – This is Similar to uploadMode=Local, and persistMessages=None. LocalAPI is deprecated and will not be supported in the coming releases. Use the Local mode to upload. All processing occurs on the client, and caches are built and used to improve performance. No BO Message is created.

You should increase the Number of Threads in the Engine configuration to the number of available cores to improve performance.

#### I NOTE: To switch to uploadMode=Local on an existing implementation, you need to:

- Make sure to either COMPLETE or CANCEL any pending BO messages and consume all the events related to PSEventRepublish.
- Run the schema data with (GatewaySchemaBase.xml and GatewaySchemaData.xml).
- Import the workflows GATEWAYMSG.wf and UPLOADSOURCEMSG.wf as they have been modified.
- Get latest datauploader and gatewayservice properties. Change the properties in configuration files as explained above.
- Configure the Update Manager Engine UploaderPublishEngine is no longer needed when using the uploadMode=Local. All these events will be handled by UpdateManagerEngine.]

#### Enabling SSL for IBM MQ


SSL connectivity with CipherSuite enabled only works with the following versions of IBM MQ, and NOT on lower versions: 7.0.1.13+, 7.1.0.7+, 7.5.0.5+, 8.0.0.2+, 9.1.x.x+.

The following properties need to be added in "calypso\_uploader\_config.properties" to enable SSL for IBM MQ:

#IBM SSL Support
sslVersion=TLSv1.2
sslKeystore=keystore.jks
sslKeystorePassword=<password>
sslTrustStore=truststore.jks
sslTrustStorePassword=<password>
#sslTrustStoreAlgorithm=
#sslKeystoreAlgorithm=
sslCipherSuite=TLS RSA WITH AES 256 CBC SHA256

#### The SSL functionality does NOT work with JNDI. Connection must be done via those properties.

Also, when running the Uploader Import Message Engine, please add the following argument to engineserver.bat: -Dcom.ibm.mq.cfg.useIBMCipherMappings=false

#### Lost Messages when persistMessages=None

When a message is not delivered and an event is not created in Calypso, the message will go to the Dead Letter Queue.

You can check failed messages from that queue.

You need to have the following properties in "calypso\_uploader\_config.properties":

input.queue.name=dynamicQueues/input

dynamicQueues/input.queue.transacted=true

And while sending data from MQ, the "Persistent Delivery" flag should be checked.

#### 2.6.6 Sender Engine Configuration

Add the following items to the Message Sender configuration.

From the Calypso Navigator, navigate to Configuration > Messages & Matching > Message Sender.



| 📈 Message Sende    | r Config (User: calypso_use      | r)       |              |          | _ 🗆 ×    |
|--------------------|----------------------------------|----------|--------------|----------|----------|
| Sender Config Cop  | y Config                         |          |              |          |          |
|                    |                                  |          |              |          |          |
| Message Status     | PENDING_TRADE                    | -        | Product Type | ALL      | -        |
|                    |                                  |          |              |          |          |
| Advice Type        | GATEWAYMSG                       | *        | Address Type | Uploader | <b>•</b> |
|                    |                                  |          |              |          |          |
| Static Data Filter |                                  |          | Gateway      | Uploader | <b>*</b> |
| _                  |                                  |          |              |          |          |
| Save               |                                  |          |              |          |          |
|                    |                                  |          |              |          |          |
| 🔽 Send             | Sender By Method                 | 🔽 Sender | By Gateway   |          |          |
| GatewayUploade     | rDocumentSender class will be ca | alled    |              |          |          |

| Message Status | Product<br>Type | Advice Type | Address<br>Type | Gateway  | Send | Sender<br>By<br>Gateway |
|----------------|-----------------|-------------|-----------------|----------|------|-------------------------|
| PENDING        | ALL             | GATEWAYMSG  | Uploader        | Uploader | Ø    | Ø                       |
| PENDING_TRADE  | ALL             | GATEWAYMSG  | Uploader        | Uploader | Ø    | $\square$               |
| PENDING_VALID  | ALL             | GATEWAYMSG  | Uploader        | Uploader | V    | $\square$               |
| COMPLETED      | ALL             | GATEWAYMSG  | Uploader        | Uploader | Ø    |                         |

When the domain PersistBOMessageForStaticDataErrors is set to true, SenderEngine generates NACK for rejected static data.

#### Avoid repeating NACK in case of Kick-Off (For Local Mode only)

To stop from sending repeating ack/nack each time message is reprocessed (and still fails), user needs to create a static filter that checks for message action is not 'REJECT', as shown below. This static filter must be set onto the MessageSenderConfig of GATEWAYMSG for message status as 'PENDING'.



| ļ | 💋 Static Dat   | Static Data Filter Window [150014/BLOOMBERGFIT/calypso_user] |                     |        |                   |             |  |  |  |  |  |  |
|---|----------------|--|---------------------|--------|-------------------|-------------|--|--|--|--|--|--|
|   | Name           | Stop_Duplicate_Nack  |                     |        |                   |             |  |  |  |  |  |  |
|   | External Ref.  |  |                     |        |                   |             |  |  |  |  |  |  |
| · | Comment        |  |                     |        |                   |             |  |  |  |  |  |  |
|   | Groups         | ANY  |                     |        |                   |             |  |  |  |  |  |  |
|   | 🏹 🖓 Criteri    | э  |                     |        |                   | 🜔 Simulate  |  |  |  |  |  |  |
|   |                | Attribute  | Criteria            |        | Filter Value(s)   |             |  |  |  |  |  |  |
|   | Message Actior | 1  | ▼ NOT_IN            | Add    | REJECT            |             |  |  |  |  |  |  |
| I |                |  |                     |        |                   |             |  |  |  |  |  |  |
|   | ÷              | Load O Pend  | ling Modifs 🛛 🞦 New | 🔚 Save | e as 🛛 😨 Delete 🕵 | Usage Close |  |  |  |  |  |  |

**D** Note: For Sender Engine please refer to documentation for setting up SenderEngine.

#### 2.6.7 Running Multiple Instances of the Import Message Engine

The Data Uploader module can read messages from external messaging systems using the JMS API through the Import Message engine.

By default, the Import Message engine uses config = Uploader based on the property file "calypso uploader config.properties", which indicates the queue names to connect to.

But it is also possible to run the Import Message Engine using config = TradeUploader. Additionally, multiple instances of the import message can be configured, each with its own property file to connect to different messaging systems. This allows for handling of messages from different sources, such as, Kafka, IBMMQ, or ActiveMQ, without needing custom classes.

#### **Property File**

Add the property file

For ex:

calypso\_tradeuploader\_config.properties.

#### Domain values

Add the config name to the domain "gateway" so that when sending acknowledgements for that message, the IEAdapterConfig is initialized from the message's gateway, and the acknowledgement will be sent to the proper queue.

| Name:          | gateway       |  |
|----------------|---------------|--|
| <u>V</u> alue: | TradeUploader |  |
| Comment:       |               |  |



Add the config name to the domain "addressMethod".

| Name:          | addressMethod |
|----------------|---------------|
| <u>V</u> alue: | TradeUploader |
| Comment:       |               |

#### Engine Config

Add an engine configuration:

#### TradeUploader

| onable to edit a running engile. Displayi | ing in read-only mode.                |                               |                  |                      |
|---|---------------------------------------|-------------------------------|------------------|----------------------|
| Engine Name: 🕜                            | Engine ID:                            | Max Queue Size: 🛛 🛛 Ma        | ax Batch Size: 🔞 | Number of Threads: 6 |
| TradeUploaderImportMessageEngine          | 116                                   |                               |                  |                      |
| Engine Class:                             |                                       | Event Pool Policy: 😢          | Pricing Envi     | ronment: 🕜           |
| com.calypso.tk.engine.UploadImportMe      | ssageEngine                           | UploaderImportMessageEr       | ngine 🔻          | T                    |
| Display Name: 🕜                           | Application Type:                     | Save settle position changes: | 0                |                      |
| TradeUploaderImportMessageEngine          | EngineServer                          |                               |                  |                      |
| Description:                              |                                       | Configuration attributes      |                  |                      |
|   |                                       | PROJECTED_DAYS                |                  |                      |
| Persisted Event Configuration:            |                                       | REVERSAL_CRE                  |                  |                      |
| PSEventAccountBilling                     | •                                     | STARTUP                       |                  |                      |
| PSEventUpload                             | <b>^</b>                              | TIMEOUT_RESTART               |                  |                      |
|   |                                       | USE_BOOK_PRICING_ENV          |                  |                      |
|   | -                                     | VALUATION_TIMES               |                  |                      |
| Event Filters:                            |                                       | VALUATION_TIMEZONES           |                  |                      |
| AllTransfersKnownEventFilter              | •                                     | VERSION_CHECK                 |                  |                      |
|   | 1                                     | XFER_CHECK_FIRST              |                  |                      |
|   |                                       | XFER_NEVER_BV                 |                  |                      |
| Engine Manager Configuration              | Start on Startun                      | XFER_NEXT_EVENT               |                  |                      |
| engineserver V                            | <ul> <li>Start on Startup.</li> </ul> | XFER_PAST_GENERATION          |                  |                      |
|   |                                       | XFER_USE_AUTOMATIC_ACCO       | UNT              |                      |
|   |                                       | XFER_USE_MONEYDIFF            |                  |                      |
|   |                                       | XFER_USE_POS_AGGREGATIO       | N_ONLY           |                      |
|   |                                       | XFER_USE_REVERSE              |                  |                      |
|   |                                       | confin                        | Tradallala       | adar                 |



. . .

| Engine Configuration                  |                          |                       |                         |               |     |
|---------------------------------------|--------------------------|-----------------------|-------------------------|---------------|-----|
|                                       |                          |                       |                         |               |     |
| Inable to edit a running engine. Dis  | splaying in read-only mo | ide,                  |                         |               |     |
| igine Name: <table-cell></table-cell> | 2                        | Max Queue Size:       | Max Batch Siz           | ze: 🛛         |     |
| ipioad import message Engine .        | Engine ID:               | Number of Thread      | is: 10 Event Pool Polic | v: Ø          |     |
|                                       | 421027                   |                       |                         | ~             |     |
| gine Class:                           | 80                       | _                     | Pricing Environm        | nent: 0       |     |
| om.calypso.tk.engine.UploadIm         | portMessageEngine        |                       |                         | <u>·</u>      |     |
| splay Name: 👽                         | 3                        | Save settle positio   | on changes: 👽           |               |     |
| piede import message Engine :         | Application Tv           | De: Configuration att | ributes                 |               |     |
|                                       | EngineServe              | STARTIN               |                         |               |     |
| scription:                            |                          | TIMEOUT RESTAR        | π                       |               |     |
|                                       |                          | LISE BOOK PRICE       | NG ENV                  |               |     |
| virted Event Configuration:           |                          | VALUATION TIME        | <                       |               |     |
| ollateralWebEvent                     | ~                        | VALUATION TIME        | 70NES                   |               |     |
| EventTrade                            |                          | A VERSION CHECK       | LUTILI                  |               |     |
| EventUpload                           |                          | VEED CHECK EID        | CT                      |               |     |
|                                       |                          | YEER NEVER BY         | /                       |               |     |
| ent Filters:                          |                          | YEER NEYT EVEN        | т                       |               |     |
| TransfersKnownEventFilter             | ~                        | VEED DAST GENE        | RATION                  |               |     |
|                                       |                          | A YEER DOS AGOD       | EGATION NAME            |               |     |
|                                       |                          |                       | NATIC ACCOUNT           |               |     |
|                                       |                          | VEED USE MONE         | VDICE                   |               |     |
| ne Manager Configuration:             | Start on Star            | VEED USE DOS A        | CORECTION ONLY          |               | - 1 |
|                                       |                          | VEED LISE DEVED       | OGREGATION_ORLY         |               |     |
|                                       |                          | APER_USE_REVEN        | GE .                    | Heleader      | - 8 |
|                                       |                          | adapterconfig         |                         | Oploader      |     |
|                                       |                          | coning                |                         | Opioaderkaika |     |

#### IBMMQ

| C | Adenza        | Cal | ypso 🖪 | bob - Put yo | ur peo     | ESS - Home    | Calypso Internal Do  | JenkinsCore     | Test Repository - Ca | 🙆 Last 🛙 |
|---|---------------|-----|--------|--------------|------------|---------------|----------------------|-----------------|----------------------|----------|
| 4 | $\rightarrow$ | C D | 0      | ocalhost:81  | 00/admin/r | nanage/editEn | gine?engineName=Uple | oader%20Import% | 620Message%20Engine% | 203      |





#### ActiveMQ

| ← → C ⋒ O localho  | st:8100/admin/manage/editE  | ngine?engineName=Up | loader%20Import% | 20message%20engine2  |        |
|--|---|---------------------|------------------|----------------------|--------|
| Calypso 🖪 bob - P  | ut your peo 🖋 ESS - Home  | Calypso Internal Do | JenkinsCore      | Test Repository - Ca | 🙆 Last |
| ierver• Metrics• Manage• Profile<br>Engine Configuration         | er • Manitoring • Logs  |                     |                  |                      |        |
| Unable to edit a running engine. Displaying in read-only mode    | N   |                     |                  |                      |        |
| Engine Name: 0<br>Uploader Import message engine2<br>Engine ID:  | Max Queue Size:  Max Batch Size: Max Batch Size: Number of Threads:           | 0                   |                  |                      |        |
| Engine Class:<br>com calypso tk engine UploadimportMessageEngine | Pricing Environmen  | × 0                 |                  |                      |        |
| Jisplay Name: 0<br>Uploader Import message engine2               | Save settle position changes: 0   |                     |                  |                      |        |
| Application Type<br>EngineServer                                 | e Configuration attributes<br>STARTUP<br>TIMEOUT_RESTART                      | -                   |                  |                      |        |
| Versisted Event Configuration:<br>Collateral/WebEvent            | USE_BOOK_PRICING_ENV<br>VALUATION_TIMES<br>VALUATION_TIMEZONES                |                     |                  |                      |        |
| PSEventUpload  | VERSION_CHECK<br>XFER_CHECK_FIRST<br>XFER_NEVER_BY                            |                     |                  |                      |        |
| vent Filters:<br>AllTransfersKnownEventFilter                    | XFER_NEXT_EVENT<br>XFER_PAST_GENERATION                                       |                     |                  |                      |        |
| engine Manager Configuration: Start on Startup                   | XFER_US_AUTOMATIC_ACCOUNT<br>XFER_USE_AUTOMATIC_ACCOUNT<br>XFER_USE_MONEYDIFF |                     |                  |                      |        |
| engineserver 🕶 🗌   | XFER_USE_POS_AGGREGATION_ONLY  VFER_USE_REVERSE                               |                     |                  |                      |        |
|  | adapterConfig   | Uploader            |                  |                      |        |
|  | convig  | opioaderactiveing   |                  |                      |        |

#### Sender Config

Add a Sender Configuration as below:

| 💋 Message Sende    | r Config                        |           |              |               |          |
|--------------------|---------------------------------|-----------|--------------|---------------|----------|
| Sender Config Copy | y Config                        |           |              |               |          |
|                    |                                 |           |              |               |          |
|                    |                                 |           |              |               |          |
| Message Status     | PENDING                         | <b>_</b>  | Product Type | ALL           | <b>_</b> |
|                    |                                 |           |              | ·             |          |
|                    |                                 |           |              |               |          |
| Advice Type        | GATEWAYMSG                      | <b>–</b>  | Address Type | TradeUploader | <b>_</b> |
|                    |                                 |           |              |               |          |
| Static Data Filter |                                 |           | Gateway      | TradeUploader | *        |
|                    |                                 |           |              |               |          |
|                    |                                 |           |              |               |          |
| Save               |                                 |           |              |               |          |
|                    |                                 |           |              |               |          |
|                    |                                 |           |              |               |          |
| E Sand             | Sender By Method                | E Sandar  | Pu Cataway   |               |          |
| je sena            | j sender by Method              | Jender    | by Gateway   |               |          |
| GatewayTradeUp     | loaderDocumentSender class will | be called |              |               |          |

You can now run each instance of the Import Message engine as needed.



