

# Nasdaq Calypso Traiana Matching Integration Guide

Version 5.3.0

Revision 7.0 January 2024 Approved

- Internal WsepDistribution limited to Nasdaq personnel and authorized and Confidential arties subject to confidentiality obligations



#### 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	August 2019	First edition for version 2.0.1
2.0	November 2019	Second edition for version 2.0.2
3.0	March 2020	Third edition for version 3.0.1
4.0	November 2020	Fourth edition for version 2.0.5, 3.2.1
5.0	September 2021	Fifth edition for version 3.3.0
6.0	February 2022	Sixth edition for version 3.5.0, 4.1.0.
7.0	January 2024	Seventh edition for version 5.3.0 (compatibility with version 18)

This document describes the Calypso Traiana-matching Interface setup. The Traianamatching interface allows buy-side client (Calypso) to send out trades booked in Calypso to Traiana Harmony platform for equity CFD which can further be matched inside Traiana platform.

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.

Nasdaq - Internal Use: Distribution limited to Nasdaq personnel and authorized third parties subject to Page 2 / 31 confidentiality obligations



## Table of Contents

Introduct	tion	5
1.1	Calypso Traiana-matching Interface Flow	5
1.2	Calypso Traiana Matching Interface Features	7
	1.2.1       Supported         1.2.2       Not Supported	
Setup Ins	structions	8
2.1	Software Requirements	8
	2.1.1       Supported JRE Versions         2.1.2       Supported Calypso Versions	
2.2	Installation Instructions	8
	<ul> <li>2.2.1 Setup config data using Execute SQL</li> <li>2.2.2 Message Workflow Setup.</li> <li>2.2.3 Task Station Setup</li> </ul>	
Legal En	tity Mapping	
3.1	Broker Mapping	10
3.2	PO Mapping	12
3.3	Counterparty Mapping	12
Book Ma	apping	
Product	Security Code Mapping	
Fix-Engir	ne Configuration	17
6.1	Configure the Engine	17
6.2	Setup the FIX Properties File	17
	6.2.1       Sample Properties File         6.2.2       QuickFIXJ Settings	
6.3	Launching the Traiana FIX Engine	19
	<ul> <li>6.3.1 Adding Logging Categories</li> <li>6.3.2 Running the Traiana FIX Engine</li> <li>6.3.3 Daily Stop/Restart</li> </ul>	
Trade W	'orkflow	
7.1	Trade Workflow Setup	21
7.2	Trade Capture Sequence	22

Nasdaq - Internal Use: Distribution limited to Nasdaq personnel and authorized third parties subject to Page 3 / 31 confidentiality obligations



Trade Ke	eywords	
8.1	Supported Trade Keywords	24
8.2	Custom Trade Keyword Mapping	24
Custom	Tag Support based on Mapping	29
9.1	Fix Tags for Keywords	
9.2	Fix tags for Mapping Value	
Test Too	bl Setup: GUI	
10.1	Setup the GUI Config File	31
10.2	Uploading via the GUI	



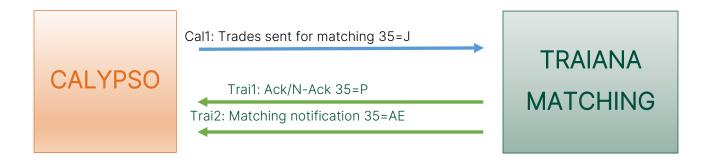
## Introduction

This document describes the Calypso Traiana-matching Interface setup. The Traiana-matching interface allows buy-side client (Calypso) to send out trades booked in Calypso to Traiana Harmony platform for equity CFD which can further be matched inside Traiana platform.

This document describes the configuration required to setup the Traiana-matching interface to run successfully.

() Note - The Traiana-matching interface is distributed as part of the Traiana-matching module, which may support other Traiana-matching interfaces which must be licensed separately.

## 1.1 Calypso Traiana-matching Interface Flow



Flow explanation:

- Cal1: Trades from Calypso (Buy-side client) sent for matching to Traiana
  - 1. Trades in Calypso are sent to Traiana platform individually for matching
  - 2. These trades are sent to Traiana using a fix message
  - 3. The fix message created is unique for every trade and its identified uniquely through Calypso trade-id which is generated in fix message when it is sent to Traiana platform.
  - 4. For this outgoing message the type of fix message generated is of type 35=J
  - 5. Following are the type of fix message generated from Calypso based on trade attribute PlatformSubmitStatus

Trade msg type	Fix representation	PlatformSubmitStatus
New	71 = 0 (AllocTransType)	When this keyword is not present on trade it is considered that this is new trade allege to Traiana
Replace	71 = 1 (AllocTransType)	When this keyword is present on trade it is considered as amend trade allege to Traiana

Nasdaq - Internal Use: Distribution limited to Nasdaq personnel and authorized third parties subject to Page 5 / 31 confidentiality obligations



Trade msg type	Fix representation	PlatformSubmitStatus
Cancel	71 = 2 (AllocTransType)	When this keyword is present on trade with value "Confirmed" then we are alleging trade for un-match to Traiana of type Cancel

- Trai1: Ack/N-Ack received from Traiana
  - 1. For the trades sent from Calypso to Traiana are first evaluated by the platform for correct statics and Traiana acknowledges them by sending back Ack or N-Ack
  - 2. Based on the type of acknowledgement received trade attributes are updated on trade with status and reason
  - 3. For this incoming message the type of fix message we receive is of type 35=P
  - 4. Acknowledgement type received by Calypso for the trades sent in CAL1

Ack type	Fix representation	Update on Calypso trade
Ack	87 = 0 (AllocStatus)	Trade attribute is updated with status as Pending
N-Ack	87 = 1 (AllocStatus)	Trade attributes are updated with status as Rejected and reject reason coming in fix field 58

