Oracle Financial Services Data Foundation Integration with Accounting Hub Cloud Service

User Guide

Release 8.1.2.0.0

April 2024

F40785-01

Oracle Financial Services Data Foundation Integration with Accounting Hub Cloud Service User Guide

Document Control

Version Number	Revision Date	Change Log
03	April 2024	Updated GL and SR balance Extraction section, removed Download Files for Insert Connectors section from the guide for Release 8.1.2.3.0.
02	September 2023	Updated Import Certificate into Keystore section (Doc 35471667).
01	April 2022	Created the document with instructions for the usage of the OFS Data Foundation Integration with Accounting Hub Cloud Service Release 8.1.2.0.0

Table of Contents

1 About the Guide

This section provides a brief description of the scope, the audience, the references, the organization of the user guide, and conventions incorporated in this user guide.

1.1 Scope of the Guide

This user guide explains the features and functions of FSDF Integration with AHCS. Setup and configuration of the system, pre-packaged business or functional content, technical integration aspects, and guidelines on the usage of the system are explained in detail.

Note that additional setup and configuration needs to be performed in AHCS to support accounting rules, journals, and general ledger. See the Oracle ERP Cloud documentation (**Learn More** > **Documentation Library**) on AHCS for more details.

1.2 Intended Audience

This manual is intended for the following audience:

- Technologists: Technical specialists who deal with setup and configuration of the interface between Data Foundation and Accounting Hub Cloud Service, and those responsible for the care and maintenance of the OFSAA instance.
- Business Users: Functional specialists who deal with actuarial sciences, risk management, accounting, or a combination of these, seeking to understand details of the interface between Data Foundation and Accounting Hub Cloud Service.

1.3 Prerequisites

- See the DIH Installation Guide for details on OFSAA components, Oracle applications, and environment that constitute technical prerequisites to deploy and use FSDF Integration with AHCS.
- User credentials set up by an authorized user is required before you can access the solution through its user interface.

1.4 Related Information Sources

Along with this user guide, you can also refer to the following documents in OHC Documentation Library:

- Oracle Financial Services Data Integration User Guide Release
- Oracle Financial Services Data Integration Application Pack Installation Guide Release
- Oracle Financials Cloud - Implementing Accounting Hub Guide
- Oracle Financials Cloud - Accounting Hub Best Practices Guide

1.5 Abbreviations

The following table defines the abbreviations used in this guide.

Table 1: Abbreviations

Acronym	Description
ADI	Application Data Interface
AHCS	Accounting Hub Cloud Service
Apps	Application
DIH	Data Integration Hub
EDD	External Data Descriptor
GL	General Ledger
KM	Knowledge Module
PT	Pass-Through
SR	Supporting References
SLA	Subledger Application / Subledger Application
UI	User Interface
UCM	Universal Content Manager

1.6 Reference Documents

The following are the reference documents available in this guide.

- Subledgers and Configurations
- SLA Template Validations
- Extract Connectors Structure
- Insert Connectors Structure
- SLA Coverage
- Pre-packaged Subledger Configurations
- List of Seeded Product Codes

2 Introduction

2.1 About Financial Service Analytical Applications

Oracle Financial Services Analytical Applications (OFSAA) enables financial institutions to:

- Measure and meet risk-adjusted performance objectives
- Cultivate a risk management culture through transparency
- Lower the costs of compliance and regulation
- Improve insight into customer behavior

OFSAA uses industry-leading analytical methods, shared data models, and application architecture to enable integrated risk management, performance management, customer insight, and compliance management. OFSAA actively incorporates risk into decision-making, enables to achieve a consistent view of performance, promote a transparent risk management culture, and provide pervasive intelligence.

OFSAA delivers a comprehensive, integrated suite of financial services analytical applications for both banking and insurance domains. It includes many applications such as Profitability Management, Asset Liability Management, Customer Insight, and Risk Management.

See Oracle Financial Services Analytical Applications portal for additional details.