#### 2.6.8 Using the Update Manager Engine

The Update Manager engine is only needed for uploadMode=Local, and persistMessages=All or persistMessages=Failure.

It handles the reprocessing of the failed messages that are reprocessed by applying REPROCESS action.

#### Engine Configuration

The Update Manager engine is configured in the Engine Manager of Web Admin: event subscription and engine parameters.

It subscribes to PSEventUploadReprocess events.

You may need to add this engine if it is not available for configuration: Create a new engine called UpdateManagerEngine with class name com.calypso.tk.engine.UpdateManagerEngine.

Event filter = UpdateManagerEngineEventFilter

Please refer to Calypso Web Admin documentation for complete details.

#### Starting the Update Manager Engine

The Update Manager engine can be started from the Engine Manager in Web Admin.

Please refer to Calypso Web Admin documentation for complete details.

#### Reprocessing Bloomberg VCON Messages for Missing Bonds

Importing Bond definitions using Bloomberg DL engine in case the message fails to upload because of missing Bond.

To achieve this, we require the following engines:

- UpdateManagerEngine
- Bloomberg Engine

And following message rules:

- UploadReprocess
- RequestBloomberg

The Bloomberg Engine is used for fetching the Bond:



| Engine Manager                         |           |  |                            |             |                        |                |             |                        |     |                              |                  |                              |           |
|--|-----------|--|----------------------------|-------------|------------------------|----------------|-------------|------------------------|-----|------------------------------|------------------|------------------------------|-----------|
| This page is to manage and monitor eng | ines as v | vell as display runtime statistics as pr | eriodically reported by th | e engine. T | hese statistics are up | pdated every 3 | 0 seconds.  |                        |     |                              |                  |                              |           |
|  |           |  |                            |             |                        |                |             |                        |     | Create New                   | Engine Manager ( | Configuration Create New E   | ngine     |
| Engine Name                            |           | Manager Configuration Name               | Application Type           | Startup     | Host                   | Consumed       | Engine Pool | Total Unconsumed Queue | Bad | Last Consumed                | Oldest           | Last Updated                 | Mode      |
| Accounting Engine                      |           | engineserver                             | EngineServer               | 21          | IE-DUB-PF1G2LLP        | 1              | 0           | 0                      | 0   | Wed Nov 16 15:17:20 GMT 2022 | no events        | Wed Nov 16 15:19:29 GMT 2022 | Real-Time |
| Balance Engine                         | 0         | engineserver                             | EngineServer               |             |                        |                |             |                        |     |                              |                  |                              | Stopped   |
| Billing Engine                         |           | engineserver                             | EngineServer               | 21          | IE-DUB-PF1G2LLP        | 55             | 0           | 0                      | 0   | Wed Nov 16 15:19:19 GMT 2022 | no events        | Wed Nov 16 15:19:29 GMT 2022 | Real-Time |
| Bloomberg Engine                       |           | engineserver                             | EngineServer               |             | IE-DUB-PF1G2LLP        | 5              | 0           | 0                      | 7   | Wed Nov 16 15:19:14 GMT 2022 | 15 seconds ago   | Wed Nov 16 15:19:29 GMT 2022 | Real-Time |

In Engine Configuration, the UpdateManagerEngine is configured for reprocessing the Pending GATEWAYMSG:

| Engine Configuration                                       |           |                              |                 |   |
|--|-----------|------------------------------|-----------------|---|
| Unable to edit a running engine. Displaying in read-on     | ily mode. |                              |                 |   |
| Engine Name: 🔮<br>UpdateManagerEngine<br>Engin             | ne ID:    | Max Queue Size: Anax Batc    | Policy:         | 1 |
| Engine Class:<br>com.calypso.tk.engine.UpdateManagerEngine | 012       | Pricing Envi                 | ronment: 0      | 6 |
| Display Name: 9  |           | v                            |                 |   |
| Opdate Manager Engine                                      | ion Type: | Configuration attributes     |                 |   |
| Engines  | Server    | Attribute Name               | Attribute Value |   |
| Description:   |           | adapterConfig                |                 |   |
|  |           | BALANCE_MODE                 |                 |   |
| Persisted Event Configuration:                             |           | CLASS_NAME                   |                 |   |
| PSEventAccountBilling ~                                    |           | config                       |                 |   |
| PSEventDataPersistor                                       | -         | DateType                     |                 |   |
| PSEventTrade   |           | DISPLAY NAME                 |                 |   |
| PSEventUploadReprocess                                     | *         | EVENT ORDER                  |                 |   |
| Event Filters:   | 1         |                              |                 |   |
| AllTransfersKnownEventFilter                               |           |                              |                 |   |
| UpdateManagerEngineEventFilter                             | *         | EXCLUDE_STATUS               |                 |   |
|  |           | HANDLE_FUTURE_LIQ_CASH_FLOWS |                 |   |
|  | ~         | IGNORE_ACTION                |                 |   |
| Engine Manager Configuration: Start or                     | Startup:  | INSTANCE_NAME                |                 |   |
| engineserver 🗸 🗌   |           | INV_MAX_POSITION             |                 |   |
| 4  | - F       | LIQUIDATION_TIMEOUT          |                 |   |

UploadReprocess and RequestBloomberg message rule in the workflow



| Workflows P  | A Workflow Diagra | m WorkFlow Table | Kick Off / Cut Off | Breaks Closin | 0       |           |         |                  |             |                    |                    |            | Action Details       |                    | ę            |
|--|-------------------|------------------|--------------------|---------------|---------|-----------|---------|------------------|-------------|--------------------|--------------------|------------|----------------------|--------------------|--------------|
| 18 CA 14   |                   | 101              |                    |               |         |           |         |                  |             |                    |                    |            | Id 36700             |                    | 1            |
| 0-   |                   |                  |                    |               |         |           |         |                  |             |                    |                    | 00         | Orig Status PENDING  |                    | ~ 5          |
| 4 Workflows  | Id Orig Status    | Action           | Resulting Status   | Create Task   | Use STP | Same User | Rules   | Needs Man. Auth. | Use Comment | Kick Off / Cut Off | Static Data Filter | Transition | Action Name REPROD   | ESS                | ~ 8          |
| AMBlockOrder   | 36696 COMPLETED   | RESEND           | COMPLETED          | On Failure    | False   | True      |         | False            | False       | False              |                    |            | Result Status REPROC | ESSING             |              |
| AMOrder  | 36697 NONE        | NEW              | PENDING            | On Failure    | True    | True      |         | False            | False       | False              | UploadModeLocal    |            | Headin Status (10)   | 200210             |              |
| DecSuppOrder   | 36698 PENDING     | CANCEL           | CANCELED           | On Failure    | False   | True      |         | False            | False       | False              |                    |            | Create               | ask On Failure 🖓   |              |
| Order  | 36699 PENDING     | COMPLETE         | COMPLETED          | On Failure    | True    | True      |         | False            | False       | False              | UploadMessageC     |            |                      | CT0                |              |
| PSEventMessage   | 36700 PENDING     | REPROCESS        | REPROCESSING       | On Failure    | False   | True      | Upload. | False            | False       | False              |                    |            | unuse                | SIP                |              |
| PO:ALL   | 36701 PENDING     | UPDATE           | PENDING            | On Failure    | False   | True      |         | False            | False       | False              |                    |            | 🧏 🗌 Gen              | erate Intermediary | Event        |
| A Product:ALL  | 36702 REPROCESS   | . CANCEL         | CANCELED           | On Failure    | False   | True      |         | False            | False       | False              |                    |            | 0.                   |                    |              |
| Subtype:ALL     Subtype:ALL  | 36703 REPROCESS   | . COMPLETE       | COMPLETED          | On Failure    | False   | True      |         | False            | False       | False              | UploadMessageC     |            | "12, Priority        | 0                  |              |
| Subtype:CONFIRM  | 36704 REPROCESS.  | . REJECT         | PENDING            | On Failure    | False   | True      |         | False            | False       | False              |                    |            | A Nee                | Manual Authoriza   | tion         |
| Subtype:DKCONF   |                   |                  |                    |               |         |           |         |                  |             |                    |                    |            |                      |                    |              |
| Subtype:DTCCNovationReply     Subtype:DTCCNovationReply                        |                   |                  |                    |               |         |           |         |                  |             |                    |                    |            | com                  | ment               |              |
| Subtype:DTCC_CONFIRM   |                   |                  |                    |               |         |           |         |                  |             |                    |                    |            | 🚺 Diffe              | rent User          |              |
| Subtype:GATEWAYMSG   |                   |                  |                    |               |         |           |         |                  |             |                    |                    |            |                      | and taking         |              |
| Subtype:INCOMING   |                   |                  |                    |               |         |           |         |                  |             |                    |                    |            | T Pres               | erred Action       |              |
| Subtype:INCOMINGCONF     Subtype:INC EXCONERN                                  |                   |                  |                    |               |         |           |         |                  |             |                    |                    |            | A Ued                | ste Only           |              |
| Subtype:INC_SETTALLEGE   |                   |                  |                    |               |         |           |         |                  |             |                    |                    |            | Rules                |                    | ⊇聯眼團●        |
| <ul> <li>Subtype:IncomingNovationReques</li> <li>Subtype:PAYMENTMSG</li> </ul> |                   |                  |                    |               |         |           |         |                  |             |                    |                    |            | Name                 | Rule Param         | Task Comment |
| Subtype:PLATFORMMSG  |                   |                  |                    |               |         |           |         |                  |             |                    |                    |            | UploadReprocess      |                    |              |
| Subtype:RECEIPTMSG   |                   |                  |                    |               |         |           |         |                  |             |                    |                    |            | RequestRicomberg     |                    |              |
| <ul> <li>Subtype:SEC_DELIVERYMSG</li> </ul>                                    |                   |                  |                    |               |         |           |         |                  |             |                    |                    |            |                      |                    |              |
| Subtype:SEC_RECEIPTMSG     Subtype:SW/ETCONEIPM                                |                   |                  |                    |               |         |           |         |                  |             |                    |                    |            |                      |                    |              |
| Subtype:SW_MSG   |                   |                  |                    |               |         |           |         |                  |             |                    |                    |            |                      |                    |              |
| Subtype:SWIFTCONFIRM     Subtype:SW_MSG     Subtype:III: OADREOUESTMSC         |                   |                  |                    |               |         |           |         |                  |             |                    |                    |            |                      |                    |              |
| <ul> <li>Subtype:UPLOADREQUESTMSG</li> <li>Subtype:UPLOADSOURCEMSG</li> </ul>  |                   |                  |                    |               |         |           |         |                  |             |                    |                    |            |                      |                    |              |
| PSEventTrade   |                   |                  |                    |               |         |           |         |                  |             |                    |                    |            | N                    |                    |              |
| PO:ALL   |                   |                  |                    |               |         |           |         |                  |             |                    |                    |            | 3                    |                    |              |