- Trai2: Matching confirmation received from Traiana
  - 1. Post acknowledging the trade request sent from Calypso, Traiana performs matching on the trades which are acknowledged without any error.
  - 2. Traiana sends the result of the matching confirmation performed internally to Calypso via this matching notification.
  - 3. For this incoming message the type of fix message we receive is of type 35=AE
  - 4. Matching notification can be of following three types based on which trade keywords are updated with matching status

Matching msg type	Fix representation	Update on Calypso trade
Matched	573 = 0 (MatchStatus)	Trade attribute are updated with status as Confirmed
Unmatched	573 = 1 (MatchStatus)	Trade attribute are updated with status as Unmatched
Mismatched	573 = 2 (MatchStatus)	Trade attribute are updated with status as <b>Dispute</b>

Nasdaq - Internal Use: Distribution limited to Nasdaq nersonnel and authorized third parties subject to Page 6 / 31 confidentiality obligations



### 1.2 Calypso Traiana Matching Interface Features

#### 1.2.1 Supported

#### Product Type:

The Traiana-matching interface supports the following Calypso trade types:

• Portfolio Swap trade

#### Trade lifecycle:

- NEW
- AMEND
- CANCEL
- Note Trade messages which are sent out for matching are not part of allocation lifecycle executed in Calypso could be a scenario client would receive them as part of external allocation, but the interface does not consider the allocation parent-child relation.

#### Fix Messages:

Outbound message (Calypso-Traiana)

Calypso trades which are intended to be sent to Traiana is sent via fix message generated of type 35 = J

- Inbound message (Traiana-Calypso)
  - Ack/N-ack

For trades sent by Calypso to Traiana we receive acknowledgement from Traiana of type 35 = P

- Matching confirmation

For trades successfully acknowledged by Traiana further matching notifications are sent via fix message of type 35 = AE

Note - The interface only supports processing of message 35=P & 35=AE, messages other than this for example 35=8 are ignored.

#### 1.2.2 Not Supported

The following are not supported by the Traiana-matching for interface:

• Allocation performed in calypso and sending block and child trades in a single message for matching.



## Setup Instructions

### 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 Traiana-matching module, please refer to the release notes to determine which module versions are applicable for your implementation. The cal-upload.jar is bundled with the Data Uploader and no longer has a separate listing in the table. cal-upload.jar must still appear in the CLASSPATH.

### 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:

- GatewaySchemaBase.xml
- GatewaySchemaData.xml
- FIXSchemaData.xml
- TraianaSchemaData.xml

#### 2.2.2 Message Workflow Setup

The Traiana module uses the UPLOADSOURCEMSG, GATEWAYMSG and PLATFORMMSG workflows when processing 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.

PLATFORMMSG workflow is use for publishing the trades to external platform from Calypso.

#### 2.2.3 Task Station Setup

The Traiana-matching 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.



## Legal Entity Mapping

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

The Legal Entity for the Party and Counterparty and Executing Broker identifiers populated in the FIX message need to be configured in Calypso using the Legal Entity Attribute '**TraianaParticipant**', and the value must be the BIC code provided by Traiana.

However, clients can generate their own BIC code map it against the calypso legal-entity attribute as mentioned above and ask Traiana to configure same legal entities.

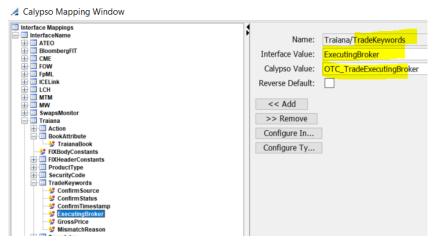
The logic will look for the attribute value TraianaParticipant configured on the Legal Entity involved in trade booking (Book-LE, CP-LE and Broker-LE) which will used to generate the fields for legal entity.

This lookup logic will be applied to PO (Book-LE), Counterparty and Executing Broker keyword lookups. If no Calypso Legal Entity is found using the rules above, an error will be raised.

## 3.1 Broker Mapping

For the Executing Broker keyword provided on the Calypso trade, the mechanism to handle and process the mapping is described below.

1. Identify the Executing Broker keyword name from the mapping whose value needs to be chosen from the keywords shown below:



2. Knowing the keyword name from the mapping, get the value of it from the Calypso trade to be alleged out to Traiana as shown below:



纉 Trade Attributes	×		
🕸 SetUp + 🙀 🛛 🖓	Editable		
Name	Value		
OTC_TradeExecutingBroker	CALYPSO_BROKER		
TradeSource	Traiana	^	
26T	<b>v</b>		
ACCOMMODATION_CHARGE_ID			
AccountNumber			
ADR Currency			
ADR Fee			
AFMAPricingCashRate			
AFMAPricingSwapRate			
AFMAPricingTM			
AfterSettlementCutoffTime			
Agent			
AllocatedFrom			
AllocationClearingType AmendmentEffectiveDate			
AmendmentEffectiveDate			
AmendmentTradeDatetime		~	
	OK	Cancel	

- 3. The value to be alleged will be decided using the following logic:
  - Find the Legal Entity in Calypso with a short name the same as the keyword value and with the Role
    of Broker mapped to it.

Legal Entity- Version - 2 [161018/CALYPSO_16_1/admin]									
Utilities He	elp								
Short Name	CALYPSO_BRO	KER			Sta	<b>Enabled</b>		$\sim$	
Full Name	Calypso Broker				Rol	Broker			
Parent									
Country	NONE		~						
Inactive As		U ac	lmin						
Entered Date	10/20/2011	2:13:4	41 PM				l		
External Ref									
Holidays				na on Fina.					
			0.	on rina.					

- When a boker LE is found, the value configured in the LE attribute **TraianaParticipant** is retrieved and used to allege in the message.