2.2 About Oracle Fusion Accounting Hub Cloud Service

Fusion Accounting Hub Cloud, part of Oracle Enterprise Resource Planning Financials Cloud suite, is a robust accounting engine that integrates and aligns information from virtually any source system to consistently enforce accounting policies and meet multiple reporting requirements, giving institutions the agility, control, and insight needed to run their business. Oracle Fusion Cloud Financials empowers modern, adaptable finance with increased productivity and improved business decisions. Oracle Financials is a comprehensive solution delivering extensive support for global companies in a variety of industries, and continuous innovation in key technologies such as machine learning, intelligent automation and analytics, deployed in the cloud to achieve more, faster and with fewer resources.

See Oracle Financials Cloud portal for additional details.

2.3 Objectives

Integration of AHC and OFSAA provides a basis for a unified finance and risk architecture to the financial services industry. The benefits of having a ready-to-use interface between AHC and OFSAA are as follows:

- The OFSAA repository of transaction and reference information becomes a single, unified analytical repository for risk and finance.
- The integration improves consistency in risk and finance data for producing risk-adjusted measures.
- OFSAA customers can take advantage of AHC rules and accounting engines to improve the efficiency and audit ability to transform transactions from financial services systems such as deposits, loan accruals, payments, trades, and withdrawals into accounting.
- Information in the OFSAA foundation such as customer, account, product, branch, and channel information is accessible in AHC, for defining accounting treatments in addition to generating

accounting balances. Conditional logic can be used to vary accounting treatments based upon values from transactions and contract balances. The same OFSAA transactions and contract balances are used by AHC to book entries and generate financial balances, which in turn are reused by OFSAA for analytical processing.

- Using AHC balances for analytical processing facilitates the reconciliation of operational risk losses and accounting as prescribed by regulations such as Basel II and III. These accounting balances can be reconciled using the OFSAA Reconciliation Framework. Reconciliation rules can be defined to map AHC balances to product processor information. For example, accounting for provisions and losses can be based upon customer account balances, as provided to OFSAA and read by AHC enhancing the reliability of operational risk and provisioning measurement.

2.4 About Financial Services Data Foundation Integration with Accounting Hub Cloud Service

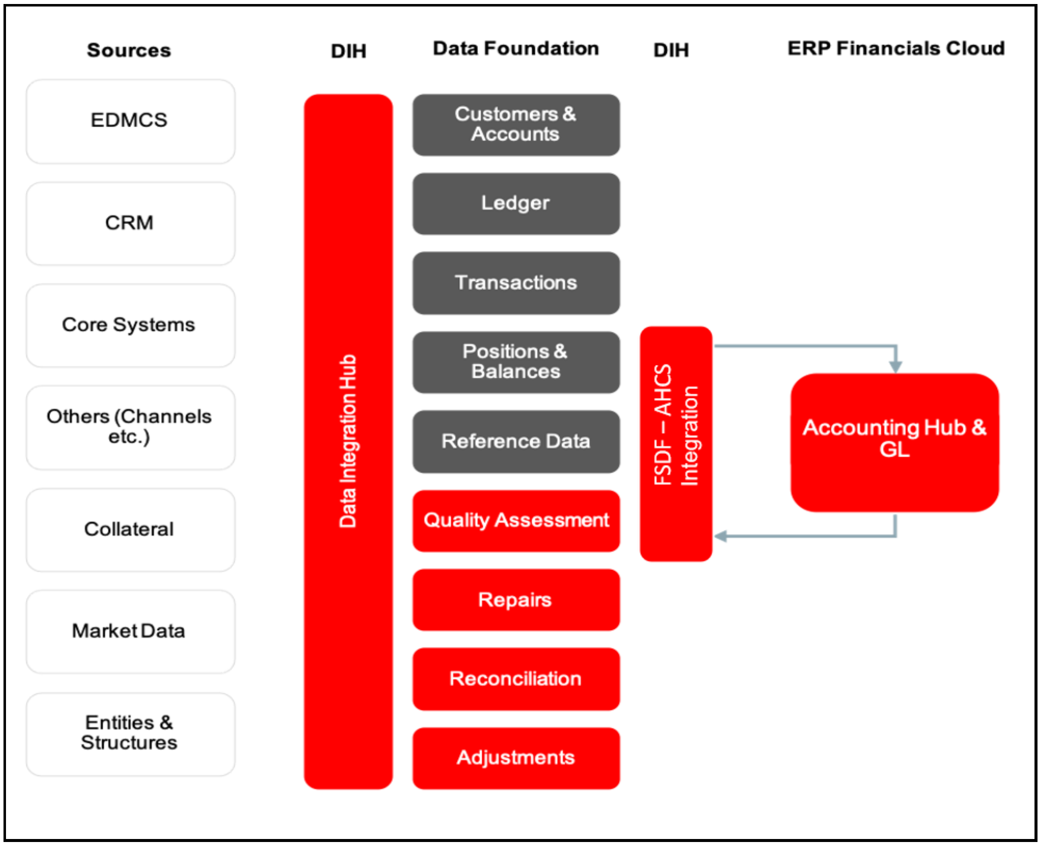
The following are the core constituents or functions of FSDF Integration with AHCS:

- Preconfigured Subledger Applications:
  - Retail Banking
  - Commercial Banking
  - Investment Banking
  - Islamic Banking
  - Treasury
  - Core Banking (using pass-through)
- Automated population of the source system and Subledger Application registration with AHCS.
- The configurable mechanism for grouping events data.
- The prebuilt mechanism for publishing events information to Accounting Hub Cloud Service.
- User interface allowing configuration of Subledger Applications and associated events information.
- The prebuilt mechanism to ingest balance information against the AHCS chart of accounts (GL Balance) and with Supporting Reference information (SR Balance).

2.5 OFSAA-AHC Interface Architecture Overview

The following illustration displays the OFSAA-AHC Interface Architecture.

Figure 1: OFSAA-AHC Interface Architecture



The data flow between OFSAA and ERP Financials Accounting Hub Cloud and GL (abbreviated AHCS) is bi-directional. OFSAA includes all the accounts, contracts, and transaction information that AHC generates for its Subledger accounting. Additionally, the various products under the OFSAA suite require General Ledger and supporting reference balances.

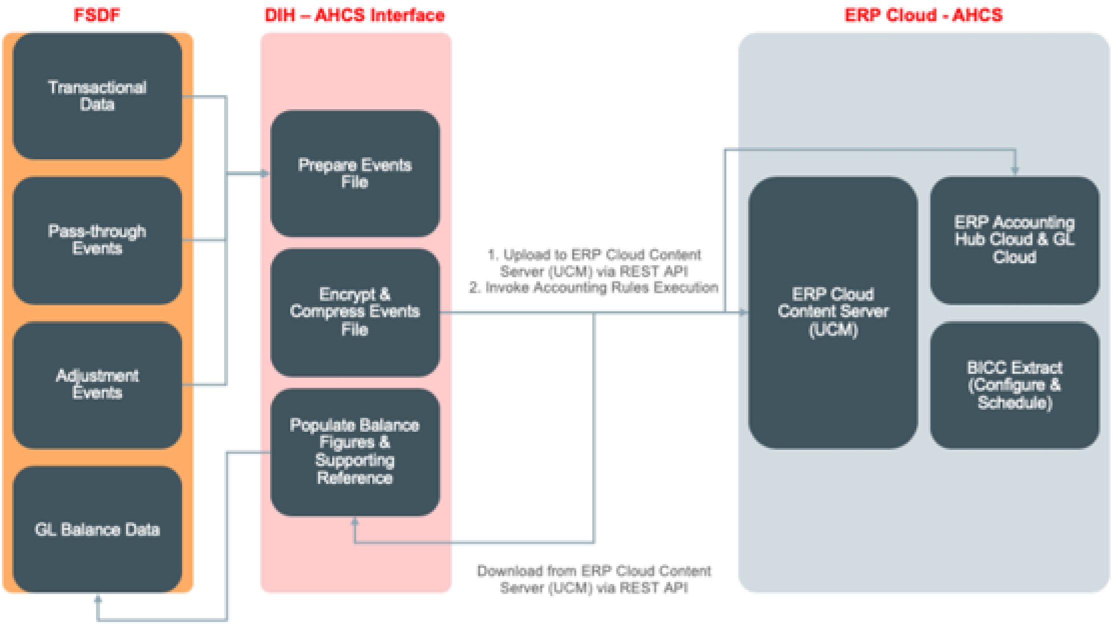
Banking data is extracted from OFSAA to AHC through DIH connectors (pre-packaged or custom) and is processed within the AHC using accounting rules. The updated GL balances are then extracted, to load General Ledger Balances and Supporting Reference Balances.