#### Example:

We are uploading a VCON message, and it fails because of missing bond in the system:

|               | C           |             |         |                 |                       |                   |                |             |                         |  |   |
|---------------|-------------|-------------|---------|-----------------|-----------------------|-------------------|----------------|-------------|-------------------------|--|---|
| irce:         | BloombergFI | T V Format: | VCON    | ✓ Select File / | Folder C:\calyps      | o\calypso-16.1.0. | 77\samples\VCO | N_SAMPLE.fb | x 🎝 Refresh 🛃 Upload    |  |   |
| sist Message: | All         | ~           |         |                 |                       |                   |                |             |                         |  |   |
| VCON SAMPLE   | fix X       |             |         |                 |                       |                   |                |             |                         |  |   |
| Filter Data   |             |             |         |                 |                       |                   |                |             |                         |  |   |
| ploadStatus   | BeginString | BodyLength  | MsgType | SenderCompID    | TarCompID             | MsgSeqNum         | TarSubID       | OnBehalf    | Upload Field            | Value  | _ |
| ected         | FIX:4.4     | 1042        | AE      | MAP BLP PROD    | MAP AROT PRO          | 0 1650            | 30188866       | LVCONSRV    | BeginString             | FIX.4.4  |   |
|               |             |             |         |                 | terres and the second |                   |                |             | BodyLength              | 1042   |   |
|               |             |             |         |                 |                       |                   |                |             | MsgType                 | AE   |   |
|               |             |             |         |                 |                       |                   |                |             | SenderCompID            | MAP BLP PROD   |   |
|               |             |             |         |                 |                       |                   |                |             | TargetCompID            | MAP ARDT PROD  |   |
|               |             |             |         |                 |                       |                   |                |             | MsgSegNum               | 1650   |   |
|               |             |             |         |                 |                       |                   |                |             | TargetSubID             | 30188866   |   |
|               |             |             |         |                 |                       |                   |                |             | OnBehalfOfCompID        | I.VCONSRV  |   |
|               |             |             |         |                 |                       |                   |                |             | SendingTime             | 20210414-13:08:51  |   |
|               |             |             |         |                 |                       |                   |                |             | TransactTime            | 20210414-13:08:51  |   |
|               |             |             |         |                 |                       |                   |                |             | PreviouslyReported      | N  |   |
|               |             |             |         |                 |                       |                   |                |             | LastPx                  | 80.375   |   |
|               |             |             |         |                 |                       |                   |                |             | TradeReportID           | 00064fec:27f3:3f75:6071422b                              |   |
|               |             |             |         |                 |                       |                   |                |             | LastQty                 | 2500000  |   |
|               |             |             |         |                 |                       |                   |                |             | PriceType               | 1  |   |
|               |             |             |         |                 |                       |                   |                |             | MatchStatus             | 1  |   |
|               |             |             |         |                 |                       |                   |                |             | SettIDate               | 20210416   |   |
|               |             |             |         |                 |                       |                   |                |             | TradeReportTransType    | 0  |   |
|               |             |             |         |                 |                       |                   |                | r           | SecondaryTradeReportID  | 6950999687244808241_3                                    |   |
|               |             |             |         |                 |                       |                   |                | L           | Product                 | 3  | _ |
|               |             |             |         |                 |                       |                   |                |             | Status Field            | Value  |   |
|               |             |             |         |                 |                       |                   |                |             | ExternalRef             | BFIT_CALYPSO_6950999687244808241_3                       |   |
|               |             |             |         |                 |                       |                   |                |             | ProcessingOrg           | CALYPSO  |   |
|               |             |             |         |                 |                       |                   |                |             | CalypsoTradeId          | -1   |   |
|               |             |             |         |                 |                       |                   |                |             | Status                  | Rejected   |   |
|               |             |             |         |                 |                       |                   |                |             | Action                  | NEW  |   |
|               |             |             |         |                 |                       |                   |                |             | ErrorType               | InvalidData  |   |
|               |             |             |         |                 |                       |                   |                |             | ErrorCode               | 284  |   |
|               |             |             |         |                 |                       |                   |                |             | ErrorItem               | Bond Underlying Product Definition                       |   |
|               |             |             |         |                 |                       |                   |                |             | ErrorMessage            | No Product exists for the given Sec Code Name and Value. |   |
|               |             |             |         |                 |                       |                   |                |             | ErrorValue              | ISIN / US91282CDJ71                                      |   |
|               |             |             |         |                 |                       |                   |                |             | ITrado Custom Kennuorde |  |   |

We see the GATEWAYMSG will remain in Pending state as shown below:



| A Message F | enort (16/1  | 1/22 14-50-47) / FIX    |                      |                    |           |                      |           |                 |                          |          |                  |                        |                       |             |               | _                     | n x              |
|-------------|--------------|-------------------------|----------------------|--------------------|-----------|----------------------|-----------|-----------------|--------------------------|----------|------------------|------------------------|-----------------------|-------------|---------------|-----------------------|------------------|
| incosoge i  | coport (10/1 | 1/66 14:35:41/7104      |                      |                    |           |                      |           |                 |                          |          |                  |                        |                       |             |               |                       |                  |
| Report Data | View Export  | Market Data Process     | Utilities Help       |                    |           |                      |           |                 |                          |          |                  |                        |                       |             |               |                       |                  |
|             | 3            |                         |                      |                    |           |                      |           |                 |                          |          |                  |                        |                       |             |               |                       |                  |
| Criteria    |              |                         |                      |                    |           |                      |           |                 |                          |          |                  |                        |                       |             |               |                       |                  |
| MESSAGE_ID  | Msg Status   | Msg_Attr.UploadObject   | Msg_Attr.UploadObje  | ectExternalRef     | Trade Id  | EVENT_TYPE           | Family    | Product Type    | MESSAGE_TYPE             | Sender   | Sender.Role      | Sender.Contact Type    | Sender.Address Code   | Receiver    | Receiver.Role | Receiver.Contact Type | Receiver_Address |
| 79076       | COMPLETED    | BloombergFITMessage     |                      |                    | 67431     | X_UPLOADSOURCEMSG    |           |                 | UPLOADSOURCEMSG          | CLIENT   |                  |                        |                       | CALYPSO     |               |                       |                  |
| 79077       | COMPLETED    | CalypsoTrade            | BFIT_CALYPSO_6950    | 999687244808241_2  | 67431     | EX_GATEWAYMSG        |           |                 | GATEWAYMSG               | CLIENT   |                  |                        |                       | CALYPSO     |               |                       |                  |
| 79078       | COMPLETED    | BloombergFITMessage     | DETT. CALVERO, COFOL | 0000070440000041 0 | 0         | EX_UPLOADSOURCEMSG   |           |                 | UPLOADSOURCEMSG          | CLIENT   |                  |                        |                       | CALYPSO     |               |                       |                  |
| 79075       | PENDING      | Calypsorrade            | 8P11_CALTPSO_6950    | 999087244808241_3  | 0         | CA_GATEWATMSG        |           |                 | GATEWATMSG               | CLIENT   |                  |                        |                       | CALTPSO     |               |                       |                  |
| A Messa     | ae Viewer    |                         |                      |                    |           |                      |           |                 |                          |          |                  |                        |                       |             |               | -                     | D X              |
|             |              |                         |                      |                    |           |                      |           |                 |                          |          |                  |                        |                       |             |               |                       |                  |
| Message     |              |                         |                      |                    |           |                      |           |                 |                          |          |                  |                        |                       |             |               |                       |                  |
| Main Audit  | Grouping T   | ransfers Tasks Advice L | ocuments Matching A  | Audit Matching     |           |                      |           |                 |                          |          |                  |                        |                       |             |               |                       |                  |
| Task Id T   | rade Id Tra  | deStatus Date & Time    |                      | Book Task Status   | Task Owne | r Comment            |           |                 |                          |          |                  |                        |                       |             | Prio          | rity Action Performed | Resulting Status |
| 185125      | 0            | 16/11/22 15:13          | 7:57.648 o'clock GMT | NEW                |           | type=InvalidData~sul | type=~ite | m=Bond Under    | ying Product Definition- | -00284=N | o Product exists | for the given Sec Code | Name and Value.~value | =ISIN / US9 | 1282CDJ71 NOR | MAL COMPLETE          |                  |
| 185124      | 0            | 16/11/22 15:1           | 7:57.654 o'clock GMT | NEW                |           | Filter UploadMessage | Complete  | d rejects Messa | e                        |          |                  |                        |                       |             | NOR           | MAL COMPLETE          |                  |
|             |              |                         |                      |                    |           |                      |           |                 |                          |          |                  |                        |                       |             |               |                       |                  |
|             |              |                         |                      |                    |           |                      |           |                 |                          |          |                  |                        |                       |             |               |                       |                  |
|             |              |                         |                      |                    |           |                      |           |                 |                          |          |                  |                        |                       |             |               |                       |                  |
|             |              |                         |                      |                    |           |                      |           |                 |                          |          |                  |                        |                       |             |               |                       |                  |
|             |              |                         |                      |                    |           |                      |           |                 |                          |          |                  |                        |                       |             |               |                       |                  |
|             |              |                         |                      |                    |           |                      |           |                 |                          |          |                  |                        |                       |             |               |                       |                  |
|             |              |                         |                      |                    |           |                      |           |                 |                          |          |                  |                        |                       |             |               |                       |                  |
|             |              |                         |                      |                    |           |                      |           |                 |                          |          |                  |                        |                       |             |               |                       |                  |
|             |              |                         |                      |                    |           |                      |           |                 |                          |          |                  |                        |                       |             |               |                       |                  |
|             |              |                         |                      |                    |           |                      |           |                 |                          |          |                  |                        |                       |             |               |                       |                  |

Now the GATEWAYMSG need to be reprocessed. On reprocessing the GATEWAYMSG, RequestBloomberg Message rule will create an even for Bloomberg Engine to fetch the Bond.

However, the GATEWAYMSG will remain in Pending state because still the Bond is not present in the system:



In the next reprocess of the GATEWAYMSG, the message will move to Completed state as bond is imported by the Bloomberg Engine and Trade is created:

|                          |   |  |   |  |  |  |  |   |   | -  | o ×   |
|--------------------------|---|--|---|--|--|--|--|---|---|--|---|
|                          |   |  |   |  |  |  |  |   |   |  |   |
|                          |   |  |   |  |  |  |  |   |   |  |   |
|                          |   |  |   |  |  |  |  |   |   |  |   |
| Trade Id EVENT_TYPE      | Family P  | Product Type M   | IESSAGE_TYPE  | Sender   | Sender.Role  | Sender.Contact Type  | Sender Address Code  | Receiver  | Receiver.Role   | Receiver.Contact Type  | Receiver.Address  |
| 67431 EX UPLOADSOURCEMSG | 1   | UP   | PLOADSOURCEMSG  | CLIENT   |  |  |  | CALYPSO   | 1   |  |   |
| 2 67431 EX_GATEWAYMSG    |   | GA   | ATEWAYMSG   | CLIENT   |  |  |  | CALYPSO   |   |  |   |
| 67434 EX_UPLOADSOURCEMSG |   | UP   | PLOADSOURCEMSG  | CLIENT   |  |  |  | CALYPSO   |   |  |   |
| 3 67434 EX_GATEWAYMSG    |   |  |   |  |  |  |  |   |   |  |   |
|                          |   |  |   |  |  |  |  |   |   |  |   |
|                          |   |  |   |  |  |  |  |   |   |  |   |
|                          | Trade M         EVENT_TYPE           6743115_UPCORDSOURCEMSG         6743115_UPCORDSOURCEMSG           6743115_UPCORDSOURCEMSG         6743115_UPCORDSOURCEMSG           3         6743115_CORTEWINSG | Trole M         EVENT_TYPE         Femily           47431EV_UPLOADSOURCENSE         2         47431EV_GATEWAYKSE         2           47441EV_UPLOADSOURCENSE         2         47441EV_UPLOADSOURCENSE         2           47441EV_UPLOADSOURCENSE         2         47441EV_UPLOADSOURCENSE         2           47441EV_UPLOADSOURCENSE         2         47441EV_UPLOADSOURCENSE         2 | Trade M         EVENT_TYPE         Femily         Froduct Type         M           67431EX_UPLOADSOURCEMEG         U         U         0431EX_CATEWANES         G         G         U         0431EX_CATEWANES         G         U         U         0431EX_CATEWANES         G         U         U         0431EX_CATEWANES         G         U         U         U         0431EX_CATEWANES         G         U | Trade M         EVENT_TYPE         Femily         Product Type         MESSAGE_TYPE           67431 EX_UPLOADSOURCEMSG         UPLOADSOURCEMSG         UPLOADSOURCEMSG         CATEWAYNEG           67445 EX_UPLOADSOURCEMSG         UPLOADSOURCEMSG         UPLOADSOURCEMSG         UPLOADSOURCEMSG           67445 EX_UPLOADSOURCEMSG         UPLOADSOURCEMSG         UPLOADSOURCEMSG         UPLOADSOURCEMSG           3         67031 EX_UPLOADSOURCEMSG         UPLOADSOURCEMSG         UPLOADSOURCEMSG | Trole M EVENT_TYPE Family Product Type MESSAGE_TYPE Sender<br>67431EX_UPLOADSOURCEMISG UPLOADSOURCEMISG CLEMIT<br>2 67431EX_CATENUMISG CATENUMISG CLEMIT<br>67494EX_UPLOADSOURCEMISG UPLOADSOURCEMISG CLEMIT<br>3 eVGHEX_CATENUMISG CLEMIT | Trole M         EVENT_TYPE         Family         Product Type         MESSAGE_TYPE         Sender         Sender | Trode M         EVENT_TYPE         Family         Product Type         MESSAGE_TYPE         Sender         Sender.Role         Sender.Contact Type           67431 [DC_URLOADSOURCENSS         URLOADSOURCENSS (CLENT         URLOADSOURCENSS (CLENT         Sender.Sontact Type           37431 [DC_URLOADSOURCENSS         CATEWAYNSS         CATEWAYNSS         CLENT         Sender.Sontact Type           3         SASH DC_GATEWAYNSS         CATEWAYNSS         CLENT         Sender.Sontact Type | Trode M         EVENT_TYPE         Femily         Product Type         MESSAGE_TYPE         Sender         Sender.Role         Sender.Contact Type         Sender.Address Code           47431EV_UFLOADSOURCENSG         URIQADSOURCENSG         URIQADSOURCENSG         CLEINT         2         AVAILEV_UFLOADSOURCENSG         2         AVAILEV_UFLOADSOURCENSG         2         2         2         2         2         2         2         2 | Trole M         EVENT_TYPE         Family         Product Type         MESSAGE_TYPE         Sender         Sender.Role         Sender.Contact Type         Sender.Address.Code         Receiver           67431 EV_URLAADSOURCEMSG         URLOADSOURCEMSG         URLOADSOURCEMSG         CALENT         CALYPSO           2         67431 EV_URLOADSOURCEMSG         CALENT         CALYPSO         CALYPSO           3         67431 EV_URLOADSOURCEMSG         CLENT         CALYPSO         CALYPSO           3         67431 EV_URLOADSOURCEMSG         CLENT         CALYPSO         CALYPSO | Trode M         EVENT_TYPE         Family         Product Type         MESSAGE_TYPE         Sender         Sender.Role         Sender.Contact Type         Sender.Address Code         Receiver         Receiver | Trole M         EVENT_TYPE         Family         Product Type         Sender         Sender-Role         Sender-Address         Code         Receiver-Role         Receiver-Contact Type           67431 [DC_UPLOADSOURCEMSG         UPLOADSOURCEMSG         UPLOADSOURCEMSG         UPLOADSOURCEMSG         CULINT         CALTRON           2         67431 [DC_UPLOADSOURCEMSG         CULINT         CALTRON         CALTRON         CALTRON           3         6764 [DC_UPLOADSOURCEMSG         CULINTSS         CALTRON         CALTRON         CALTRON |

The reprocessing of the Pending GATEWAYMSG can be done manually or automatically by configuring kickoff and cutoff for the WF transition.