🦽 Le	gal Entity Attrib	utes Window					-		$\times$
	arch gal Entity CALYP ute Group	SO_BROKER	) ) ) )	Role Attribute Type			sing Org Value		~
Id 3501 20319 21221 27026	ALL	Legal Entity CALYPSO_BROKER CALYPSO_BROKER CALYPSO_BROKER CALYPSO_BROKER	Role ALL ALL ALL ALL	Attribute Group	Attribute Ty SwapswireSvPartic BloombergF TraianaParti	oker ipant TParticipant	CALYP	te Value 50_BROKEF 20YLR19V7	{ 4PWZ81
			Authori	zation 📮 Load	😼 Delete	Save O	Show Pe	nding Auth	orizations

Nasdaq - Internal Use: Distribution limited to Nasdaq personnel and authorized third parties subject to Page 11 / 31 confidentiality obligations



 If any configuration is missing, such as if an LE is not found, an LE with the specific role is not found or an LE does not have the attribute TraianaParticipant configured, the keyword value is sent as it is in the outgoing message to Traiana.

The value "EBXX" is populated in the fix message under fields:

448=EBXX 447= C 452=1

The field PartyRole -> 452 =1 signifies that EBXX should act or be mapped in Traiana to an Executing Firm.

### 3.2 PO Mapping

For the Calypso legal entity mapped on the Book which is intended to be used in trade booking must be mapped to attribute **TraianaParticipant** with the BIC code value configured in Traiana platform. See below illustration.

📕 Legal Entity- V		con16/calyp	oso_user]				
Utilities Help							
Short Name	PO				Enabled	<b>*</b>	
Full Name	Default Processing Orga	anisation			AccountHolder		
Parent					CounterParty ProcessingOrg		
Country	UNITED STATES		·				
Inactive As From	u	Jser		-			
Entered Date	02/13/2019	, 9:55:25 PM	_				
External Ref	,,	_					
Holidays	NYC		Financial				
	•		-				- 0
🔎 Legal Entity A	Attributes Window						
Legal Entity A Q - Search Legal Entity F			Đ	Role	ALL	<ul> <li>Processing O</li> </ul>	
Q- Search			Ð •	Role Attribute Type	1	▼ Processing O ▼ € Vali	
Q - Search Legal Entity		Lega			1		rg ALL
Q - Search Legal Entity F Attribute Group	PO Processing Org	PO	n Entity	Attribute Type	9005	→ Val	rg ALL 💌 ue 🖉 🖯
C - Search Legal Entity Attribute Group	PO Processing Org ALL		al Entity	Attribute Type	9005	Attribute Type	rg ALL 💌 ue 🖉 🖸
Q - Search Legal Entity F Attribute Group Id 13884	PO Processing Org ALL ALL	PO	▼ ∋	Attribute Type Role	9005	Attribute Type	rg ALL
Q ← Search Legal Entity F Attribute Group Id 13884 28982	PO Processing Org ALL ALL	PO PO	▼ ∋	Attribute Type Role ALL ALL	9005	Attribute Type DTCC_INCREASE SwapswireParticipant DEFAULT_BOOK DTCC_PAYREC_PARTICIPANT DTCC_PAYREC_PARTICIPANT	rg ALL  ve Attribute Value V CALVPIXCOX
Q - Search Legal Entity F Attribute Group Id 13884 28982 198267	PO Processing Org ALL ALL ALL ALL	PO PO PO	I Entity	Attribute Type Role ALL ALL ALL	9005	Attribute Type  DTCC_INCREASE  SwapswireParticipant  DEFAULT_BOOK	rg ALL  ve Attribute Value V CALVPXCOX Global
Q ← Search Legal Entity F Attribute Group Id 13884 28982 198267 13887	PO Processing Org ALL ALL ALL ALL ALL ALL	PO PO PO PO	Jentity	Attribute Type Role ALL ALL ALL ALL	9005	Attribute Type DTCC_INCREASE SwapswireParticipant DEFAULT_BOOK DTCC_PAYREC_PARTICIPANT DTCC_PAYREC_PARTICIPANT	rg ALL  v Attribute Value  V CALYPCOCX Global  Y
Q - Search Legal Entity F Attribute Group Id 13884 28962 198267 13887 13883	PO Processing Org ALL ALL ALL ALL ALL ALL ALL	PO PO PO PO PO	I Entity	Attribute Type Role ALL ALL ALL ALL ALL ALL	9005	Attribute Type     DTCC_INCREASE     SwapswireParticipant     DEFAULT_BOOK     DTCC_PAYREC_PARTICIPANT     DTCC_BOUGHTEACK	rg ALL  P Attribute Value Y CALYPXOOS Global Y Y
Image: Constraint of the second sec	PO ALL ALL ALL ALL ALL ALL ALL ALL ALL AL	PO PO PO PO PO PO	I Entity	Attribute Type Role ALL ALL ALL ALL ALL ALL ALL	9005	Attribute Type DTCC_INCREASE SwapswireParticipant DEFAULT_BOCK DTCC_PAYREC_PARTICIPANT DTCC_BOUGHTBACK DTCC_SSIGN	rg ALL  Y Attribute Value Y CALYPSYCOX Global Y Y Y Y Y Y
Image: Constraint of the second sec	PO Processing Org ALL ALL ALL ALL ALL ALL ALL AL	PO PO PO PO PO PO PO PO	al Entity	Attribute Type Role ALL ALL ALL ALL ALL ALL ALL ALL	9005	Attribute Type DTCC_INCREASE SwapowreParticipant DEFAULT_BOOK DTCC_PARTEC_PARTICIPANT DTCC_ASSIGN DTCC_ASSIGN CLSBestPractice	rg ALL  V Attribute Value V CALYPX00X Global V V Y V V V V S DTCC_PO 00001111
Id           1d           13884           28982           198267           13883           13883           13883           13883           13882	PO Processing Org ALL ALL ALL ALL ALL ALL ALL ALL ALL AL	PO           PO	al Entity	Attribute Type Role ALL ALL ALL ALL ALL ALL ALL ALL ALL	9005	Attribute Type DTCC_INCREASE SwapowieParticipant DFAULT POOK DTCC_PAIREC_PARTICIPANT DTCC_PAIREC_PARTICIPANT DTCC_SAUGHTBACK DTCC_SAUGHTBACK DTCC_PAIRECOM_JAIME DTCC_PREPORT	rg ALL  Y Attribute Value Y Attribute Value Y CALYPSOOX Global Global Y Y Y Y Y Y Y Y Y Y Y Y Y Y Y S DTCC_PO

The value "TRADER1" will be populated in fix message under fields:

448=TRADER1|447=C |452=3

The field PartyRole -> 452 = 3 signifies that TRADER1 should act or mapped in Traiana to a ClientID.

