TrueEx Integration Guide

Version 1.0.1

First edition — October 2018

Revision date	Comment
October 2018	First edition for version 1.0.1.

Contents

Section 1. Introduction

This document describes the Calypso TrueEx Interface setup. The TrueEx interface allows an End User Firm (Buy-Side) to import trades which have been booked through the TrueEx Terminal using TrueEx's workflow.

The Calypso TrueEx integration connects to TrueEx using a FIX interface. Once the connectivity is setup, a trader can book a trade through the TrueEx Terminal, and TrueEx will send that as a FIX message to the Calypso TrueEx interface. The message will then flow through the configured Calypso workflows which route the message through the appropriate stages to create a Calypso trade. Additional clearing lifecycle messages will also be sent over the FIX connection, and the appropriate lifecycle actions will be applied to the Calypso trade.

This document describes the configuration required to setup the workflows, etc. for the TrueEx interface to run successfully.

Note *The TrueEx interface is distributed as part of the TrueEx module, which may support other TrueEx interfaces which must be licensed separately.*

1.1 Supported Features

Product Types

The TrueEx interface supports the following TrueEx trade types:

- IRS (Vanilla)
- OIS
- Allocation- IRS

Fix Messages

The interface supports the following TrueEx messages (i.e. these message types can be consumed by Calypso):

- Execution Report (Trade): Once the trade is executed within the TrueEx terminal and the Dealer accepts, TrueEx will send a message over the FIX interface. A Bilateral trade will be created in Calypso to represent the trade between the two parties (Dealer vs. Buy-Side), with the Calypso user's party as the PO.
- Allocation (Trade): Allocation trade messages sent from TrueEx terminal are created as block trade and allocated child trade

For inbound trades (Trades booked via TrueEx terminal window), the following messages are supported:

- Trade Capture Report (**35=n**)
- ALLOCATION (**35=n**)

Note *The interface only supports processing of message 35=n into trade, messages other than this for example 35=8 are ignored.*

1.2 Not Supported

The TrueEx interface does ***NOT*** support the following workflows:

- Amend
- Clearing
- Cancel

Section 2. Installation and Setup

2.1 Software Requirements

2.1.1 Supported JRE Versions

Please use the appropriate JRE version depending on the supported version for the base Calypso release you are running.

2.1.2 Supported Calypso Versions

The module supports specific versions of Calypso. In addition, your implementation must have the current Hotfixes applied. Before downloading the TrueEx module, please refer to the table below to determine which module versions are applicable for your implementation. The cal-upload.jar is bundled with Data Uploader and no longer has a separate listing in the table. cal-upload.jar must still appear in the CLASSPATH.

Calypso TrueEx Module Compatibility

Calypso version	Module Name	Module version
14.4 to 15.X	datauploader	7.2.4
	calypso-trueEx	1.0.1 and up
16.X	datauploader	7.3.0 and up
	calypso-trueEx	1.0.1 and up

2.2 Installation Instructions

2.2.1 Setup Config Data using Execute SQL

Add the following files to Execute SQL from $CALYPSO_HOME/bin/dbscripts as needed:

- GatewaySchemaBase.xml
- FIXSchemaData.xml
- FpMLSchemaData.xml
- TrueExSchemaData.xml

2.2.2 Message Workflow Setup

The TrueEx module uses the UPLOADSOURCEMSG and GATEWAYMSG workflows when importing messages. These should have been setup as part of the Data Uploader Setup Guide.

Messages from the UPLOADSOURCEMSG workflow are translated from the external message format into Calypso's internal format and placed in the GATEWAYMSG workflow. The GATEWAYMSG workflow then translates the internal format, performs verifications, and saves the trade to the database.

2.2.3 Task Station Setup

The TrueEx module uses the Data Uploader Framework to create task station entries for all the messages and exceptions that are encountered. The user can view / reprocess the messages that are failed in validation from the task station.

Please see the Data Uploader Setup Guide for how to add the appropriate messages and exceptions to the Task Station.

Section 3. Book Mapping

Once the TrueEx interface determines the appropriate Legal Entity to use as the PO, it must then choose a Calypso Book for the trade. The section below outlines the logic used by the TrueEx interface.