With the help of kickoff and cutoff the complete process will work without manual intervention.

## 2.7 Using the Data Uploader Engine

The Data Uploader engine is only used for manual uploads of CSV or XML files. It starts the File Watcher application.

### 2.7.1 Property File

The file "<calypso home>/client/resources/datauploader.properties" contains the File Watcher properties - Modify as needed.



# This is the encoding scheme to be used for the INPUT (i.e. CSV) files # This feature is provided to take care of CSV files created using different # encoding schemes e.g. SHIFT JIS for Japanese INPUT ENCODING SCHEME=UTF-8 # This is the encoding scheme for the output of the CSV uploader. # The CSV uploader creates an XML file that is read by the XML uploader # The output scheme instructs the CSV uploader to create the output XML from # the CSV uploader # (input for the XML uploader) in this encoding scheme OUTPUT ENCODING SCHEME=UTF-8 # File Watcher Properties #This is the folder where we place the files. fileDir=C:\\uploader\\ interval=10000 #is the time to sleep after receiving the file before actually processing the file sleepInterval=15000 # This tells the file watcher to call the XML Uploader fileWatcherType=data #This tells the file watcher to monitor CSV & XML Files only fileFilter=csv, xml, txt csvDelimeter=, txtDelimeter=\t #If the following property is true, the transformation files will be generated in CALYPSO HOME\DataUploader folder DEBUG XML=true uploadMode=Local persistMessages=All

#### 2.7.2 Launching the Data Uploader Engine

The Data Uploader engine is configured in the Engine Manager of Web Admin: event subscription and engine parameters.

You may need to add this engine if it is not available for configuration: Create a new engine called DataUploaderEngine with class name com.calypso.tk.engine.DataUploaderEngine.

The "config" engine parameter is config = datauploader.properties

The Data Uploader engine can be started from the Engine Manager in Web Admin.



▶ Please refer to Calypso Web Admin documentation for complete details.

#### **Operating Considerations**

Based on your volume, you may need to increase the memory allocated to the Data Uploader engine (e.g., -Xmx384, -Xmx512).

With the Data Uploader engine operating, you can then place a file to import in the watched directory (specified by fileDir). At the end of the current wait interval, the File Watcher will notify the Data Uploader which will then process the file accordingly.

The Task Station will display any errors that may occur.



## 2.8 Using the Scheduled Task DATA\_UPLOADER

You can also upload files manually using the scheduled task DATA\_UPLOADER.

From the Calypso Navigator, navigate to Configuration > Scheduled Tasks (menu action scheduling.ScheduledTaskListWindow), and create a task of type DATA\_UPLOADER.

| Tesh Turay DATA UDIOADED  |  |
|---|--|
| Task Type: DATA_OPLOADER  |  |
| External Reference:   |  |
| Comments:   |  |
| Description:  |  |
| Description.  |  |
| Execution Parameters  |  |
| Attempts: 1 Retry After: 0  | minutes Expected Execution Time  |
| JVM Settings: -Xms512m -Xmx1024m  |  |
| Log Settings:   |  |
| Task Notification Options   |  |
| Send Emails Publish Business Event  | s To User: 🗸 🗸   |
| ± Common Attributes   |  |
| Task Attributes   |  |
| INPUT_FILE_LOCATION   | Cul Calvara DTUD   |
|   | C:\Calypso\DTUP  |
| INPUT_FILE  | FXSWAP_NEW_CSV.csv   |
| INPUT_FILE<br>OUTPUT_LOCATION   | C:\Calypso\DTUP<br>FXSWAP_NEW_CSV.csv<br>C:\Calypso\DTUP\Ulpoad-results  |
| INPUT_FILE<br>OUTPUT_LOCATION<br>INPUT_ENCODING   | C:\Calypso\DTUP<br>FXSWAP_NEW_CSV.csv<br>C:\Calypso\DTUP\Ulpoad-results<br>UTF-8   |
| INPUT_FILE<br>OUTPUT_LOCATION<br>INPUT_ENCODING<br>OUTPUT_ENCODING  | C:\Calybso\DTUP<br>FXSWAP_NEW_CSV.csv<br>C:\Calybso\DTUP\Ulpoad-results<br>UTF-8<br>UTF-8  |
| INPUT_FILE<br>OUTPUT_LOCATION<br>INPUT_ENCODING<br>OUTPUT_ENCODING<br>RENAME_INPUT_FILE   | C:(Calypso()) OF<br>FXSWAP_NEW_CSV.csv<br>C:(Calypso()DTUP/Ulpoad-results<br>UTF-8<br>false  |
| INPUT_FILE<br>OUTPUT_LOCATION<br>INPUT_ENCODING<br>OUTPUT_ENCODING<br>RENAME_INPUT_FILE<br>UPLOAD_SOURCE  | C:(Calypso()) OP<br>FXSWAP_NEW_CSV.csv<br>C:\Calypso\DTUP\Ulpoad-results<br>UTF-8<br>false<br>Uploader<br>Uploader   |
| INPUT_FILE<br>OUTPUT_LOCATION<br>INPUT_ENCODING<br>OUTPUT_ENCODING<br>RENAME_INPUT_FILE<br>UPLOAD_SOURCE<br>UPLOAD_FORMAT<br>DEDECT   | C:(Calypso\DTUP)<br>FXSWAP_NEW_CSV.csv<br>C:\Calypso\DTUP\Ulpoad-results<br>UTF-8<br>false<br>Uploader<br>Uploader<br>UploaderXML                          |
| INPUT_FILE<br>OUTPUT_LOCATION<br>INPUT_ENCODING<br>OUTPUT_ENCODING<br>RENAME_INPUT_FILE<br>UPLOAD_SOURCE<br>UPLOAD_FORMAT<br>PERSIST_MESSAGE<br>ENDOC   | C:(Calypso()) OF CF<br>FXSWAP_NEW_CSV.csv<br>C:(Calypso()DTUP\Ulpoad-results<br>UTF-8<br>false<br>Uploader<br>Uploader<br>UploaderXML<br>None              |
| INPUT_FILE<br>OUTPUT_LOCATION<br>INPUT_ENCODING<br>OUTPUT_ENCODING<br>RENAME_INPUT_FILE<br>UPLOAD_SOURCE<br>UPLOAD_FORMAT<br>PERSIST_MESSAGE<br>IGNORE_WARNINGS   | C:(Calypso())TUP<br>FXSWAP_NEW_CSV.csv<br>C:\Calypso\DTUP\Ulpoad-results<br>UTF-8<br>false<br>Uploader<br>Uploader<br>UploaderXML<br>None<br>true          |
| INPUT_FILE<br>OUTPUT_LOCATION<br>INPUT_ENCODING<br>OUTPUT_ENCODING<br>OUTPUT_ENCODING<br>UPLOAD_SOURCE<br>UPLOAD_SOURCE<br>UPLOAD_FORMAT<br>PERSIST_MESSAGE<br>IGNORE_WARNINGS<br>CUSTOM_TRADE_KEYWORDS_TO_PUBLISH<br>OPTTOMIL TO F_WARE DOFTSY | C:(Calypso())TUP<br>FXSWAP_NEW_CSV.csv<br>C:\Calypso\DTUP\Ulpoad-results<br>UTF-8<br>UTF-8<br>false<br>Uploader<br>Uploader<br>UploaderXML<br>None<br>true |

#### Task Attributes

- INPUT\_FILE\_LOCATION Location of the file to be imported.
- INPUT\_FILE File name.
- OUTPUT\_LOCATION Location of the converted file.
- INPUT\_ENCODING Encoding scheme to be used for the file to be imported.
- OUTPUT\_ENCODING Encoding scheme to be used for the converted file.
- RENAME\_INPUT\_FILE Select true to rename the imported file so that you know it has been imported.
- UPLOAD\_SOURCE Select the source of the file to be imported.
- UPLOAD\_FORMAT Select the format of the file to be imported.

You can also enter regular expressions for the attributes INPUT\_FILE\_LOCATION, INPUT\_FILE, OUTPUT\_LOCATION.

Examples:

INPUT\_FILE\_LOCATION = \${user.home}\datauploader\input

INPUT\_FILE = Futures\_[0-9]{1}.xml



OUTPUT\_LOCATION = \${user.home}\datauploader\output

Upon import, the following files are created:

In INPUT\_FILE\_LOCATION – "<file name>.finished" to indicate that the file has been imported.

In OUTPUT\_LOCATION:

- "<fine name>\_result.xml" to show the items successful imported.
- "<file name>\_rejected.xml" to show the rejected items.

The rejected file is created if Publishers=File,RejectedFile in "datauploader.properties".

## 2.9 Resources Files

#### IMPORTANT NOTE:

All resource files, property files and configuration files that have been modified need to be copied to <calypso home>/tools/calypso-templates/resources.

You will need to re-deploy your environment to your application servers so that they can be included.

Please refer to the Calypso Installation Guide for details on deployment.

## 2.10 Dates Format

For proper date-time conversion, a date should contain the following information: Date, Time, Timezone offset.

Example: Date = 20170602T16:12:39+05:30 (where +05:30 is the timezone offset with respect to GMT)

If the timezone is not specified, the Data Uploader uses the system timezone.

The DataUploaderEngine uses the EngineServer timezone by default.

The Data Uploader window uses the User Default timezone.

## 2.11 Fees Management

When uploading a trade amendment with no trade fees in the uploaded file, the trade fees are not removed - There is no change to the trade fees.

To remove the fees when there are no fees in the uploaded file, you need define the product types for which you want to remove fees in the domain "UploaderRemoveFeesForProduct".

If you want to preserve some fees when domain "UploaderRemoveFeesForProduct" is set, you can add the fee types to the domain "UploadPreserveFee".

Example: Cash is set in domain "UploaderRemoveFeesForProduct", and domain "UploadPreserveFee" contains FEE2. A Cash trade has 2 fees, FEE1 and FEE2, and an amendment is uploaded with no fees. After the amendment, FEE1 is removed and FEE2 is preserved.



# 3 Operations

## 3.1 Requirements

Before using the Data Uploader, the Data Server, Event Server, and Task engine must be running. Additionally, you must monitor the Task Station for errors and blocked messages.

## 3.2 Handling Blocked Messages

In the event of a blocked message, after clearing the issue, you can use the VALIDATE or LOAD action to reprocess a manually uploaded message. For messages received from an engine, use the action appropriate for that system.

For example, after clearing an issue for a blocked MarkitWire message, use the REPROCESS action to resubmit the message to the system.

For messages from external systems, refer to the specific Calypso integration guide for that system.

## 3.3 Task Station Errors

If Task Station error messages are improperly displayed, ensure that the file "ErrorBundle.properties" is present in the resources directory.

## 3.4 Trade Allocations

Allocation in the Data Uploader is performed in two modes.

- API Mode: In this mode a trade in Calypso will be allocated using the Allocation API in Calypso, and the child trades are created when the action is applied. The child trades are rekeyed (external reference is updated to match the external system) when the child trade notifications are received in calypso. This is used when external system is sending the allocation information to Calypso and we use the percentages or notionals to create the allocations.
- Non-API Mode: This use case comes across when allocations are done externally in a system and we only
  receive the block trade and the post-allocated trade. But not the actual allocation details itself. In this mode
  allocation is not be performed in Calypso using the Allocation API. Only the status of the block trade will be
  updated to ALLOCATED, by directly calling Allocate action. We will then also create the child trades in Calypso
  (which are passed from the external system) and then link up the trades.

The Uploader schema is extended to support Trade Allocations.

The Following sample xml defines that the trade is to be allocated into 2 child trades with amounts 1000 & 2000 respectively.



| <tradeallocations></tradeallocations>  |
|--|
| <tradeallocation></tradeallocation>  |
| <allocationamount>1000</allocationamount>                                    |
| <allocationpercentage></allocationpercentage>                                |
| <allocationparty>CALYPSO1</allocationparty>                                  |
| <allocationpartyrole>CounterParty</allocationpartyrole>                      |
| <childtradeexternalreference>Child_IRS_Trade_1</childtradeexternalreference> |
|  |
| <tradeallocation></tradeallocation>  |
| <allocationamount>2000</allocationamount>                                    |
| <allocationpercentage></allocationpercentage>                                |
| <allocationparty>CALYPSO2</allocationparty>                                  |
| <allocationpartyrole>CounterParty</allocationpartyrole>                      |
| <childtradeexternalreference>Child_IRS_Trade_2</childtradeexternalreference> |
|  |
|  |

The <TradeAllocations> structure should be used with <Action>ALLOCATE</Action> when you want the trade to be allocated in calypso, otherwise just the trade is uploaded with action <Action>ALLOCATE</Action>

Please Check the samples on how these approaches are used.

Once the Allocation is performed, the child trades are created and the External Reference of the child trades needs to be REKEYED, so that they can be used in the further life cycle actions.

Child trades can be rekeyed in 2 ways.

• At the time of Allocation

While Allocating the Trade in the <TradeAllocations> block pass the External Reference for each Child trade in the tag < ChildTradeExternalReference>, The child trade's external reference will be changed accordingly.

• After the Allocation

Using <CalypsoTrade> xml with <Action>ALLOCATION\_REKEY</Action>. The Parent Trades external Reference and the New External Reference has to be passed, then uploader will find out the child trades from the parent trade and find the appropriate child trade by matching the counterparty in case of counterparty allocation & book in case of book allocation. Please refer to samples for more examples.