The flow from AHC to OFSAA is achieved through prebuilt DIH connectors. These connectors extract the GL and SR Balances from AHC and load them into the OFSAA staging post, which is available for all downstream applications.

2.6 OFSAA–AHC Integration Overview

The data flow between OFSAA and AHC takes place through files, as AHC is in the cloud environment.

Figure 2: OFSAA–AHC Integration

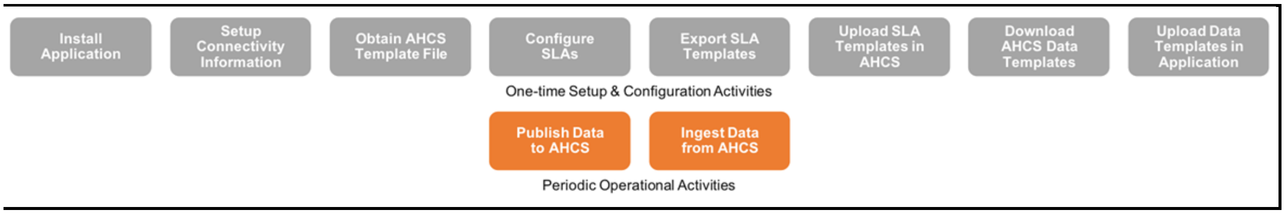

FSDF
DIH – AHCS Interface
ERP Cloud - AHCS
Transactional Data
Pass-through Events
Adjustment Events
GL Balance Data
Prepare Events File
Encrypt & Compress Events File
Populate Balance Figures & Supporting Reference
1. Upload to ERP Cloud Content Server (UCM) via REST API
2. Invoke Accounting Rules Execution
Download from ERP Cloud Content Server (UCM) via REST API
ERP Cloud Content Server (UCM)
ERP Accounting Hub Cloud & GL Cloud
BICC Extract (Configure & Schedule)

The data flow between OFSAA and AHC takes place through files, with communication over REST APIs.

The integration consists of the following processes:

1. Generate SLA templates in the application for registration flow with AHCS
2. Extract data from FSDF for AHCS using DIH Connectors
3. Load GL data from AHCS using DIH Connectors

AHCS provides a macro-enabled XLSM template for the definition of SLAs while integrating with third-party systems. This template is deployed as part of FSDF Integration with AHCS installation and must be obtained from your instance of AHCS before setup or configuration.


Install Application
Setup Connectivity Information
Obtain AHCS Template File
Configure SLAs
Export SLA Templates
Upload SLA Templates in AHCS
Download AHCS Data Templates
Upload Data Templates in Application
One-time Setup & Configuration Activities
Publish Data to AHCS
Ingest Data from AHCS
Periodic Operational Activities

The sequence of steps are as follows:

1. Set up the application with information including details of your AHCS instance and required credentials.
2. Configure Sub Ledger Applications, required data attributes, and transaction types.
3. Register Sub Ledger Applications with AHCS:
    a. Obtain the XLSM template file from your instance of AHCS

b. FSDF Integration with AHCS (the application) automates the population of the templates required for SLA registration by AHCS. This is relevant for SLAs defined out of the box and those configured by you.

c. Manually validate the filled templates using the macro-enabled functions built into them.

d. In case of errors, make required changes in the application and repeat the process.

e. After completion, use the built-in function to generate a ZIP-format file from the filled, validated template files.

   i. The file contains metadata of the source system that has to be registered, along with details of the SLA, its required attributes, and transaction types. The sequence of steps needs to be performed for each SLA defined in the application. See Registering with AHCS section below for more details.

f. The ZIP-format file thus generated must needs to be uploaded manually into your instance of AHCS.

g. After completion, Data Template for the SLA needs to be downloaded from AHCS and uploaded against the SLA in the application.

h. Configure your instance of AHCS as per financial accounting requirements.