If no Calypso Book is found using the rules below, an error will be raised.

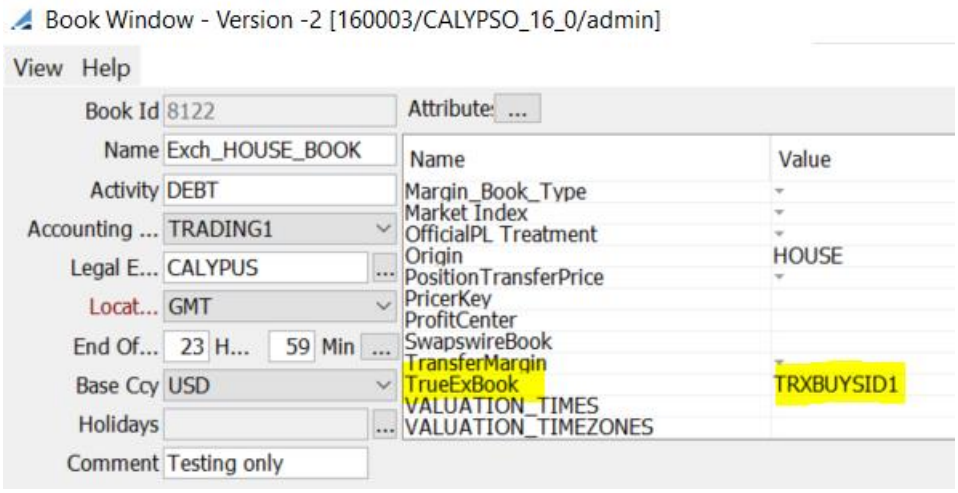
**Note* When a Calypso Book is determined based on the lookup rules described below, the interface will verify that the Book's Legal Entity matches the Calypso Legal Entity found using the rules specified in the previous section. This is especially important to note when using a parent LE; the Book must belong to the parent LE, not the child.*

When a trade is booked in the TrueEx Terminal, the user is NOT required to set any Book or Account value. However, in Calypso we require the trade to be assigned to a Calypso Book for the PO using the following mechanism.

3.1 Book Attribute

TrueEx account value coming in following message is mapped to book attribute **TrueExBook** in Calypso as shown below:

```
    </party>
    <party id="tradingPartner1">
        <partyId partyIdScheme="http://www.fpml.or
        </partyId>
        <partyId partyIdScheme="http://www.trueex.
    </party>
    <account id="ClearingBroker2Acct">
        <accountId>TRXBUYSID1</accountId>
    </account>
</nonpublicExecutionReport>
```



3.2 Book Name as Account Value

When no book is found using book attribute TrueExBook, the TrueEx account value coming in FIX message is searched in Calypso as **book name**.

3.3 Default Book Mapping

When no book is found using book attribute TrueExBook or book name, the TrueEx account value is mapped to PO legal entity attribute “TrueExBook” in Calypso.

Section 4. Legal Entity Mapping

The incoming TrueEx FIX messages contain Legal Entity identifiers for all parties involved in the trade (Party, Counterparty and CCP).

The Legal Entity for the Party, Counterparty and CCP identifiers within the FIX message will be identified in Calypso using the legal entity attribute TrueExParticipant, and the value would match the legal entity value provided in the FIX message.