### 3.3 Counterparty Mapping

For the Calypso legal entity which is intended to be used as counterparty in trade booking must be mapped to attribute **TraianaParticipant** with the BIC code value configured in Traiana platform. See below illustration.



🗾 Legal En	itity- V	ersion - 6 [1610	29/vcon16/ca	lypso_user]						
Utilities H	lelp									
Short	: Name	GIGA_TEST1			s	tatus Enabled	×			
Full	Name	Giga Test Fund1			R	ole(s) Agent				
F	Parent					Broker CounterF	arty			
Co	ountry	NONE		¥		Issuer				
Inactive As	s From		User caly	oso_user		Processin	gOrg			
Entered	d Date	08/20/2019	8:12:08 Af	4						
Extern	nal Ref									
Ho	olidays			Financial O Non Finan	cial					
Legal	Entity	Attributes Wind	low							_ 🗆 >
Q,- Sea	irch									
Legal	l Entity	GIGA_TEST1		6	•	Role ALL		-	Processing Org ALL	*
Attribute	Group			· (	Attribu	ute Type 9005		• Ð	Value	Ð
	Id		ssing Org	Laule	- 6.96	Role	Attribute Grou		All to the Trans	Attribute Value
		5768 ALL	ssing Org	Legal E GIGA_TEST1	nacy	ALL	Attribute Grou	P	Attribute Type TraianaParticipant	JPMC
	20	D/ OU ALL		GIGW_ICOLI		MLL			i alariar al titiparit	DENK

The value "JPMC" will be populated in fix message under fields:

539=1|<mark>524=JPMC</mark>|525=C|538=4

The field PartyRole -> 538 =4 signifies that JPMC should act or mapped in Traiana to a Prime Broker.



## **Book Mapping**

In Calypso the book that is to be used for trade bookings for Traiana related trades must be mapped to the legal entity as mentioned in PO Mapping.

🔏 Calypso Mapping Window			
Interface Mappings InterfaceName ATEO BloombergFIT CME FoW FpML FOW FpML CLCLink CLCH MTM MW SwapsMonitor Traiana GAction BookAttribute FIXBodyConstants FIXBodyConstants		Nai Interface Val Calypso Val Reverse Defa << Add >> Remov Configure I Configure T	lue: OTC_ClientPortfolioID ult: e n
Book Window - Version -13 [161029/vcon16/caly	rpso_user]		
View Help			
Book Id 4060	Attributes		
	Accidates		
Name Global	Name		Value
Name Global	FloatingBondAccretion		Value
Name Global Activity All Business Lines	Name FloatingBondAccretion HKMA-Eligible		Value .
	FloatingBondAccretion		Value
Activity All Business Lines Accounting Link TRADING	FloatingBondAccretion HKMA-Eligble ICELInBook LCH_CPTY_BOOK LCH_CPTY_BOOK		Value 7
Activity Al Business Lines Accounting Link TRADING	FloatingBondAccretion HKMA-Elipible ICELinkBook LCH_CPTY_BOOK LooKOutBook MARKITWRE_PARTY_ID		Value 7
Activity All Business Lines Accounting Link TRADING	FloatingBondAccretion HKMA-Eligble ICELInBook LCH_CPTY_BOOK LCH_CPTY_BOOK		,
Activity Al Business Lines Accounting Link TRADING Legal Entity PO Location GMT	Name           FloatingBondAccretion           HKM4-Elipble           ICELin/Book           LCH_CPTY_BOOK           LookOutBook           MARKITWIRE_PARTY_ID           MARKITWIRE_PARTY_ID           Margin_Book_Type           Market: Index		,
Activity Al Business Lines Accounting Link TRAD/ING Legal Entity PO Location GMT End Of Day 23 Hour 59 Min	Name FloatingBondAccretion HKMA-EligIble ICELinkBook LCH_CPTY_BOOK LCOKOUtBook MARKITWIRE_PARTY_ID MARS-EligIble Margin_Book_Type Market Index ORIGIN	Y	, ,
Activity Al Business Lines Accounting Link TRADING Legal Entity PO Location GMT	Name FloatingBondAccretion HKMA-Elipble ICELinkBook LCH_CPTY_BOOK LookCutBook MAAKITWRE_PARTY_ID MASEIglible Margin_Book_Type Market Index ORIGIN ORGIN OFC_ClentRefortfoloID	Y	,
Activity Al Business Lines Accounting Link TRAD/ING Legal Entity PO Location GMT End Of Day 23 Hour 59 Min	Name FloatingBondAccretion HKMA-EligIble ICELinkBook LCH_CPTY_BOOK LCOKOUtBook MARKITWIRE_PARTY_ID MARS-EligIble Margin_Book_Type Market Index ORIGIN	Y	, ,
Activity Al Business Lines Accounting Link TRADING Legal Enkity PO Location GMT End Of Day 23 Hour 59 Min Base Ccy USD	Name FloatingBondAccretion HKMA-Eligble LCH_CPTY_BOOK LCH_CPTY_BOOK LCH_CPTY_BOOK MARKITWIRE_PARTY_ID MAS-Eligble Margin_Book_Type Market_Index CRIGIN OTC_ClientFortfolioID OTGabPL restment	Y	, ,
Activity Al Business Lines Accounting Link TRADING Legal Entity PO Location GMT End Of Day 23 Hour 59 Min Base Ccy USD Holidays LON,NYC,TARGET,TOK	Name FloatingBondAccretion HKMA-Elighte ICELInH8ook ICL_CT_CTY_BOOK ICCL_CTY_BOOK ICCL_CTY_BOOK MARKITWIRE_PARTY_ID MA3-Elighte Margin_Book_Type Market Index ORIGIN OTC_ClentPortfolioID OfficialPLifeetment PositTION_ACCOUNT_ID	Y	, ,

The Calypso value configured in the mapping window under Traiana >BookAttribute > TraianaBook is looked up on the trade book and populated under tag 79.