4. After the setup, configuration, and registration steps are completed, execute batches of tasks that are automatically configured by the application. The application processes events, pass-throughs, and adjustments data, prepare required data files, format them appropriately, publish them to AHCS using its interfaces, and optionally, trigger processing of accounting rules in AHCS.

   a. Configuration and processing of accounting rules, journals, sub-ledgers, and General Ledger are performed in your instance of AHCS.

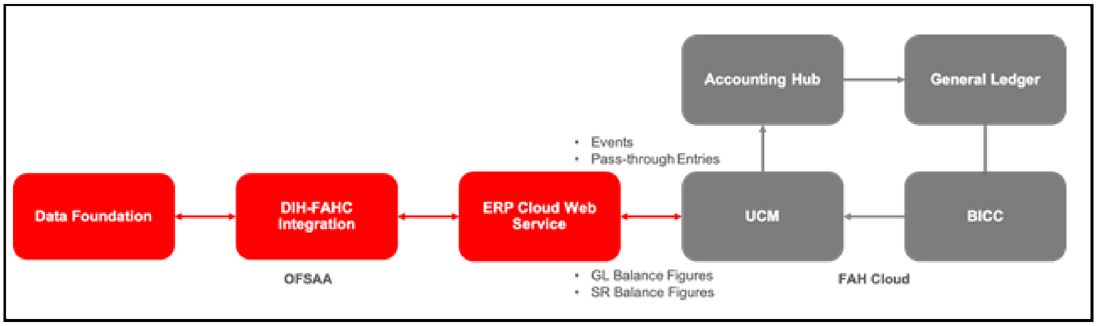
   b. Your instance of AHCS also needs to sequence or schedule production of GL balance and balance information with Supporting Reference using its standard processes

5. On completion of AHCS processing, including the production of files with GL balance and supporting reference balance information, execute a batch of tasks that obtains the files and load data into General Ledger and Management Leger entities in Data Foundation.

2.7 Data Flow for FSDF Integration with AHCS

This section explains the data flow for FSDF Integration with AHCS.

Figure 3: Data Flow for FSDF Integration with AHCS



The FSDF Integration with AHCS is as follows:

- Transactions and pass-through accounting information are collated into Data Foundation through DIH Foundation.
- If required, said data is standardized in Data Foundation.
- DIH – Accounting Hub integration extracts relevant data from Data Foundation and prepares ZIP file, with header and line details, and metadata.
- DIH – Accounting Hub integration invokes ERP Cloud web service over REST APIs to a byte-stream ZIP file to ERP Cloud Universal Content Manager.
- See Oracle ERP Financials documentation for information on Accounting Hub Cloud. Note that, the URL for ERP Financials is accurate at the time of publishing this manual and may change with time. See your AHCS tenancy documentation for up-to-date details.
- Extracts of GL and supporting reference balance figures from Accounting Hub Cloud Service are configured and scheduled in BI Cloud Connector console, as detailed here.
- GL and SR balance extracts so configured in BICC produce output files and put those in UCM.
- Data Foundation Integration with Accounting Hub Cloud invokes an ERP Cloud web service over REST APIs to obtain the files produced by BICC.
- Data Foundation Integration with Accounting Hub Cloud loads GL and SR balance figures into appropriate Staging entities in Data Foundation as per mapping information detailed in this guide.