<KeywordName>NewExternalRef</KeywordName> <KeywordValue>NEW\_XCCY\_CHILD\_AFTER\_ALLOCATION01</KeywordValue> </Keyword> <Keyword> <KeywordName>AllocatedFrom</KeywordName> <KeywordValue>ParentTradeExternalReference</KeywordValue> </Keyword> </TradeKeywords>

#### CSV Support

The following columns need to be added to support the Allocations in the existing CSV files.

## Allocation\_AllocationAmount,Allocation\_AllocationPercentage,Allocation\_AllocationParty,Allocation\_AllocationPartyR ole,Allocation\_ChildTradeExternalReference

1000,0.02,CALYPSO,CounterParty,Child\_IRS\_Trade\_2

Repeat these columns in the same order, for every allocation child to be created.

#### Workflows

Add the **UpdateAllocationChild** rule between VERIFIED and ALLOCATED, which is mandatory for setting the external references on the child trades.

Please Note that the rule **UpdateAllocationChild** must be present wherever ALLOCATE action is applied as follows.

| 🕌 WorkFlow Action  |                  |                     |                            |  |  |  |  |
|--|------------------|---------------------|----------------------------|--|--|--|--|
| Id 110   |                  | Action ALLOC        | ATE                        |  |  |  |  |
| Orig Status VERIFIED   |                  | Result Status ALLOC | ATED                       |  |  |  |  |
| Event Class PSEventTrade   |                  | Subtype ALL         |                            |  |  |  |  |
| Product ALL  |                  | Processing Org ALL  |                            |  |  |  |  |
| 🔲 Different User   | 🔽 Create Task    | Use STP             | Use KickOff/Cut Off        |  |  |  |  |
| Log Completed  | Preferred Action | 🔲 Update Only       | 🔲 Generate Intermediary Ev |  |  |  |  |
| 🔲 Needs man. Auth.   |                  |                     | 0 Priority                 |  |  |  |  |
| Rules       CheckFullAllocation, CheckValidAllocation, UpdateAllocationChild        Help |                  |                     |                            |  |  |  |  |
| Filter   |                  | Custom Rules De     | finition                   |  |  |  |  |



Make sure that you have configured your Calypso trade workflow to handle the allocation of a block trade along with the generic lifecycles for block and child trades which are described in the Calypso Allocation Documentation.

Calypso Recommends the following:

| Original Status   | Action   | Rules  | Resulting<br>Status | Comments  |
|-------------------|----------|--|---------------------|---|
| VERIFIED          | ALLOCATE | CheckFullAllocation<br>CheckValidAllocation        | ALLOCATED           | Primary transition for fully allocating<br>a trade.<br>Mandatory for<br>performing allocations.   |
| ALLOCATED         | AMEND    | CheckFullAllocation                                | ALLOCATED           | For making amendments to a block<br>trade.<br>The action is propagated to the<br>child trades through the action<br>specified in domain<br>PropagateBlockTradeChangesActio<br>n - See PROPAGATE action below.   |
| ALLOCATED         | ALLOCATE | CheckFullAllocation<br>CheckValidAllocation        | ALLOCATED           | Transition for updating the block<br>trade when child trades are directly<br>modified.<br>Optional if you do not allow direct<br>child trade modification -<br>See Modification of Child Trades for<br>details. |
| PARTIAL_ALLO<br>C | ALLOCATE | CheckFullAllocation<br>CheckValidAllocation        | ALLOCATED           | Transition for completing the allocation of a partially allocated trade.  |
| VERIFIED          | ALLOCATE | CheckPartialAllocatio<br>n<br>CheckValidAllocation | PARTIAL_ALLO<br>C   | Primary transition for partially allocating a trade.  |
| ALLOCATED         | ALLOCATE | CheckPartialAllocatio<br>n<br>CheckValidAllocation | PARTIAL_ALLO<br>C   | Transition for taking a fully allocated trade to partially allocated by changing the children.  |
| ALLOCATED         | AMEND    | CheckPartialAllocatio<br>n                         | PARTIAL_ALLO<br>C   | Transition for taking a fully allocated<br>trade to partially allocated by<br>modifying the block trade.<br>The action is propagated to the<br>child trades through the action                                  |



| Original Status   | Action        | Rules  | Resulting<br>Status | Comments  |
|-------------------|---------------|--|---------------------|---|
|                   |               |  |                     | specified in domain<br>PropagateBlockTradeChangesActio<br>n - See PROPAGATE action below.   |
| ALLOCATED         | UPDATE        | CheckPartialAllocatio<br>n<br>CheckValidAllocation | PARTIAL_ALLO<br>C   | Transition for taking a fully allocated<br>trade to partially allocated by<br>updating the block trade - This<br>action is applied on the block trade<br>when the allocation is modified. |
| PARTIAL_ALLO<br>C | ALLOCATE      | CheckPartialAllocatio<br>n<br>CheckValidAllocation | PARTIAL_ALLO<br>C   | Transition for partially completing the allocation of a partially allocated trade.  |
| VERIFIED          | PROPAGAT<br>E |  | VERIFIED            | Transition allowing block trade<br>amendments to be applied to<br>children.<br>This action name must be set in<br>domain<br>PropagateBlockTradeChangesActio<br>n.                         |
| VERIFIED          | AMEND         | NotAllocationChild                                 | VERIFIED            | Transition for rejecting direct<br>amendments to child trades<br>See Modification of Child Trades for<br>details.   |
| VERIFIED          | UPDATE        | NotAllocationChild                                 | VERIFIED            | Transition for rejecting direct<br>update of child trades<br>See Modification of Child Trades for<br>details.   |
| ALLOCATED         | TERMINATE     |  | TERMINATED          | Transition for terminating a block trade and all its children.  |
| ALLOCATED         | NOVATION      |  | TERMINATED          | Transition for novating a block trade and all its children.   |
| ALLOCATED         | CANCEL        |  | CANCELED            | Transition for canceling a block trade and all its children.  |



# 4 IBM MQ and JMS Support

The Data Uploader provides the ability to read messages from real-time messaging systems such as, IBM MQ or Active MQ, using the JMS API.

#### I NOTE: Calypso supports IBM MQ version 7 and above.

The Data Uploader uses the Upload Import Message for event handling and message processing. An event is created in Calypso for each received message, which is listened for and then processed by the Uploader Import Message engine, thus providing the ability to asynchronously process messages.

Information received from the queues is in the form of XML-formatted string messages. Calypso's Import engines support XML-messages that use the Data Uploader-specific CalypsoUploadDocument format.

The following setup is required to support the Data Uploader's usage of MQ and JMS.

#### Step 1 - Install IBM MQ Series.

In addition to installing MQ series, you need to obtain the following JARs from IBM:

- com.ibm.mq.commonservices\_7.0.1.3.jar
- com.ibm.mq.jar
- com.ibm.mqjms.jar
- com.ibm.mq.pcf.jar
- dhbcore.jar
- com.ibm.mq.headers.jar
- com.ibm.mq.jmqi.jar
- com.ibm.mq.jms.Nojndi.jar
- connector.jar
- fscontext.jar
- jms.jar
- jndi.jar
- provideerutil.jar

These files need to be added to the Calypso Installer to "Common Third Party Libraries & Extensions":



| Setup - Calypso 14.0.0.15.SP1   |  |
|---|--|
| Common Third Party Libraries & Extension<br>Please specify any third party libraries or extensions.   |  |
| This screen allows for the addition of common third party libraries such as JDBC drivers, vendor specific librari<br>extensions. Libraries and resources added here will be deployed to all Calypso components on the server an<br>You may also provide specific resources required by these libraries below. | ies for connectivity to feeds or custom<br>d client. |
| jconn4-5.5.jar<br>ojdbc6-11.2.0.1.0.jar   | Add<br>Remove  |
|   |  |

**Step 2** - Launch MQ Explorer and create the following objects:

- Queue Manager
- Input Queue
- Output Queue

Note that you will need each of these object names in a later step.

**Step 3** - Generate the Bindings file as explained below. The Bindings file is used as the PROVIDER\_URL to connect to the Middleware.

Modify the file "UploaderJMSAdmin.config".

Update the PROVIDER\_URL property as necessary for your implementation. This is the path to the bindings file:

PROVIDER\_URL=file:/usr/local/calypso/resources/uploader

Update the SECURITY\_AUTHENTICATION property as needed to the authentication model: None, simple, or CRAM\_MD5.

SECURITY\_AUTHENTICATION=none

**Step 4** – Modify "calypso home/client/bin/uploaderMQSeries.bat" on Windows platforms or "calypso home/client/bin/uploaderMQSeries.sh" on \*nix platforms.

Change the highlighted text to match the object names you have created in Step 2.

- QM\_Name = Your Queue Manager
- InputQueueName = Your Input Queue





#### OutputQueueName = Your Output Queue

```
echo + Creating script for object creation within JMSAdmin
echo del qcf(QueueConnectionFactory) > uploadermqsetup.scp
echo del q(OutputQueueName) >> uploadermqsetup.scp
echo def qcf(QueueConnectionFactory) qmgr(QM_Name) >> uploadermqsetup.scp
echo def q(OutputQueueName) qu(OutputQueueName) qmgr(QM_Name) tc(JMS)>> uploadermqsetup.scp
echo def q(InputQueueName) qu(InputQueueName) qmgr(QM_Name) tc(JMS)>> uploadermqsetup.scp
echo def q(InputQueueName) qu(InputQueueName) qmgr(QM_Name) tc(JMS)>> uploadermqsetup.scp
echo end >> uploadermqsetup.scp
```

**Step 5** - Run "calypso home/client/bin/uploaderMQSeries.bat" on Windows platforms or "calypso home/client/bin/uploaderMQSeries.sh" on \*nix platforms.

**Step 6** – Run the Uploader Import Message engine, the Sender engine, and the Update Manager engine if uploadMode = Local, and persistMessages=All or persisteMessages =Failure

#### 1 NOTE: Connection to the IBM MQ can be done by 2 ways:

- 1. Binding file approach Here connection established is of type CONNX.
- 2. Property file.
- In both the cases retry and delay functionality are supported.

The Uploader Import Message engine and the Sender engine can be started from the Engine Manager in Web Admin.

Please refer to Calypso Web Admin documentation for complete details.

#### Processing

The Uploader Import Message engine is now listening to messages from the MQ input queue.

The Sender engine is now listening to Message Events.

When a Message of a configured type reaches a particular state, the Sender engine processes the message and sends an Acknowledgement to the MQ output queue.

The engines use the file "calypso\_uploader\_config.properties" to connect to the MQ input and output queues.



While sending the messages to IBMMQ/ActiveMQ, user can provide the list the properties in the property file that need to be send as a message header tested for GatewayUploaderDocumentSender and GatewayJMSDocumentSender.

For example, if the message needs to be sent by JMS Document sender, then in the "calypso\_reporthub\_config.properties" file user can specify the headers as follows:

jms.header.TenantId=01422406

jms.header.SourceSystemName=Calypso

jms.header.AssetClass=Calypso\_IR\_Intraday

| 🛃 Message Sender   | Config                           |                                  |                          | +           |   |
|--------------------|----------------------------------|----------------------------------|--------------------------|-------------|---|
| Sender Config Cop  | y Config                         |                                  |                          |             |   |
|                    |                                  |                                  |                          |             |   |
| Message Status     | VERIFIED                         | ✓ Product Type                   | ALL                      | $\sim$      |   |
| Advice Type        | REPORTHUB_TRADE                  | <ul> <li>Address Type</li> </ul> | REPORTHUB                | ~           |   |
| Static Data Filter |                                  | Gateway                          | ReportHub                | ~           |   |
| Save Save          | Master and Copies AdviceDo       | ocuments will be saved in DB     |                          |             |   |
| Send Send          | Sender By Method                 | Sender By Gateway                | ] Sender By JMS JMS Name | reporthubnp | ~ |
| Gateway JMSDoc.    | umentSender class will be called |                                  |                          |             |   |

For GatewayUploaderDocumentSender, calypso\_uploader\_config file will be used with the header properties:



| 🔀 Message S   | Sende  | er Config                     |                 |               |              | — | × |
|---------------|--------|-------------------------------|-----------------|---------------|--------------|---|---|
| Sender Config | Cop    | y Config                      |                 |               |              |   |   |
|               |        |                               |                 |               |              |   |   |
| Message St    | tatus  | TO_BE_SENT                    | $\sim$          | Product Type  | ALL          |   | ~ |
|               |        |                               |                 |               |              |   |   |
| Advice        | Туре   | GATEWAYMSG                    | ~               | Address Type  | Uploader     |   | ~ |
|               |        |                               |                 |               |              |   |   |
| Static Data I | Filter |                               |                 | Gateway       | Uploader     |   | ~ |
| ☑ Save        |        | Master and Copies AdviceD     | ocuments will b | e saved in DB |              |   |   |
| 🗹 Send        |        | Sender By Method              | ✓ Sender        | r By Gateway  | Sender By JM | S |   |
| GatewayUp     | loade  | rDocumentSender class will be | called          |               |              |   |   |
|               |        |                               |                 |               |              |   |   |

In the MQ if we check the message we should see the headers like below Tenantld:

| ieneral         | Named Properties |                        |
|-----------------|------------------|------------------------|
| eport           |                  |                        |
| ontext          | Name             | Value                  |
| dentifiers      | AssetClass       | WallStreet_FX_Intraday |
| egmentation     | SourceSystemName | WallStreet             |
| amed Properties | TenantId         | 01422406               |

#### JMX Support

JMX is disabled by default using environment property DU\_MESSAGING\_SERVER\_USE\_JMX = false. When DU\_MESSAGING\_SERVER\_USE\_JMX = false, you need to set RISKCONTROL\_TELEMETRY\_DISABLE = true.

JMX management can be enabled by setting DU\_MESSAGING\_SERVER\_USE\_JMX = true.



When DU\_MESSAGING\_SERVER\_USE\_JMX = true, the Data Uploader tries to setup a remote JMX. The environment property DU\_MESSAGING\_SERVER\_USE\_REMOTE\_JMX = false (default) allows preventing the Data Uploader to setup a remote JMX.