When the book Global is the intended book to be used on the trade with the LE mapped to PO, then the value of the book attribute (GIGA\_CCTEST1 in this case) found from the Calypso mapping (OTC\_ClientPortfolioID) is populated under tag 79, which corresponds to a fund level account.

78=1 79=GIGA\_CCTEST1



## Product Security Code Mapping

On the underlying product selected in trade when there are multiple security code/value pair configured then below preference is followed to choose one and generate its value in Fix tag 22 and tag 48:

- **Bloomberg Symbol** which is sent as 'A' in tag 22 and tag 48 = <security value of underlying for Bloomberg Symbol> in fix message
- ISIN which is sent as '4' in tag 22 and tag 48 = < security value of underlying for ISIN> in fix message
- SEDOL which is sent as '2' in tag 22 and tag 48 = < security value of underlying for SEDOL > in fix message
- RIC which is sent as '5' in tag 22 and tag 48 = < security value of underlying for RIC > in fix message
- CUSIP which is sent as '1' in tag 22 and tag 48 = < security value of underlying for CUSIP > in fix message
- CUSTOM, see below for details.

Above are default mappings, however you can customize them for any of following two requirements:

• Override <u>Bloomberg Symbol - A</u> mapping with some other security code, suppose from underlying product if we need to send some other code in place of Bloomberg Symbol then configure following mapping.

With this mapping if BB\_TICKER is found on your underlying its security value will be sent in fix message in tag 48 and tag 22 will be sent as A

Interface Mappings		
THE ATEO	Name:	Traiana/SecurityCode
BloombergFIT	Interface Value:	A
FOW FOM	Calypso Value:	BB_TICKER
	Reverse Default:	
	<< Add	
SwapsMonitor	>> Remove	
Action     BookAttribute	Configure In	1
FIXBodyConstants	Configure Ty	]
FiXHeaderConstants     FiXHeaderConstants     FiXedStatus     ProductType     ScarthyCode     JocurityCode     JocurityCode     JocurityCode     Translator     Translator     UBs     UBs		

 If there is a requirement to send in fix message an altogether separate security code and value apart from these defaults ones -> A (Bloomberg Symbol), 4 (ISIN), 1 (CUSIP), 2 (SEDOL), 5 (RIC),

Following mapping is needed and make sure the underlying has no other default security codes mentioned above otherwise they would be given more priority over this.

Note: Custom security code is used for understanding purpose can be some actual business code.





Name:	Traiana/SecurityCode
Interface Value:	P
Calypso Value:	Custom
Reverse Default:	
<< Add	
the second se	
Instanting of the second	
Configure Ty	
	Interface Value: Calypso Value:

In this case when we find Custom on underlying then it will be sent in fix messages as tag 22=P and tag 48 = <Security value of underlying for Custom>.



## **Fix-Engine Configuration**

Calypso FIX Engine is responsible for sending and receiving messages from the Traiana-matching platform.

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.

## 6.1 Configure the Engine

Following is screen snap which shows parameters need to be configured on Fix-engine for connecting to Traiana platform and events with filter need to be present for sending and receiving messages to Traiana

Engine Co	onfiguration	1 I					
Engine Name: (	0		Engine ID:	Max Queue Size: 🕜	Max Batch	Size: 🕜	Number of Threads: 🛛
FIXEngine			33				
Engine Class:				Event Pool Policy: 🛛		Pricing Environn	ient: 🕜
com.calypso.t	tk.engine.FIXEngi	ine		FIXEngine	۲		•
Display Name:	0	Appl	ication Type:	Save settle position c	hanges: የ		
FIX Engine		Eng	ineServer	•			
Description:				Configuration attribut			
					INV		*
				VALUATION_TIMES			
	t Configuration:			VALUATION_TIMEZON	IES		
PSEventAcco	ountBilling		_ 2	VERSION_CHECK			
PSEventFIXM	lessage			XFER_CHECK_FIRST			
PSEventPlatfe				XFER_NEVER_BV			
				XFER_NEXT_EVENT			
Event Filters:			*	XFER_PAST_GENERAT	ION		
	nownEventFilter	•	00	XFER_POS_AGGREGAT	ION_NAME		
FIXEngineEve				XFER_PRODUCT_FOR	EPAST		
			XFER USE AUTOMAT	C ACCOUNT			
			_	XFER_USE_MONEYDIF	-		
Epgine Manage	r Configuration:	C+	t on Startup:	XFER_USE_POS_AGGR			
engineserver		Jai	c on scarcup.	XFER_USE_REVERSE	Lourion_Oner		
		_				And the Comment	
				config		traiana-fix.pro	perties
				privingenv			
				tcid			T

## 6.2 Setup the FIX Properties File

To run the Traiana-matching FIX Engine out-of-the-box you will need a properties file with the name "**Traianafix.properties**" with the appropriate FIX connection settings.

A sample file is included under \$CALYPSO\_HOME/client/resources with the name "traiana-fix.properties.sample".

You will need to rename the file to "traiana-fix.properties.properties".

Once modified, copy the property file 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.

Note that, as previously mentioned, the Traiana 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 Traiana-matching FIX Engine uses this same file for internal settings as well.