3 Mapping OFSAA User to AHC User Groups

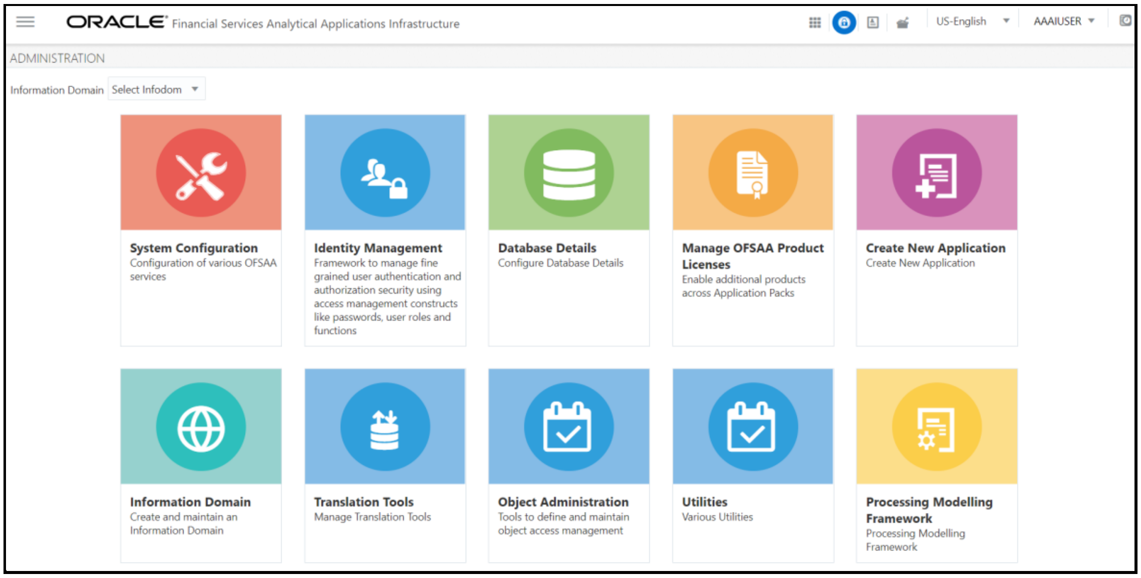
User group mapping enables you to map users or users to a specific user group that in turn is mapped to a specific Information Domain and role. Every user group mapped to the Infodom must be authorized.

The User Group Map window displays fields such as **User ID**, **Name**, and the corresponding **Mapped Groups**. You can view and modify the existing mappings within the **User Group Maintenance** window.

To access User Group Mapping, follow these steps:

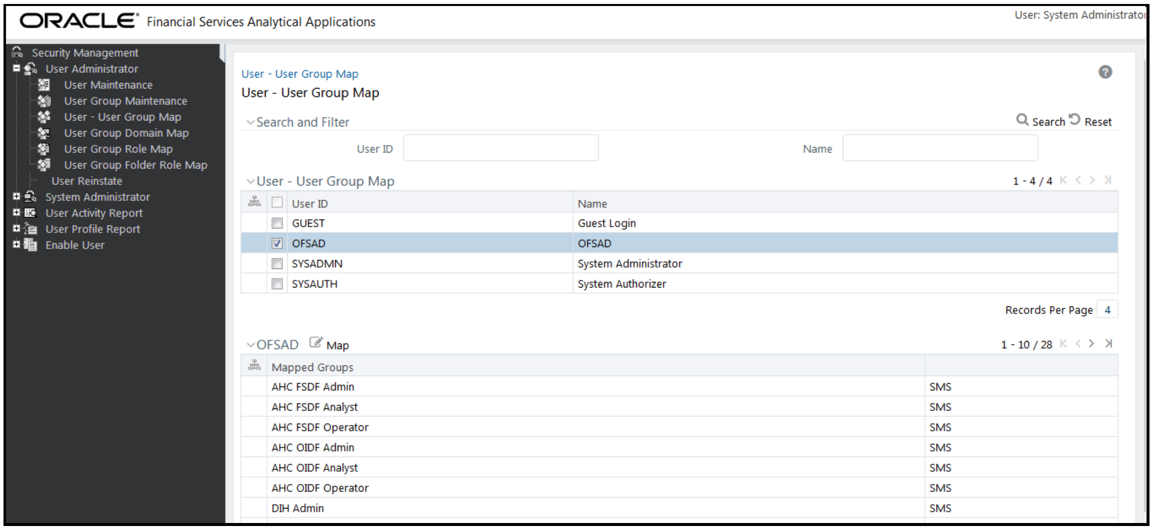
1. From the **Oracle Financial Services Analytical Applications Infrastructure** window, click **Administration** from the **Header** to display the Administration tools.