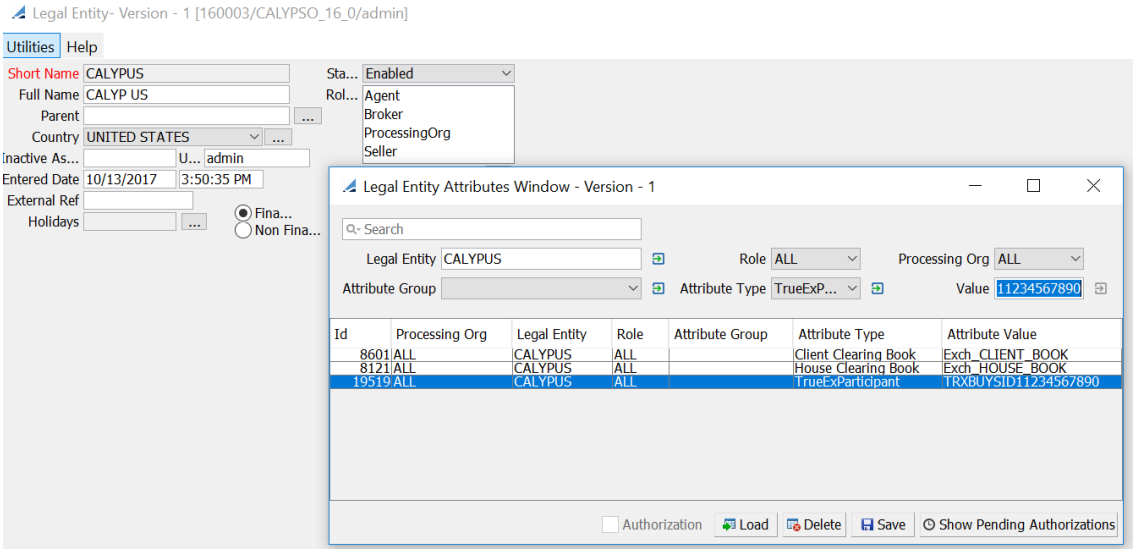
The fetching logic will first search for a legal entity with the attribute value matching the value specified in the message, and if not found then it will search for a legal entity having a matching Short Name (case-sensitive or all uppercase).

This lookup logic will be applied to PO, Counterparty and CCP lookups. If no Calypso Legal Entity is found using the rules above, an error will be raised.

4.1 PO Mapping

TrueEx partyId value coming in following message need to be mapped to a Legal-entity attribute **TrueExParticipant** in Calypso as shown below:

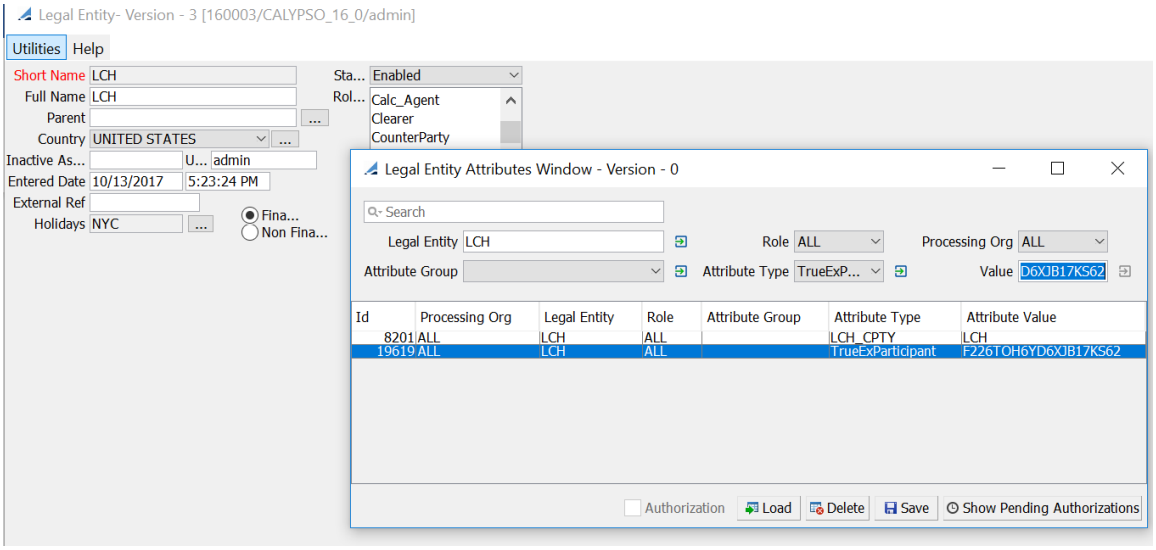
```
<party id="party2">
    <partyId partyIdScheme="http://www.fpml.org/coding-scheme/external/cftc/interim-compliant-identifier">TRXBUYSID11234567890
    </partyId>
    <partyName>Calypso Buy Side</partyName>
    <businessUnit>
        <name>IRS Desk</name>
        <country>840</country>
    </businessUnit>
    <person id="calypso_trader2">
        <personId>calypso_trader2</personId>
        <country>840</country>
    </person>
</party>
```