If DU\_MESSAGING\_SERVER\_USE\_REMOTE\_JMX = true, environment properties

DU\_MESSAGING\_SERVER\_JMX\_HOST (default is localhost) and DU\_MESSAGING\_SERVER\_JMX\_PORT (default is 2001) control which host, and port are used for the remote JMX.



# 5 Source Message Workflow

The Data Uploader also provides dual- message capability similar to that found in the Calypso Exchange Feed module.

The Calypso Exchange Feed module implements a dual-message workflow. Markitwire however lacks this feature and can miss critical information from the data source if the item is not mapped (Book, PO, etc.). In this event, the source message is "lost" and must be replayed.

Dual message workflow allows Calypso to store Source Messages for Audit and Tracking purposes, as well as for replay via the Task Station rather than a scheduled task.

DS-Match (CCP) and ICELink also use the Dual Message workflow and Upload Source Message.

## NOTE: This change is mandatory for Calypso's Markitwire module beginning with V2.3.12+, V2.9.3+, V3.0.2+)

## 5.1 Implementation

When the Data Uploader receives a message from the source interface, it creates a Source BO Message and stores the Incoming Message, itself, as a payload so that messages received from the source are never lost.

Dual message support requires you to import the "UPLOADSOURCEMSG.wf" workflow. In addition, you must import an updated GATEWAYMSG workflow using "GATEWAYMSG.wf".

You can import these workflows as described for the GATEWAYMSG workflow earlier in the document.

#### UPLOADSOURCEMSG Workflow





The UPLOADSOURCEMSG workflow does the following:

- Translates the Source Message. If the Translation fails, the BO Message stops with a RECEIVED status. This blocked BO Message also blocks any further messages related to the same Source ID.
- If the Translation succeeds, a second BO Message is created. The second message is for Validation of the Translated message. It also creates the Trade in Calypso.





When Calypso acts with the CCP role, the following workflow rules link both party's messages and combine them. These additional rules are not required for TRADER Role.

BACKLOAD - BACKLOAD - PENDING\_VALID



| 🕌 WorkFlow Action (User: calypso   | _user)           |                      | <u>_   ×</u>             |  |  |
|--|------------------|----------------------|--------------------------|--|--|
| Id 37696   |                  | Action BACKLO        | AD                       |  |  |
| Orig Status BACKLOAD   |                  | Result Status PENDIN | G_VALID                  |  |  |
| Event Class PSEventMessage   |                  | Subtype              | EWAYMSG                  |  |  |
| Product ALL  |                  | Processing Org ALL   |                          |  |  |
| Different User   | Create Task      | Use STP              | Use KickOff/Cut Off      |  |  |
| Log Completed  | Preferred Action | Update Only          | Generate Intermediary Ev |  |  |
|  |                  |                      | Needs man. Auth.         |  |  |
| Rules CheckLink,DataBackLoad,DeClear, <mark>UpdateLinkedToAttribute Help Help</mark> |                  |                      |                          |  |  |
| Filter   |                  | Custom Rules Def     | finition                 |  |  |

#### • PENDING\_VALID - REJECT - REJECTED

| 🕌 WorkFlow Action (User: a | :alypso_user)    |                     |                          |
|----------------------------|------------------|---------------------|--------------------------|
| Id 39696                   |                  | Action REJEC        |                          |
| Orig Status PENDING_V      | ALID             | Result Status REJEC |                          |
| Event Class PSEventMes     | sage             | Subtype MWG4        | ATEWAYMSG                |
| Product ALL                |                  | Processing Org ALL  |                          |
| Different User             | Create Task      | Use STP             | Use KickOff/Cut Off      |
| Log Completed              | Preferred Action | Update Only         | Generate Intermediary Ev |
|                            |                  |                     | Needs man. Auth.         |
| Rules ApplyLinked          | MsgAction        |                     | Help                     |
| Filter RejectedMe:         | ssage            | Custom Rules D      | efinition                |

Static data filter " RejectedMessage":



| Static Data Filter Window [1200005P5/ft12/] (User: calypso_user) |          |  |        |          |  |  |
|--|----------|--|--------|----------|--|--|
| Name: RejectedMessage Attributes Simulate                        |          |  |        |          |  |  |
| Comment: Pending Modifs  |          |  |        |          |  |  |
| Groups: ANY  |          |  |        |          |  |  |
| Attribute  | Criteria |  | Filter | Value(s) |  |  |
| MSG_ATTRIBUTE.UploadObjectStatus  V LIKE  Rejected               |          |  |        |          |  |  |
|  |          |  |        |          |  |  |

#### • PENDING\_VALID - VALIDATE - PENDING\_TRADE

| 🕌 WorkFlow Action (User: cal | ypso_user)        |                     |                          |
|------------------------------|-------------------|---------------------|--------------------------|
| Id 37707                     |                   | Action VALID        | ATE                      |
| Orig Status PENDING_VALI     | D                 | Result Status PENDI | NG_TRADE                 |
| Event Class PSEventMessag    | je                | Subtype MWGA        | TEWAYMSG                 |
| Product ALL                  |                   | Processing Org ALL  |                          |
| Different User               | Create Task       | Use STP             | Use KickOff/Cut Off      |
| Log Completed                | Preferred Action  | Update Only         | Generate Intermediary Ev |
|                              |                   |                     | Needs man. Auth.         |
|                              | gAction, Validate |                     | Help                     |
| Filter ValidMessage          |                   | Custom Rules D      | efinition                |

Static data filter "ValidMessage":

| Static Data Filter Window [120000SP5/ft12/] (User: calypso_user) |            |            |              |   |  |  |
|--|------------|------------|--------------|---|--|--|
| Name: ValidMessage   | 4          | Attributes | Simulate     | e |  |  |
| Comment: Pending Modifs  |            |            |              |   |  |  |
| Groups: ANY  |            |            |              |   |  |  |
| Attribute  | Criteria   | Filt       | ter Value(s) |   |  |  |
| MSG_ATTRIBUTE.UploadObjectStatus                                 | - NOT_LIKE | ▶ Reje     | cted         |   |  |  |
|  |            |            |              |   |  |  |



#### Message Report - DSMATCH

| 3            | 3             |          |                   |                   |  |                 |                |               |            |                        |                         |
|--------------|---------------|----------|-------------------|-------------------|--|-----------------|----------------|---------------|------------|------------------------|-------------------------|
| emplate Desc | ription       |          |                   |                   |  |                 |                | I⊽ Internal   | Diterra    | r                      |                         |
| Start 07/    | 5105/00       | -        | · OreationDate    | Type              | UPLOADSOURCEMSG  |                 |                | Att           | ributes    |                        |                         |
| End          | •             | -        | -                 | Receiver          |  |                 | Filter Set     | _             |            | (                      |                         |
| Inde         | td 10         | -        | 6                 | Method            |  |                 | Product Family |               |            | 1                      |                         |
| Transfer     | ad ba         |          |                   | Contact Id        | <u> </u>   |                 | Product Type   | <u> </u>      |            | 122                    |                         |
| Satement     | Id            |          | Pn                | ocessing Org      | ALL  | -               | Ratus          | COMPLETED     |            | - Internet             |                         |
| Message      | 1d            | -        |                   | Message LE        |  | 1               | Grouping       |               |            |                        |                         |
| Templa       | te [          | 100      | - 1 - F           | balinked Id       | 6  |                 |                | -             |            |                        |                         |
|              |               | - 15     |                   | oy craed to       | and in the local division of the local divis |                 | ACCOUNT        |               |            |                        |                         |
| 50/0         | · 1           |          | a 🔟               |                   | 201  |                 |                |               |            |                        |                         |
| 3            |               |          |                   |                   |  |                 |                |               |            |                        |                         |
| SSAG         | Mig Linked Id | Trade Id | Mog_Attr.LinkedTo | ACTION            | Msg_Attr.UploadObje  | dExternalRe     | Mag_Attr.U     | OltheedObeedq | Mug Skahus | Msg_Attr.MessageSource | Msg_Attr.MessageFo      |
| 142004       | 0             | 0        | 10.000 million    | NEW               | MW_79943685_3  |                 | 142005         |               | COMPLETED  | MW                     | ClearingSWML            |
| 142082       | 0             | 0        | 8                 | NEW               | MW_79943686_3  |                 | 142083         |               | COMPLETED  | MW                     | ClearingSWIME.          |
| 142080       | 0             | 0        | C                 | NEW               | MW_79943605_4  |                 | 142001         |               | COMPLETED  | Hw                     | ClearingSWHL            |
| 142078       | 0             | 0        | 2                 | NEW               | MW_79943685_4  |                 | 142079         |               | COMPLETED  | MW                     | ClearingSWML            |
| 142076       | 0             | 0        | Cannon            | NEW               | MW 7994368G 3  |                 | 142077         |               | COMPLETED  | HW.                    | Clearner/WML            |
| 141595       | 0             | 71436    | 141593            | NEW               | MW OMNEANK 7994  | 607             | 71436          |               | COMPLETED  | MW.                    | ClearingSWML            |
| 141594       | ò             | 0        |                   | NEW               | MW 79H360F 3   |                 | 141595         |               | COMPLETED  | MW                     | Clearner/wm             |
| 141592       | 0             | 0        | -                 | NEW               | MW 7994360F 4  |                 | 141593         |               | COMPLETED  | MW                     | Cearcolivite            |
| 141591       | 0             | 71435    | 141589            | NEW               | MW DEALERCORP 79   | HAVE            | 71435          |               | COMPLETED. | 2011                   | Centrolive              |
| 141590       |               | 0        | 0                 | NEW               | MW 799434/JE 4   | a sume          | 141591         |               | OUMPLITED  | MW.                    | Generatives.            |
| 141500       |               | 71434    | 141501            | VALUE AND ADDRESS | Mail Completion 2004   | 100             | 71434          |               | COMPLETED  |                        | descerios de            |
| 141500       |               |          | 111100A           | NEW               | May 2004 Mar 1   |                 | 141589         |               | COMPLETED  | 100                    | Clear Contract Contract |
| 141597       |               | 71433    | TATES             | NEW .             | Mar MEMORY TODAY   | 100             | 71423          |               | COMPLETED. | Mar                    | (Tanger and a state     |
| 141004       |               | 71435    | 111.200           | ARTIN .           | BELL TODATION S  | 000             | 141537         |               | COMPLETED  | have                   | ClassingTidde           |
| 141000       |               |          | 141807            | ALL PROPERTY.     | BALL DEALERCOOD TO   | the fact starts | 21442          |               | COMPLETED  | Party -                | Clearly Make            |
| 141000       | 0             | /1432    | 141301            | THE ILIGHTE       | POW_DEPERSONORP_PV   | H3660           | 71432          |               | COMPLETED  | rue -                  | Charry Switz            |
| 141204       | 0             | 0        | 111404            | PAE.W             | PON_F9943600_4   | 14.0            | 141905         |               | COMPLETED  | PTW .                  | Cearrigwine,            |
| 141583       | 0             | 71431    | 141501            | NEW               | PTW_NEWYORK_79943  | 60C             | 71431          |               | COMPLETED  | PW .                   | CearnowML               |
| 1. A A 40.00 | 0             | 0        | 1.000             | NEW               | PWW_7994360C_3   |                 | 141583         |               | COMPLETED  | Privi                  | Cears (Switt            |
| 141582       |               |          |                   | The second of the |  |                 |                |               |            |                        |                         |

When you double-click an UPLOADSOURCEMSG message, the document viewer displays the source message and the translated UploadXML simultaneously.





The document viewer displays the source message in the left-hand panel and the translated UploadXML in the right-hand panel. By default, if a second level UPLOADSOURCEMSG exists, the right-hand panel is populated with its contents. If it does not exist, then the application performs a run-time translation. If an error occurs during run-time translation, then the application leaves the right-hand panel empty.

When you double-click a GATEWAYMSG message, the document view only displays the translated document, as shown below.





## 5.2 Benefits

Dual Message handling provides the following benefits:

- The system does not lose track of source messages when a mapping is incorrect.
- Messages are linked if they fail translation, blocking further life cycle actions.
- A separate Replay Scheduled Task (i.e., DoRecovery) is not required for messages experiencing a translation error. For example, if a particular Book is missing or not mapped, there is no longer a need to run a DoRecovery.
- The application can potentially eliminate some FORCE\_NEW/ FORCE\_AMEND DoRecovery, since translation will is accurate as per the Trade status in Calypso.



## 5.3 Task Station Configuration

#### **UPLOADSOURCEMSG**



#### **MWGATEWAYMSG**

| Tab Name                   | MWGATEWAYMSG   |    | List of Defined Tabs  |            |
|----------------------------|--|----|---|------------|
| Event Class<br>Event Types | Message  |    | Trade in Auth Queue<br>SDI Assign Auth Queue<br>Pay/Rec Msg for Man Release   |            |
| <br>All<br>Clear           | EX_MWGATEWAYMSG<br>EX_MWGATEWAYMSG_ERROR<br>EX_MWGATEWAYMSG_REJECT<br>EX_MWGATEWAYMSG_WARNING<br>PENDING_TRADE_MWGATEWAYMS | >> | Pay/Rec Messages to be sent<br>Edited Messages Auth Queue<br>Nacked Pay/Rec Messages<br>Nacked Confirm<br>Pending Trades<br>Held Pay/Rec Messages | Up<br>Down |
| Color                      | REJECTED_MWGATEWAYMS   |    | Messages Awaiting Response  |            |
| Set                        |  |    | Other Exceptions<br>Credit Event Monitor  |            |
| Remove                     |  |    | GATEWAYMSG  |            |



#### **DSMATCH**





# 6 Exporting Quotes

The EXPORT\_QUOTES scheduled task provides the ability to export stored quotes from one Calypso system into files which further can be used to import these quotes into another Calypso system. It allows:

- Ability to specify which quotes to include/exclude in the output
- Range of dates for export
- Multiple output formats (i.e. csv, xml)

#### Out of Scope

- Custom output templates/formatting
- Full dump of all quotes will not be allowed; number of lines will be capped
- Calypso report inside of Navigator (including CWS, Risk Config, new GUI, etc)
- No usage of Pricing Environments or trade population to select relevant quotes
- No leveraging of Quote Set dependency graph; given Quote Set is used as is

## 6.1 Configuration

#### 6.1.1 Domain Value

Check if EXPORT\_QUOTES is present in "scheduledTask" domain. This will be already present if GatewaySchemaData.xml has been executed using Execute SQL.