For simplicity, we have provided a sample Traiana-matching-fix.properties file and will only refer to the minimum settings that must be changed to work with Traiana-matching 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

#### 6.2.1 Sample Properties File

The sample "traiana-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=Calypso-Adapter-Fix-FIX.1019122.jks SocketKeyStorePassword=273nbGSb EnabledProtocols=TLSv1.2 Calypso.UploadMode=Local

Calypso.PersistMessages=All

# DUMMY\_SEFCLIENT session definition (Traiana-matching SEF client)
[SESSION]
Calypso.FIXMessageType= Traiana
BeginString=FIX.1.1
DefaultApplVerID=7
SenderCompID= CALYPSO

Nasdaq - Internal Use: Distribution limited to Nasdaq personnel and authorized third parties subject to confidential to confidential to be a subject to confidential to confid



TargetCompID= TRAIANA DataDictionary=DD Traiana.xml AppDataDictionary=DD Traiana.xml SocketConnectHost=199.201.113.144 SocketConnectPort=443 FileLogHeartbeats=Y FileIncludeMilliseconds=Y FileIncludeTimeStampForMessages=Y ValidateIncomingMessage=N RequiresOrigSendingTime=N ResetOnLogon=Y ResetOnLogout=Y ResetOnDisconnect=Y StartTime=07:00:00 EndTime=23:00:00 TimeZone=Europe/London

### 6.2.2 QuickFIXJ Settings

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

For values in properties **SocketKeyStore and SocketKeyStorePassword** traiana-connectivity need to contact as they will provide the ". jks" which will contain the key-store certificates needed to create fix-session for the given Sender and Target Compld

Additional points to note regarding the core QuickFIXJ settings:

• 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.

### 6.3 Launching the Traiana FIX Engine

#### 6.3.1 Adding Logging Categories

To see logging messages for the Data Uploader and Traiana-matching 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.
- Traiana: Set this to see logging for the Traiana translation from the external format to the internal Calypso xml format.

Nasdaq - Internal Use: Distribution limited to Nasdaq personnel and authorized third parties subject to a subject to confidentiality obligations



 FIX: Set this to see logging for the shared FIX connectivity & message processing pieces of the Traianamatching FIX Engine.

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

#### 6.3.2 Running the Traiana FIX Engine

With the previous steps completed, you are now ready to run the Traiana-matching FIX Engine.

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

With the Traiana Engine operating, you can then allege trades from Calypso with workflow configuration showed in Section 6. The Traiana fix Engine will process the trade messages and will send them to Traiana platform for matching. Based on the response received from Traiana platform details will be updated on corresponding trades in Calypso.

The Task Station will display any errors that may occur.

#### 6.3.3 Daily Stop/Restart

The Traiana 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 Traiana-matching FIX Engine handles this for you automatically, based on the values set in the traiana-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.



## Trade Workflow

🚺 Nasdaq

This section describes the Traiana-matching 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.

### 7.1 Trade Workflow Setup

To support all the transitions required by the Traiana-matching workflow, the trade workflow setup for processing Traiana-matching trades must support the following **transitions**:

- NEW (for creation of new trades)
- Amend (for updating the incoming notifications on trade)
- Cancel (for cancelling trade)

#### Workflow rules:

- TraianaSource trade rule:
  - This rule gets populated in domain as part of running Traiana schema on executeSQL
  - This rule when applied on a transition say PENDING-VERIFY-VERIFIED updates the trade with keyword TradeSource=Traiana
  - The TradeSource keyword need to have the value Traiana because it helps to generate publish events which can be then be processed by fix-engine for sending the trade details to Traiana based as fix message.
  - When this workflow rule is not present on trade users have to manually add the keyword value on trade before alleging it to Traiana platform.
- PlatformAllege trade rule:
  - This rule gets populated in domain as part of running datauploader schema on executeSQL
  - This rule alleges calypso trade to target platform
  - The rule when applied on a transition say VERIFIED-PO\_ALLEGE-VERIFIED basically compares the keyword value TradeSource=Traiana against the values in domain UploadMessageSourceTypes. If the domain has the exact TradeSource value in our case which is Traiana the rule generates PSEventPlatformPublish events which are listened by fix-engine
  - Fix-engine on receiving the intended publish events processes them and send the fix message to target platform.

#### Note - Both the above rules can be applied on any transition based on how user have designed the trade WF to send messages to platform



## 7.2 Trade Capture Sequence

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

Calypso	Traiana Matching
Step1: Create a trade in Calypso having keyword value TradeSource=Traiana When this trade is alleged a FIX, message is sent to Traiana for matching	
	Step2: On receiving the fix message Traiana will send an ACK fix message that will indicate the message were processed by it without any errors or a NACK fix message to notify the Client that the message failed for some reason
Step3:Ack: For sent trades when calypso receives an Ack it updates the trade keyword with status as PendingN-Ack: If calypso receives N-Ack the trade keyword is updated with status as Rejected with reject reason. Once the trade is rejected then user cannot send it for further matching until the rejection reason is corrected in deal.	
	Step4: Post sending the Ack/N-Ack Traiana further perform matching on trades which were processed successfully and will send matching confirmation for these trades via fix message
Step5:Match: For successfully acknowledged trades when Calypso receives matching message then the trade keywords are updated with status as ConfirmedMismatch: For successfully acknowledged trades when Calypso receives mismatched message then the trade keywords are updated with status as Dispute, now user can choose here further to cancel this trade and book a completely new trade where user must start from Step1 or move to Step6.	



Calypso	Traiana Matching
Step6:	
<b>Amend trade:</b> In case of mismatch we can amend the trade in Calypso and re-allege same.	
When the trade is alleged a FIX, message is generated of type Replace or Cancel and sent to Traiana again for matching	
To support Cancel, you need the following mapping:	
Name = Traiana/Translator	
Interface Value = TradeStatus	
Calypso Value = CANCELED	
	On receiving this fix message flow from <b>Step2</b> to <b>Step 5</b> is followed.
Step7A:	
<b>Cancel trade:</b> When the trades are matched and we need to un-match them, then cancel the trade in Calypso and reallege same.	
On doing this a FIX, message is generated of type Cancel and sent to Traiana again for unmatching	
	Step7B:
	On receiving this fix message Traiana will sent Ack/N-ack like Step2
Step7C:	
On receiving the Ack/N-ack activities are performed as Step3	
	Step7D:
	Post sending the Ack/N-Ack Traiana further perform unmatching on trades which were processed successfully and will send confirmation for these trades via fix message
Step7E:	
<u>UnMatch</u> : For successfully acknowledged trades when Calypso receives unmatched message then the trade keywords are updated with status as <b>Unmatched</b>	



## Trade Keywords

Following are the trade attributes/keywords supported.