4.2 Counterparty Mapping

TrueEx partyId value coming in following message need to be mapped to a Legal-entity attribute **TrueExParticipant** in Calypso as shown below:

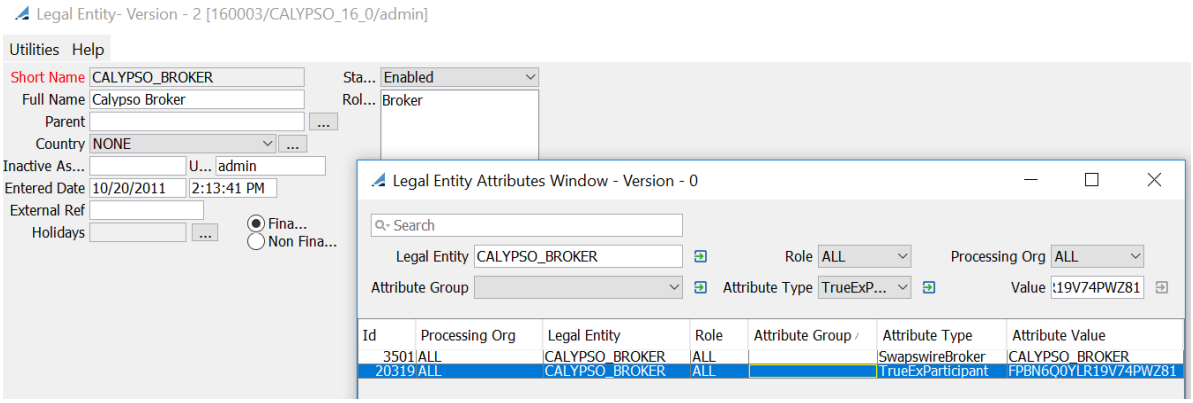
```
<party id="party1">
    <partyId partyIdScheme="http://www.fpml.org/coding-scheme/external/cftc/interim-compliant-identifier">F226TOH6YD6XJB17KS62
    </partyId>
    <partyName>LCH</partyName>
</party>
```



4.3 Clearing Broker Mapping

TrueEx partyId value coming in following message need to be mapped to a Legal-entity attribute **TrueExParticipant** in Calypso as shown below:

```
<party id="ClearingBroker2">
    <partyId partyIdScheme="http://www.fpml.org/coding-scheme/external/cftc/interim-compliant-identifier">FPBN6Q0YLR19V74PWZ81
    </partyId>
</party>
```



Section 5. Fix-Engine Configuration

The TrueEx FIX Engine is responsible for getting messages from the TrueEx platform and handing it off to the appropriate workflows.

The TrueEx FIX Engine is built on the Calypso FIX Engine framework, and therefore while setting it up you will find generic FIX setup vs. TrueEx specific setup. For clarity, all required steps are listed below.

Please review the standard Calypso documentation for Engine setup to read about several useful engine parameters (such as thread count) and how to set them.

5.1 Configure the Engine

All the database-based Engine configuration is completed as part of applying the schema, including the engine name, event subscription, event filter, event policy, as well as assigning a unique id to the Engine.

You may refer to the schema file for more details.

5.2 Setup the FIX Config File

To run the TrueEx FIX Engine out-of-the-box you will need a properties file with the name "**trueEx-fix.properties**" with the appropriate FIX connection settings.

A sample file is included under <calypso home>/resources (or <calypso home>/client/resources for Calypso 14.X) with the name "trueEx-fix.properties.sample". You will need to rename the file to "trueEx-fix.properties.properties".

In Calypso 14.X, once the file is customized, you will need to install it to the engine server war using the standard Calypso installation procedures for installing resource files.

Note that, as previously mentioned, the TrueEx FIX Engine uses the QuickFIXJ library for the FIX connectivity implementation. The QuickFIXJ library has many options that can be configured on a FIX session, using a standard properties file. The Calypso TrueEx FIX Engine uses this same file for internal settings as well.