If the scheduled task entry is not present in the domain, then either add it manually or run the latest GatewaySchemaData.xml.

|          | Statement of the Statement Statement of the |
|----------|---|
|          |   |
| Name:    | scheduledTask                               |
| Value:   | EXPORT_QUOTES                               |
| Comment: | Scheduled Task for Exporting quotes         |

#### 6.1.2 Scheduled Task

**Step 1** – From the Calypso Navigator, navigate to **Configuration > Scheduled Tasks**.

**Step 2** – In "Task and Schedule Configuration" tab under task Definitions select **New Task**. Enter "EXPORT\_QUOTES" as the Task Type, enter the attributes, and **Save** the configuration.



| Task Attributes                |              |
|--------------------------------|--------------|
| NAME                           | Name         |
| QUOTE SET                      | default      |
| START DATE (yyyy-MM-dd)        | 2020-05-13   |
| END DATE (yyyy-MM-dd)          | 2020-05-15   |
| INCLUDE QUOTE NAMES LIKE       | *            |
| EXCLUDE QUOTE NAMES LIKE       |              |
| OUTPUT FILE TYPE               | CSV          |
| OUTPUT FILE PREFIX             | MARKET_QUOTE |
| OUTPUT FILE LOCATION           | C:\Quotes    |
| OUTPUT FILE NAME DATE FORMAT   |              |
| EXCLUDE TIMESTAMP IN FILE NAME | FALSE        |
| EXCLUDE MATURED PRODUCTS       | FALSE        |
| SEND REPORT BY EMAIL TO        |              |
| SEND REPORT BY EMAIL CC        |              |
| RECEIVER Legal Entity          |              |
| RECEIVER LE Contact Type       |              |
| SENDER LE Contact Type         |              |

#### Task Attributes

- NAME This mandatory input is a drop-down of all the available product codes (Name/ISIN/CUSIP/etc). By default "Name" is used.
- QUOTE SET This mandatory input is a string that must match exactly to a defined Quote Set in the system (BackOffice/default/OFFICE/etc). Only this specific Quote Set will be exported in the process, meaning no parent Quote Sets will be picked up and exported in the case of a defined hierarchy.
- START DATE and END DATE These fields are optional not entering a date implies only exporting for the Val Date. This input must be defined in a fixed format (i.e. yyyy-MM-dd, for example "2016-10-23"), it will not follow the environment or user locale. Start Date and End Date (Val Date) are inclusive in the date range.
- INCLUDE QUOTE NAMES LIKE This mandatory input will serve as criteria to search quotes with the provided name. Exact quote names can be entered, or a wildcard operator [\*] can be leveraged for more general selections. Multiple quotes name can be entered, separated by semicolons.
- EXCLUDE QUOTE NAMES LIKE The input provided will serve as criteria to exclude quotes with the provided name. Exact quote names can be entered, or a wildcard operator [\*] can be leveraged for more general selections. Multiple quotes name can be entered, separated by semicolons.
- OUTPUT FILE TYPE This mandatory input will specify the format in which quotes will be exported, either CSV or XML.
- OUTPUT FILE PREFIX This field is optional and the file exported will be prefixed with this input. By default, "QUOTES\_REPORT" will be prefixed when the input is not provided. The file generated will have this prefix appended with a timestamp and extension format (i.e. QUOTES\_REPORT\_20160803.xml).
- OUTPUT FILE LOCATION This mandatory input will provide the directory path used to write the exported quotes record. It must be a valid directory path with access permission to write.
- UTPUT FILE NAME DATE FORMAT Enter the date format for the file name as needed Example ddMMyyyyHHmm.


- EXCLUDE TIMESTAMP IN FILE NAME It is FALSE by default. When set to TRUE, the timestamp is not appended to the file name. If the scheduled task is run multiple times, the file is overwritten in that case.
- EXCLUDE MATURED PRODUCTS
- SEND REPORT BY EMAIL TO Enter the email addresses (comma separated) to send the report as "To" (see also RECEIVER Legal Entity / Receiver LE Contact Type below for more information).
- SEND REPORT BY EMAIL CC Enter the email addresses (comma separated) to send the report as "Cc".
- RECEIVER Legal Entity / Receiver LE Contact Type Select the legal entity and LE contact that should receive the email (email address specified on the LE contact). If not set, the email is sent to the email addresses specified in "SEND REPORT BY EMAIL TO".
- SENDER LE contact Type Select the LE contact (of the scheduled task's Processing Org) that should send the email (email address specified on the LE contact).

The email subject line and content can be populated using the following property files (the naming of the property files is based on Sender PO/Receiver Legal Entity):

EmailREPORT<report type>\_<Sender PO>-<Receiver Legal entity>.properties

Example:

EmailREPORTMARKET\_QUOTE\_CALYPSO\_SFO-AAA BANK.properties

**Step 3** – After saving an instance of the created scheduled task will be created as shown below that could be executed by selecting and clicking the **Run** button.

| Calypso Scheduled Tasks   |               |                     |              |                |     |             |  |  |  |  |
|---|---------------|---------------------|--------------|----------------|-----|-------------|--|--|--|--|
| Execution Report  | Task and Sche | edule Configuration | Progress Log |                |     |             |  |  |  |  |
| Task and Schedule Configuration   |               |                     |              |                |     |             |  |  |  |  |
| Scheduled Tasks can be defined and scheduled for execution below. The first panel contains scheduled task definitions and the second contains the schedule in which the tasks will be executed. |               |                     |              |                |     |             |  |  |  |  |
| Task Definitions  |               |                     |              |                |     |             |  |  |  |  |
| 省 New Task 🐉 New Chain 👔 Clone 🚦 Delete 🕨 Run 🐧 Refresh 🚯 Info  |               |                     |              |                |     |             |  |  |  |  |
| External Reference  | e ID          | Туре                | Trade Filter | Processing Org | SLA | Description |  |  |  |  |
| ext_quote_01  | 7001          | EXPORT_QUOTES       |              |                | 5   |             |  |  |  |  |

# 6.2 Limitation and Alerts

## 6.2.1 Quotes Records

To limit the size of the reports for performance optimization, constraints are placed around the exporting process.

If the quotes criteria fetch records which are greater than the defined limit, then an error message will alert the user to kindly reduce the search criteria to generate a more manageable report size. By default, 100,000 is the count used to limit the quotes fetch when the property is not specified in calypso user environment property file.



Clients must demonstrate an atypical and valid use case to qualify for the environment property which overrides the default limit. Please raise a Help Desk if this situation applies.

## 6.3 File Split

To limit the number of quotes in a single file for some given quote search criteria, the total number of quotes record fetched are segregated and written into multiple files.

For this purpose, the user needs to configure a domain "NumberOfQuotesPerFile" and must specify the maximum number of records needed to be written in a single file. In conditions when the quote records fetched are less than the domain value entry then the records will be written in a single file.

| Z Domain Values                                  |            |               |                   |                       |  |  |  |  |  |
|--|------------|---------------|-------------------|-----------------------|--|--|--|--|--|
| 🔗 Reload ⋥ Save 🧊 Save All 🛛 🕷 Constraints Setup |            |               |                   |                       |  |  |  |  |  |
| Q - NumberOfQuotesPerFile                        | 2 of 2 🙆 😡 | Valu <u>e</u> | Name:             | NumberOfQuotesPerFile |  |  |  |  |  |
| I NPVReversalFlows                               |            |               | Nonic.            | NumberorQuotesrernie  |  |  |  |  |  |
| 🗄 📲 Null.Pricer                                  |            |               | <u>V</u> alue:    | 20000                 |  |  |  |  |  |
| Inter Number Of Quotes Per File                  |            |               | Comment:          |                       |  |  |  |  |  |
|  |            |               | <u>_</u>          |                       |  |  |  |  |  |
| 🗄 🖷 🔠 obligationCharacteristics                  |            |               |                   |                       |  |  |  |  |  |
| 🗄 📲 obligationType                               |            |               | << <u>A</u> dd    |                       |  |  |  |  |  |
| 🗄 🗄 🛄 obligValMethodChoice                       |            |               |                   |                       |  |  |  |  |  |
| 🗄 🗄 🖽 ObservableDataType                         |            |               | >> <u>R</u> emove |                       |  |  |  |  |  |
| ObservableDomainType                             |            |               |                   |                       |  |  |  |  |  |

# 6.4 Typical Use Cases

## 6.4.1 Examples and Notes

- 1. Extract All Equity Quotes for the Current Day:
  - a. Include Quotes Like: Equity.\*;EquityIndex.\*
  - b. Exclude Quotes Like: empty
  - c. Start Date = NULL
- 2. Extract All Quotes for GOOG Since 2010
  - a. Include Quotes Like: Equity.GOOG
  - b. Exclude Quotes Like:
- 3. Commas are NOT valid separators. The only valid separator when defining the quotes to export is a semicolon (;).
  - a. Per common market convention, commas, spaces, slashes, etc. are valid in Quote Names. This would therefore break the export functionality in any of these cases.



# 7 Client Channel Definition Table (CCDT) - IBM MQ

This section will describe how to create and use a Client Channel Definition Table (CCDT) for a Queue Manager and a Queue Manager Group in WebSphere MQ.

A CCDT defines client connection channels and their attributes. Clients read this file to determine which queue managers they should connect to. The CCDT file can be either JSON or binary format. The queue manager does not read the CCDT file. It is only used to provide clients with channel definitions and authentication information.

From IBM® MQ 9.1.2, you can create a CCDT in JavaScript Object Notation (JSON) format, and this format has some advantages over binary CCDT.

A binary format CCDT is created automatically when a queue manager is created. The runmqsc command is used to update the client channel definitions stored in this table.

### **Binary Format**

- The queue manager creates a binary CCDT automatically in the @ipcc directory under the data directory.
- The binary CCDT associated with a queue manager is not only created automatically, but also kept in sync with the object definitions. When you create, modify, or delete a client channel object, both the queue manager object definition and the entry in the CCDT are updated at the same time.
- The CCDT is created in the location indicated by MQCHLLIB and with the filename indicated by MQCHLTAB, which is AMQCLCHL.TAB by default. You can change the path to the CCDT by setting MQCHLLIB. Keep in mind that if you have multiple queue managers on the same server, they will share the same CCDT location.

## JSON Format

- There is no default JSON CCDT. IBM MQ does not provide any tooling to create or edit CCDTs in JSON format. However, you have more configuration options when you manually develop a JSON CCDT than when you use the runmqsc command to work with a binary CCDT.
- A JSON format CCDT is a flat text file with a .json extension. You create and update this table manually.
- Some of the advantages of using JSON CCDT files are as follows:
  - You do not need IBM MQ for Multiplatforms to create and edit a JSON CCDT file.
  - Using the JSON format, you can define duplicate channel definitions of the same name. You can use this to make your IBM MQ deployment scalable and highly available when you deploy it on the cloud.
  - The JSON file is human readable, which make queue manager configuration easier.
  - A flat file format can be integrated with:

Version control tooling to track the CCDT history

Automation tooling in continuous delivery



- You don't need any special tool to maintain the CCDT file.
- The file size is smaller.
- This format provides backwards and forwards compatibility.

## Location of CCDT

- BM<sup>®</sup> MQ supports retrieving a CCDT from a file, FTP, or HTTP URL. You can make the CCDT accessible to the client as a shared file, while it remains located on the server. Alternatively, you can distribute the CCDT, by either copying the CCDT to individual client or copying the CCDT to a location shared by multiple clients.
- If you use FTP to copy the file, use the bin option to set binary mode; do not use the default ASCII mode. Whatever method you choose to make the CCDT available, the location must be secure to prevent unauthorized changes to the channels.
- Both authenticated and unauthenticated URLs are supported. Here are some examples:
  - URL=ftp://myuser:password@myhost.sample.com//var/mqm/qmgrs/QMGR/@ipcc/AMQCLCHL.TAB
  - URL=http://myuser:password@myhost.sample.com/var/mqm/qmgrs/QMGR/@ipcc/AMQCLCHL.TAB
  - URL=ftp://myhost.sample.com//var/mqm/qmgrs/QMGR/@ipcc/AMQCLCHL.TAB
  - URL=http://myhost.sample.com/var/mqm/qmgrs/QMGR/@ipcc/AMQCLCHL.TAB

## CCDT with IBM MQ classes for JMS

- To use a client channel definition table, the CCDTURL property of a ConnectionFactory object must be set to a URL object.
- The URL object encapsulates a uniform resource locator (URL) that identifies the name and location of the file containing the client channel definition table and specifies how the file can be accessed.
- An application can set the property by creating a URL object and calling the setCCDTURL() method of the ConnectionFactory object.
- For example, if the file ccdt1.tab contains a client channel definition table and is stored on the same system on which the application is running, the application can set the CCDTURL property in the following way:

```
java.net.URL chanTab1 = new URL("file:///home/admdata/ccdt1.tab");
factory.setCCDTURL(chanTab1);
```

Another example, suppose the file ccdt2.tab contains a client channel definition table and is stored on a system that is different from the one on which the application is running. If the file can be accessed using the FTP protocol, the application can set the CCDTURL property in the following way:

```
java.net.URL chanTab2 = new URL("ftp://ftp.server/admdata/ccdt2.tab");
factory.setCCDTURL(chanTab2);
```

1. It is necessary to define 2 channels with the SAME name: one server-connection channel (SVRCONN) and one client-connection channel (CLNTCONN).