(1) 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 Traiana interface versions.

## 8.1 Supported Trade Keywords

Keyword Name	Description	Comments
TradeSource	Always set to 'Traiana'	Used for engine logic behind the scenes; do not change.
PlatformAllegeType	Platform message allege confirmation	
PlatformSubmitStatus	Platform allege status	
PlatformMatchId	Platform Id generated by Traiana platform and sent to calypso in incoming notifications	
ExecutionDateTime	Date time of last notification received from Traiana platform	
Custom-ConfirmSource	Like keyword TradeSource	Need to configure calypso mapping
Custom-ConfirmStatus	Like keyword PlatformSubmitStatus	to populate these keywords as described in section 7.2
Custom-ConfirmTimestamp	Like keyword ExecutionDateTime	

## 8.2 Custom Trade Keyword Mapping

In calypso mapping window when mappings are provided under type:

Traiana/TradeKeywords, they are used to populate following keyword-name on trades

• ConfirmSource



🥖 Calypso Mapping Window

Interface Mappings       Name: Traiana/TradeKeywords         Interface Value: Confirm Source         Calypso Value: OTC_Confirm Source         Reverse Default:         Interface Value: Confirm Source         Value: Va	InterfaceName     Name:     Traiana/Tradek       Image: Traiana/Tradek     Image: Traiana/Tradek	
Interface Mappings          Interface Value:       ConfirmStatus         Calypso Value:       OTC_ConfirmStatus         Calypso Value:       OTC_ConfirmStatus         ConfirmStatus       << Add         SwapsMonitor       << Configure In         ConfirmStatus       Configure In         ConfirmStatus       << Configure In         ConfirmStatus          ConfirmStatus       << Add         SwapsMonitor          Traisea          ConfirmStatus          ConfirmStatus          ConfirmStatus          ConfirmStatus          ConfirmStatus          ConfirmStatus          ConfirmStatus          ConfirmStatus          Configure In       Configure In         ConfirmStatus          ConfirmStatus          ConfirmStatus          ConfirmStatus          ConfirmStatus          ConfirmStatus          ConfirmStatus          Interface Value:          ConfirmStatus          Boombe	<ul> <li>CConfigure Ty</li> <li>CConfirmStatus</li> <li>ConfirmStatus</li> <li>ConfirmStatus</li> </ul>	ource
✓ Calypso Mapping Window          Interface Mappings         Interface Mame         Interface Mame         Interface Name         Interface Value:         Confirm Timestamp         Calypso Value:         OTC_Confirm Timestamp         Calypso Value:         OTC_Confirm Timestamp         Calypso Value:         OTC_Confirm Timestamp         Calypso Value:         OTC_Confirm Timestamp         Reverse Default:         Traiana         FIXHeaderConstants         SecurityCode         SecurityCode	Interface Mappings         InterfaceName	
Configure Ty     Configure Ty	✓ Calypso Mapping Window          Interface Mappings       Name: Traiana/Tradek         Interface Mappings       Interface Mame         Interface Mappings       Interface Mame         Interface Mappings       Name: Traiana/Tradek         Interface Mappings       Interface Name         Interface Mappings       Interface Name         Interface Mappings       Interface Name         Interface Name       Confirm Timest         Calypso Value:       OTC_Confirm Timest         Calypso Value:       OTC_Confirm Timest         Interface Value:       Interface Value:         Interface Value:       OTC_Confirm Timest         Interface Value:       Interface Value:         Interface Value:       Interface Value: <th>amp</th>	amp

GrossPrice

TrueEx

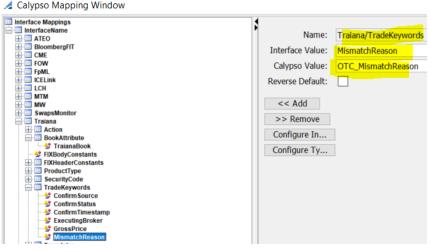
Nasdaq - Internal Use: Distribution limited to Nasdaq personnel and authorized third parties subject to confidential to confid



Calypso Mapping Window Interface Mappings InterfaceName In	Name:       Traiana/TradeKeyword         Interface Value:       GrossPrice         Calypso Value:       OTC_CFD_GrossPrice         Reverse Default:
Image: Security Code	

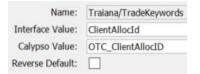
As per the above mapping value populated in keyword OTC\_CFT\_GrossPrice on trade will be sent out from Calypso to Traiana in tag **10220 – GrossPrice**.

• MismatchReason



In case of trade mis-match notification coming from Traiana, the mis-match reason comin in tag-142 will be populated in the Calypso value OTC\_MismatchReason, configured above as a trade keyword.

• Tag 70, 72 is populated from trade keyword and trade-id. To support this feature, the following mapping has been added.



If this mapping is not provided, Tag 70, 72 is populated from trade keyword OTC\_ClientAllocID by default. If trade keyword is empty, Tag 70, 72 is populated from trade-id.

• OTC\_ClientAllocID2 is sent for CANCEL messages in tag 72. To support this feature, the following mapping has been added.

Nasdaq - Internal Use: Distribution limited to Nasdaq nersonnel and authorized third parties subject to a subject to confidentiality obligations





Name:	Traiana/TradeKeyword
Interface Value:	ClientAllocId2
Calypso Value:	OTC_ClientAllocID2
Reverse Default:	

OTC\_TradeExecutingBroker

If this mapping is not provided, Tag 72 is populated from trade keyword OTC\_ClientAllocID2 by default.