For simplicity, we have provided a sample trueEx-fix.properties file and will only refer to the minimum settings that must be changed to work with TrueEx connectivity. You can view all the available settings on the QuickFIXJ Configuration page located at their documentation site at:
http://www.quickfixj.org/quickfixj/usermanual/1.5.3/usage/configuration.html

5.2.1 Sample Properties File

The sample "trueEx-fix.properties" file appears similar to the following example:

```
# Default settings for sessions.
# These are inherited by each session defined below
# unless they are overridden in the session settings.
[DEFAULT]
ConnectionType=initiator
ReconnectInterval=10
HeartBtInt=20
LogonTimeout=20
LogoutTimeout=20
Calypso.LogOnInterval=5000
Calypso.LogOnRetryCount=5
```

```
# SSL Support
SocketUseSSL=Y
SocketKeyStore=truex_dropcopy.jks
SocketKeyStorePassword=password123

Calypso.UploadMode=Local
Calypso.PersistMessages=All

# DUMMY_SEFCLIENT session definition (TrueEx SEF client)
[SESSION]
Calypso.FIXMessageType= TrueEx
BeginString=FIX.1.1
DefaultApplVerID=7
SenderCompID= CALB_IUC_1
TargetCompID= TRUEEX_IUC
SessionPassword=Calypsotrading1
DataDictionary=DD_TrueEx_SEF.XML
AppDataDictionary=DD_TrueEx_SEF.XML
SocketConnectHost=104.153.171.200
SocketConnectPort=39883
FileLogHeartbeats=Y
FileIncludeMilliseconds=Y
FileIncludeTimeStampForMessages=Y
ValidateIncomingMessage=N
RequiresOrigSendingTime=N

StartTime=07:00:00
EndTime=23:00:00
TimeZone=America/New_York
```

5.2.2 QuickFIXJ Settings

To connect with TrueEx successfully, you will need to change the SenderCompID, TargetCompID, SocketConnectHost, and SocketConnectPort connection properties to the correct values for your setup. Please contact TrueEx support for these details.

Additional points to note regarding the core QuickFIXJ settings:

- The QuickFIXJ settings allow you to configure multiple sessions in a single properties file. This means if you have multiple TrueEx session logins, you can use a single TrueEx FIX Engine to connect to all of them.
- FileStorePath and FileLogPath are defaulted to `$USER_HOME/Calypso/FIXEngine/Store` and `$USER_HOME/Calypso/FIXEngine/Log` respectively. These may be overridden at the DEFAULT or SESSION level within the config file. There is no support for other Store or Log mechanisms at this time.

5.3 Launching the TrueEx FIX Engine

5.3.1 Adding Logging Categories

To see logging messages for the Data Uploader and TrueEx modules you need to set the following log categories:

- UPLOADER: Set this to see logging for the Data Uploader translation from the internal Calypso xml format to the actual trade object.

- TrueEx: Set this to see logging for the TrueEx translation from the external format to the internal Calypso xml format.
- FIX: Set this to see logging for the shared FIX connectivity & message processing pieces of the TrueEx FIX Engine.

Additional debugging categories are listed in the Troubleshooting section of this document.

5.3.2 Running the TrueEx FIX Engine

With the previous steps completed, you are now ready to run the TrueEx FIX Engine.

- Calypso 14.X only:

  To start/stop the engine in Calypso 14.X, use the Calypso Engine Server Admin Web Console.

With the TrueEx Engine operating, you can then allege trades through the TrueEx Terminal. The TrueEx Engine will process the trade messages and create corresponding trades in Calypso.

The Task Station will display any errors that may occur.

5.3.3 Daily Stop/Restart

The TrueEx FIX server's shutdown daily after business hours and startup again at the start of business the next day. As part of this daily cycle, the Sequence Numbers for the FIX connections are reset as well.

The Calypso TrueEx FIX Engine handles this for you automatically, based on the values set in the trueEx-fix.properties settings file for the properties StartTime, EndTime, and TimeZone. These properties control when the engine determines that a new session should be started & the Sequence Numbers reset.

For more details on these settings, please refer to the QuickFIXJ documentation site.