2. The CCDT file is updated ONLY when the client-connection channel is created or changed.



- In addition to setting the CCDTURL property of the ConnectionFactory object, the QMANAGER property of the same object must be set to one of the following values:
  - The name of a queue manager
  - An asterisk (\*) followed by the name of a queue manager group
  - An application can set the property by calling the setQueueManager() method of the ConnectionFactory object.

factory.setQueueManager("QM");

- If an application then creates a Connection object from the ConnectionFactory object, IBM MQ classes for JMS
  accesses the CCDT identified by the CCDTURL property, uses the QMANAGER property to search the table for
  a suitable client connection channel definition, and then uses the channel definition to start an MQI channel to a
  queue manager.
- Note that the both CCDTURL and CHANNEL properties of a ConnectionFactory object cannot be set when the application calls the createConnection() method. If both properties are set, the method throws an exception. The CCDTURL or CHANNEL property is set if its value is anything other than null, an empty string, or a string containing all blank characters.

# 7.1 Queue Manager Groups

A Queue Manager Group is a set of connections in the client channel definition table (CCDT). The set is defined by its entries having the same value of the QMNAME attribute in their channel definitions. You can connect an application to a queue manager that is part of a queue manager group.

Some of the reasons for choosing to use a Queue Manager Group are as follows:

- You want to connect a client to any one of a set of queue managers that is running, to improve availability.
- You want to reconnect a client to the same queue manager it connected to successfully last time but connect to a different queue manager if the connection fails.
- You don't need to write any client code, to automatically reconnect a client connection to another queue manager if the connection fails.
- You don't need to write any client code, to automatically reconnect a client connection to a different instance of a multi-instance queue manager if a standby instance takes over.
- You want to balance your client connections across several queue managers, with more clients connecting to some queue managers than others.
- You want to distribute the reconnection of many client connections over multiple queue managers and over time, in case the high volume of connections causes a failure.
- You want to be able to move your queue managers without changing any client application code.
- You want to write client application programs that do not need to know queue manager names.

When to use a leading asterisk in the name of the queue manager group -

You can connect an application to a queue manager that is part of a queue manager group, by prefixing the queue manager name on an MQCONN or MQCONNX call with an asterisk.



When a queue manager group has a single entry and the names of the group and the entry are the same, then it is ok to only the name of the queue manager and not using the asterisk as a prefix.

However, when the group has multiple entries or the group name is different than the single entry, it is necessary to specify the group name with a leading asterisk.

Following are the scenarios for which we will give an example:

Scenario 1 - Simplest case, using same QMNAME for a single queue manager (QM3 in host1)

Scenario 2 - Queue Manager Group: QM1 (no leading asterisk)

Create following Queue Managers

host-1: QM1, QM2, QM3

host-2: QM1, QM2, QM3, QM4

Use following command to create Queue Manager:

crtmqm <QueueManagerName> strtmqm <QueueManagerName>

The following table is created in order to provide more specific information on the queue managers and channel names.

| Scenarios   | Queue Manager<br>Group<br>QMNAME | Queue Manager<br>Name | Hostname                  | Port         | Channel<br>Name |
|---|----------------------------------|-----------------------|---------------------------|--------------|-----------------|
| Scenario 1 - Simplest<br>case - using<br>QMNAME (QM3) for<br>queue manager<br>(QM3) in host-1 | QM3                              | QM3                   | localhost                 | 1451         | QM3             |
| Scenario 3 - Queue<br>Manager Group:<br>QM1 (no leading<br>asterisk)                          | QM1<br>QM1                       | QM1<br>QM1            | localhost<br>10.17.35.138 | 1431<br>1425 | QM1.A<br>QM1.B  |

#### 1 Note:

- Please do following steps for each Queue Manager
- Right click on the <QueueManager> "Properties" "Extended" "Connection Authentication" Remove "SYSTEM.DEFAULT.AUTHINFO.IDPWOS"
- Go to Communication property on left hand panel Channel Authentication Records Disabled
- Then run following commands to refresh:

```
runmqsc <QueueManagerName>
refresh security type(CONNAUTH)
```



## refresh security type (AUTHSERV)

```
refresh security type(SSL)
```

1 Ensure that all the above commands are executed whenever configuration on MQ Explorer is modified.

Please do the following steps after a Listener is created.

### • Specify MCA user as MUSR\_MQADMIN

• Expand Channel - Channel Authentication Records - Right Click Highlighted Block User List Type – Delete

Create following Queues:

DEFINE QLOCAL(Q9) REPLACE DEFINE QLOCAL(SOURCE) REPLACE DEFINE QLOCAL(TARGET) REPLACE

## Scenario 1 - Simplest case - using QMNAME (QM3) for queue manager (QM3) in host-1

runmqsc QM3

DEFINE CHANNEL(QM3) CHLTYPE(SVRCONN) TRPTYPE(TCP) REPLACE DEFINE CHANNEL(QM3) CHLTYPE(CLNTCONN) TRPTYPE(TCP) CONNAME('localhost(1451)') QMNAME(QM3) REPLACE DEFINE LISTENER (L3) TRPTYPE(TCP) PORT(1451) REPLACE Exit

CCDT

```
{
  "channel":
  [
    {
      "general":
      {
        "description": "cahnnel1"
      },
      "name": "QM3",
      "clientConnection":
      {
        "connection":
        Γ
           {
             "host": "127.0.0.1",
             "port": 1451
           }
        ],
        "queueManager": "QM3"
      },
```



"type": "clientConnection" } ]

## Scenario 2 - Queue Manager Group: QM1 (no leading asterisk)

```
host-1:
runmqsc QM1
DEFINE CHANNEL (QM1.A) CHLTYPE (SVRCONN) TRPTYPE (TCP) REPLACE
DEFINE CHANNEL (QM1.A) CHLTYPE (CLNTCONN) TRPTYPE (TCP) CONNAME ('localhost (1431)') QMNAME (QM1) REPLACE
DEFINE LISTENER (L1A) TRPTYPE(TCP) PORT(1431) REPLACE
exit
host-2:
runmqsc QM1
DEFINE CHANNEL(QM1.B) CHLTYPE(SVRCONN) TRPTYPE(TCP) REPLACE
DEFINE CHANNEL(QM1.B) CHLTYPE(CLNTCONN) TRPTYPE(TCP) CONNAME('localhost(1425)') QMNAME(QM1) REPLACE
DEFINE LISTENER (L1B) TRPTYPE(TCP) PORT(1425) REPLACE
exit
CCDT
{
  "channel":
  [
    {
      "general":
      {
        "description": "cahnnelQM1A"
      },
      "name": "QM1.A",
      "clientConnection":
      {
        "connection":
        [
          {
            "host": "127.0.0.1",
            "port": 1431
          }
        ],
        "queueManager": "QM1"
      },
      "type": "clientConnection"
    },
```



```
"general":
    {
      "description": "channelQM1B"
    },
    "name": "QM1.B",
    "clientConnection":
    {
      "connection":
      [
       {
          "host": "10.17.35.138",
          "port": 1425
       }
      ],
      "queueManager": "QM1"
    },
    "type": "clientConnection"
 }
]
```



# 8 Kafka Support

The Data Uploader provides the ability to receive and send data from kafka server using kafka adapter that can be plugged in by importmessage engine and sender engine.

The name of the adapter is "CalypsoKafkalEAdapter", which can be called from IEAdapterConfig.

This is similar to a queue in JMS. So, using the adapter our program can become a producer (send data to topic) or consumer (receive data from a specific topic). For connection and configuration we have added new properties that should be provided in 'calypso\_<config\_name>\_config.properties file.

- kafka.consumercount=2
- We can create multiple consumers that will work in a group (consumer group) to receive data.
- Also, each consumer will run in a separate thread.
- kafka.consumertopic=inputnew
- Name of the topic consumer will subscribe to
- kafka.producertopic=outputnew
- Name of the topic producer will subscribe to
- kafka.producerdatakey=calypsoimportmrssageenine

Apart from this the properties that are needed to be provided to Kafka configuration can be done in following ways:

For consumer the name of the property should start with "consumer."

e.g.

- kafkaconsumer.bootstrap.servers=localhost:9093
- kafkaconsumer.security.protocol=SSL
- kafkaconsumer.ssl.truststore.location=C:\\kafka\ssl\kafka.server.truststore.jks
- kafkaconsumer.ssl.truststore.password=calypso
- kafkaconsumer.ssl.keystore.location=C:\\kafka\sslclient\kafka.client.keystore.jks
- kafkaconsumer.ssl.keystore.password=calypso
- kafkaconsumer.ssl.key.password=calypso

For producer, the name of the property should start with "producer."

e.g.

- kafkaproducer.security.protocol=SSL
- kafkaproducer.ssl.truststore.location=C:\\kafka\ssl\kafka.server.truststore.jks
- kafkaproducer.ssl.truststore.password=calypso
- kafkaproducer.ssl.keystore.location=C:\\kafka\sslclient\kafka.client.keystore.jks
- kafkaproducer.ssl.keystore.password=calypso
- kafkaproducer.ssl.key.password=calypso



If the properties for producer and consumer are same, then instead of providing separately we can provide the common properties as follows:

- kafkaclient.security.protocol=SSL
- kafkaclient.bootstrap.servers=localhost:9093

We can connect to kafka with authentication, encryption, and authorization on. In the following examples we will see the properties needed for producer and consumer to connect to kafka server with different types of authentications and encryptions.

- 1. Plaintext connection without authentication and encryption
- 2. SSL encryption
- 3. SSL Authentication
- 4. SASL (Simple Authentication and Security Layer)
  - a. SSL
  - b. Plaintext

### SSL Encryption

For SSL we need to create self-signed certificates and keystore & truststore.

In the kafka server.properties file we need to specify the following properties to enable the encryption

- listeners=SSL://0.0.0.0:9093
- advertised.listeners=SSL://localhost:9093
- ssl.keystore.location=C:\\kafka\\ssl\\kafka.server.keystore.jks
- ssl.keystore.password=calypso
- ssl.key.password=calypso

#trustore information is also needed here so that different components of kafka can communicate with each other

- ssl.truststore.location=C:\\kafka\\ssl\\kafka.server.truststore.jks
- ssl.truststore.password=calypso
- inter.broker.listener.name=SSL

To clients (producers and consumers) we can provide these trustore and password which they can use or that can create a separate trustore and add the certificate in their trust store.

To start the producer and consumer from console for testing we need to add following properties In consumer.properties and producer.properties.

- security.protocol=SSL
- ssl.truststore.location=C:\\kafka\\ssl\\kafka.server.truststore.jks



• ssl.truststore.password=calypso

Command to start producer and consumer by providing the config file in command line is as follows:

- bin\windows\kafka-console-producer.bat --broker-list localhost:9093 --topic test --producer.config config/producer.properties
- bin\windows\kafka-console-consumer.bat --bootstrap-server localhost:9093 --topic test --consumer.config config/consumer.properties

#### SSL Authentication

Client can also have their certificates by which server can identify them and hence we will achieve authentication as the server will know the identity of the client.

In the encryption case:

- Only the brokers have signed servers' certificates.
- The client was verifying the broker certificates to establish an SSL connection.
- The client is "anonymous" to the broker (no identity).

In the authentication case:

- The clients AND the brokers have signed server certificates.
- The client and the brokers verify each other's certificates.
- The client now has an IDENTITY to the broker (We can apply ACLs).

After creating client side certificate and truststore, we can enable the authentication in the server.properties file by adding the following line:

#### ssl.client.auth=required

Now since the authentication is on, the client will have to pass following properties:

- security.protocol=SSL
- ssl.truststore.location=C:\\kafka\ssl\kafka.server.truststore.jks
- ssl.truststore.password=calypso

The above properties were already set before, which was used for encryption. Now we also need the following properties for authentication:

- ssl.keystore.location=C:\\kafka\sslclient\kafka.client.keystore.jks
- ssl.keystore.password=calypso
- ssl.key.password=calypso



#### SASL

For SASL we create users in kafka. The users are specified in the jaas config file.

Add the **zookeeper\_jaas.conf** file to the environment variable **KAFKA\_OPTS** before starting zookeeper. And start the Zookeeper.

In another window shell terminal add the **kafka\_server\_jaas.conf** file to the environment variable **KAFKA\_OPTS** before starting kafka server. Before starting the kafka server we need to update the property file so that various authentication and encryption are enabled.

The following server.properties file has the configuration for the following type of connections that will work with different ports.

- PLAINTEXT port: 9092
- SSL port: 9095
- SASL PLAINTEXT port: 9093
- SASL SSL port: 9094

Once the server is started, we can connect the producers and consumers to different ports with different configurations that are needed for that specific port.

- 1. We connect to port 9092 where there is no authentication and no encryption i.e. PLAINTEXT
  - a. start the producer and consumer no property file needed here
  - b. The producer and consumer are connected to port 9092 and the message is successfully delivered.
- 2. We connect to port 9095 where only SSL is enabled with authentication. Start the producer and consumer with following properties:

bin\windows\kafka-console-producer.bat --broker-list localhost:9095 --topic test --producer.config config/producer-SSL\_AUTH.properties

bin\windows\kafka-console-consumer.bat --bootstrap-server localhost:9095 --topic test -consumer.config config/consumer-SSL\_AUTH.properties

3. We connect to port 9093 where SASL PLAINTEXT is configured. Start the producer and consumer with following properties:

bin\windows\kafka-console-producer.bat --broker-list localhost:9093 --topic test --producer.config config/producer-SASL PLAINTEXT AUTH.properties

bin\windows\kafka-console-consumer.bat --bootstrap-server localhost:9093 --topic test -consumer.config config/consumer-SASL\_PLAINTEXT\_AUTH.properties

4. We connect to port 9094 where SASL-SSL is configured. Start the producer and consumer with following properties:

bin\windows\kafka-console-producer.bat --broker-list localhost:9094 --topic test --producer.config config/producer-SASL\_SSL.properties

bin\windows\kafka-console-consumer.bat --bootstrap-server localhost:9094 --topic test -consumer.config config/consumer-SASL\_SSL.properties



## Connecting to kafka server using the CalypsoKafkalEAdapter.

In calypso\_uploader\_config.properties file we have configuration for producer and consumer.

Consumer is connected to port 9094 with SASL\_SSL and Producer is connected to port 9095 with SSL 2-way authentication. We use this file in UploadImportMessageEngine and we are able to consume data from 'inputnew' topic and send the data to 'outputnew' topic.