TradeType

Name:     Traiana/TradeKeyv       Interface Value:     TradeType       Calypso Value:     OTC_CFD_TradeType       Reverse Default:		
纉 Trade Attributes		×
🌣 SetUp 🛛 😭 🕘 🔍		Editable
Name	Value	
OTC_CFD_GrossPrice	100	~
OTC_CFD_TradeType	SELLS	
OTC_ClientAllocID OTC_ConfirmSource	LO_ALLOC_900635 Traiana	
OTC ConfirmStatus	Confirmed	
OTC ConfirmTimestamp	14/10/2020 06:39:22	

As per the above mapping value populated in keyword OTC\_CFD\_TradeType on trade will be sent out from Calypso to Traiana in tag **10225** with interpretation to SHORT/LONG. When no mapping is provided, by default system will look in keyword OTC\_CFD\_TradeType.

Keyword value will be interpreted to SHORT and LONG by following logic:

EBXX

- When keyword value = BUYS or SELLS then tag 10255 = SHORT
- When keyword value = BUYL or SELLL then tag 10255 = LONG

#### Look-up logic:

Interface value (ConfirmSource, ConfirmStatus, ConfirmTimestamp, GrossPrice, MismatchReason) need to be constant and known to us so that we look for corresponding Calypso-value and populate on trade attribute, the interface value can be dynamic which would be picked over look-up performed for mapping.

If mapping is not in place, values will be populated in platform default keywords which are PlatformSubmitSatus, TradeSource, ExecutionDateTime.



#### Trade attribute window snapshot with custom keywords:

💋 Trade Attributes	×
🍄 SetUp 🕞 🙀 😧 🔍	🕅 Editable
Name	Value
OTC_CFD_GrossPrice	55.00
OTC_ConfirmSource	Traiana
OTC_ConfirmStatus	Confirmed
OTC_ConfirmTimestamp	20191107-05:32:30
OTC_TradeExecutingBroker	EBXX
PlatformAllegeType	Alleged
PlatformMatchId	1098179046
PlatformSubmitStatus	New Allege Confirmed
TradeSource	Traiana
UTIValue	1098179046

📈 Trade Attributes				
🔅 SetUp 🕞 🙀 🙆 🔍				
Name	Value			
OTC_CFD_GrossPrice	-1000.567655			
OTC_ConfirmSource	Traiana			
OTC_ConfirmStatus	Dispute			
OTC_ConfirmTimestamp	20191107-07:40:21			
OTC_MismatchReason	Message matching status changed from Unmatched to Mismatched : Settlement Date : Client Allocation Value - [11/12/2019], Allocation Confirmation Value - [11/07/2019]			
OTC_TradeExecutingBroker	EBXX			
PlatformAllegeType	Alleged			
PlatformMatchId	1098185477			
PlatformSubmitStatus	New Allege Dispute			
TradeSource	Traiana			
UTIValue	1098185477			

#### **Bock Level Matching:**

For 35=AE messages with 828=3 notification, the following trade keywords are stored:

• OTC\_BLOCKConfirmStatus – Based on the following mapping:

Name = Traiana/TradeKeywords

Interface Value = BLOCKConfirmStatus

Calypso Value = OTC\_BLOCKConfirmStatus

• OTC\_BLOCKConfirmTimestamp – Based on the following mapping:

Name = Traiana/TradeKeywords

Interface Value = BLOCKConfirmTimestamp

Calypso Value = OTC\_BLOCKConfirmTimestamp

- PlatformBLOCKMatchld Tag 880
- PlatformBLOCKRejectReason Tag 142



## Custom Tag Support based on Mapping

This feature can help a user to send fix tags in the fix message based on mapping configuration. User simply need to provide mappings as per below instructions and fix tag with its corresponding value will be populated in the outgoing fix message.

## 9.1 Fix Tags for Keywords

In this type we map a Fix tag to a Keyword-name in following way:

Mapping Type: CustomTagForKeywords

Interface Value: Fix tag

Calypso value: Trade Keyword name

🔏 Calypso Mapping Window	
Interface Mappings InterfaceName InterfaceN	Name:       Traiana/CustomTagForKeywords         Interface Value:       10250         Calypso Value:       OTC_CFD_OPEN_CLOSE         Reverse Default:

For trades sent to traiana if this keyword is present, its value will be populated in the outgoing fix message.

## 9.2 Fix tags for Mapping Value

In this type we map a Fix tag to a value in following way:

Mapping Type: CustomTagForFix

Interface Value: Fix tag

Calypso value: Valid value



🛃 Calypso Mapping Window

Interface Mappings	5	
E InterfaceName	Name: Traiana/CustomTagForFix	
BloombergFIT     GME	Interface Value: 10263	
FOW	Calypso Value: LIBOR	
	Reverse Default:	
MTM     MW     SwapsMonitor     Traiana     BookAttribute     CustomTagForFix	<< Add >> Remove Configure In Configure Ty	
Image: Security Code         Image: Security Code		

For trades sent to traiana if this mapping is present, its value will be populated in the outgoing fix message.

Note - The mapping types CustomTagForKeywords and CustomTagForFix will be populated as part of Execute SQL, however the calypso and interface value need to be configured by user

Fix tag and its value specified in mapping must be valid and should be identified by Traiana matching platform.

Trade keyword-name must be valid Calypso trade keyword-name.



## Test Tool Setup: GUI

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

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

### 10.1 Setup the GUI Config File

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

To upload Traiana FIX files through the GUI, you will need a properties file with the appropriate settings. A sample file "Traiana-matching-datauploader-gui.properties" is included under \$CALYPSO\_HOME/client/resources with the name ".sample" as the suffix.

Note that the fixSettings property must point to the Traiana 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.

### 10.2 Uploading via the GUI

With the previous steps completed, you are now ready to upload Traiana-matching 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.

#### 1 Note - The uploaded file must have a '. fix' extension, not '.xml'.