   Figure 4: OFS AAI Administration Window

   
   ORACLE Financial Services Analytical Applications Infrastructure
US-English
AAAIUSER
ADMINISTRATION
Information Domain
Select Infodom
System Configuration
Configuration of various OFSAA services
Identity Management
Framework to manage fine grained user authentication and authorization security using access management constructs like passwords, user roles and functions
Database Details
Configure Database Details
Manage OFSAA Product Licenses
Enable additional products across Application Packs
Create New Application
Create New Application
Information Domain
Create and maintain an Information Domain
Translation Tools
Manage Translation Tools
Object Administration
Tools to define and maintain object access management
Utilities
Various Utilities
Processing Modelling Framework
Processing Modelling Framework

2. Select **Identity Management** from the Tiles menu to view the **Security Management** window or click the **Navigation Menu** to access the Navigation List.
3. Select **Identity Management** to view the **Security Management** window. It is possible to map the users to the user groups in this window.

Figure 5: OFS AAI Security Management Window



User - User Group Map

User - User Group Map 1 - 4 / 4

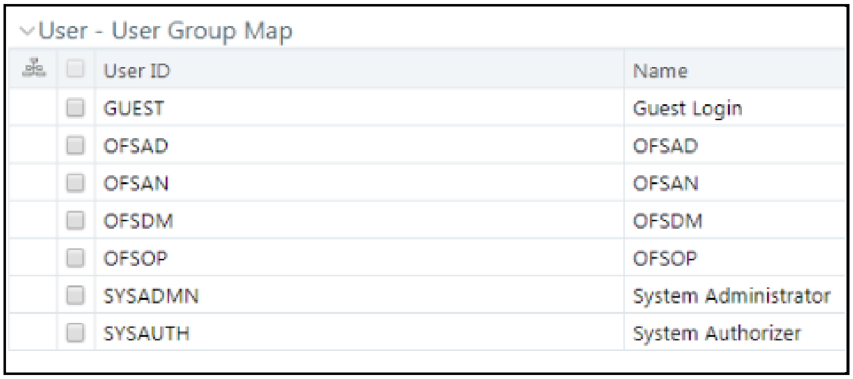
	User ID	Name
□	GUEST	Guest Login
☑	OFSAD	OFSAD
□	SYSADMN	System Administrator
□	SYSAUTH	System Authorizer

OFSAD 1 - 10 / 28

Mapped Groups	
AHC FSDF Admin	SMS
AHC FSDF Analyst	SMS
AHC FSDF Operator	SMS
AHC OIDF Admin	SMS
AHC OIDF Analyst	SMS
AHC OIDF Operator	SMS
DIH Admin	SMS

For details on mapping users to user groups, see the OFSAAI User Guide.