Section 6. Trade Workflow

This section describes the TrueEx interface trade workflow. It is important to understand these details so that the Calypso Trade Workflow can be customized accordingly. Please read and follow all setup instructions carefully to ensure a successful installation.

6.1 Trade Workflow Setup

To support all the transitions required by the TrueEx workflow, the trade workflow setup for processing TrueEx trades must support the following transitions:

- NEW (for creation of new trades)
- Allocation (for allocating a block to a child trade)

6.2 Trade Capture Sequence

The table below describes the sequence of TrueEx FIX message flows in and how are they handled.

TrueEx Business	Calypso Action
Book a trade in TrueEx terminal and approve same	Calypso receives message: Fix message with MsgType as 35=n with clearing -status as Registered (Cleared)/Pending/Rejected **Action**: NEW trade is captured in calypso with keywords listed in section.7
Book allocation trade in TrueEx terminal	Calypso receives 2 messages: • **Block trade**: Fix message with MsgType as 35=n received. **Action**: NEW trade is captured in calypso with keywords listed in section.7 • **Allocated trade**: Fix message with MsgType as 35=n received with LCH-id. **Action**: Allocation action is applied on the trade captured in step.1 and a child trade is created in calypso with keywords listed in section.7

Section 7. Keywords

Following are the trade attributes/keywords supported.

Note *This is an early release version and keyword names are subject to change. Such changes will be documented in the Release Notes, so please ensure you review them whenever upgrading from previous TrueEx interface versions.*

Keyword Name	Description	Comments
TradeSource	Always set to 'TrueEx'	Used for engine logic behind the scenes; do not change.
USIPrefix	TrueEx Regulatory Reporting value	
USIValue	TrueEx Regulatory Reporting value	
CCP	LE short name of CCP	
CCPClearingBroker	The clearing broker (when available in the trade)	
Platform	Always set to 'TrueEx'	
PlatformStatus	TrueEx Trade Status	
PlatformTradeId	TrueEx Trade Id	As per TrueEx internal identifier
CCPStatus	Registered (Cleared), Pending, Rejected	
CCPClearedDate	Date when trade was cleared by CCP	
CCPClearedDatetime	Timestamp of last message to/from CCP	
CCPTradeID	CCP assigned deal id	
CCPAccount	Is this trade in a CLIENT or HOUSE account at the CCP	Always set to 'CLIENT'
CCPOriginCode	Did this trade originate due to CLIENT or HOUSE activity	Always set to 'CLIENT'
CCPConfirmationId	TrueEx CCP confirmation Id	

Keyword Name	Description	Comments
CCPSubmittedForClearingTimeStamp	TrueEx deal submitted time	
ExecutionTradeType		
ExecutionVenueType		
NegotiatedCurrency		

Section 8. Test Tool Setup: GUI

Note *The details in this section are provided for testing purposes only, and not recommended for production use.*

The Calypso TrueEx Interface is built on the Data Uploader framework, and therefore supports uploading TrueEx FIX files through the Data Uploader GUI. This can be useful for testing and does not require you to run the TrueEx FIX Engine.

The steps below assume you've already setup the Data Uploader module as per the Data Uploader Setup Guide, including adding the GUI window to your menu.

8.1 Setup the GUI Config File

Note *The need for this step will be removed in a future release.*

To upload TrueEx FIX files through the GUI, you will need a properties file with the appropriate settings. A sample file "TrueEx-datauploader-gui.properties" is included under $CALYPSO_HOME/resources (or $CALYPSO_HOME/client/resources for Calypso 14.X) with the name ".sample" as the suffix.

Note that the fixSettings property must point to the TrueEx FIX Engine's property file, although the engine itself does not need to be running.

You must also ensure that the 3rd party jars have been installed on the client side.

8.2 Uploading via the GUI

With the previous steps completed, you are now ready to upload TrueEx FIX files using the GUI. Simply launch the Data Uploader GUI from the menu, choose the Source/Format, browse to select your '.fix' file, and upload.

For further details on using the Data Uploader GUI, please refer to the Data Uploader Setup Guide.

Note *The uploaded file must have a '.fix' extension, not '.xml'*