Figure 6: Pre-packaged User Groups for OFSAA - AHC Interface



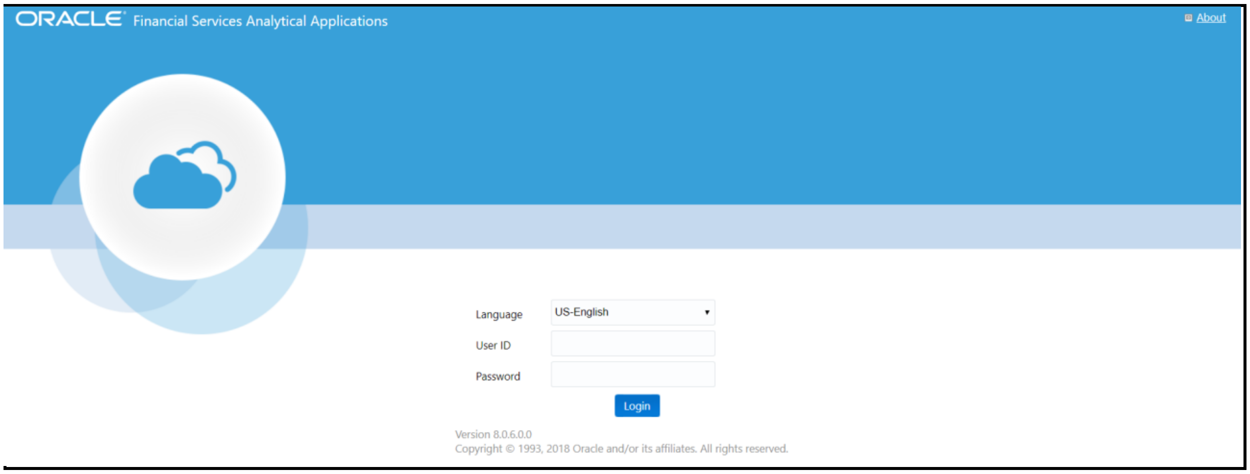
User - User Group Map

	User ID	Name
□	GUEST	Guest Login
□	OFSAD	OFSAD
□	OFSAN	OFSAN
□	OFSDM	OFSDM
□	OFSOP	OFSOP
□	SYSADMN	System Administrator
□	SYSAUTH	System Authorizer

4 Accessing the AHC-OFSAA Interface

Access the AHC-OFSAA Interface using your login credentials (User ID and Password). The built-in security system ensures that you are permitted to access the window and actions based on authorization only.

Figure 7: AHC - OFSAA Interface Login Window

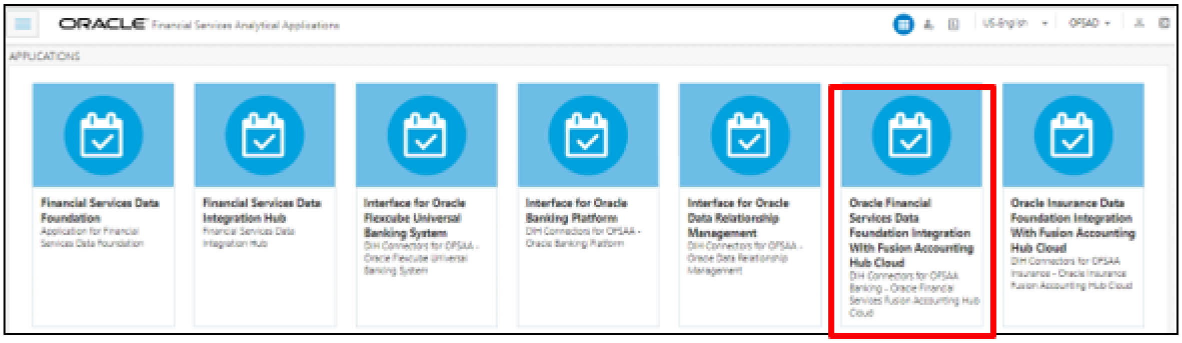


Language: US-English

Version 8.0.6.0.0
Copyright © 1993, 2018 Oracle and/or its affiliates. All rights reserved.

After logging into the application, select **Oracle Financial Services Data Foundation Integration with Accounting Hub Cloud**.

Figure 8: Applications Window


ORACLE Financial Services Analytical Applications
US-English
OFSAD
APPLICATIONS
Financial Services Data Foundation
Application for Financial Services Data Foundation
Financial Services Data Integration Hub
Financial Services Data Integration Hub
Interface for Oracle Flexcube Universal Banking System
DIH Connectors for OFSAA - Oracle Flexcube Universal Banking System
Interface for Oracle Banking Platform
DIH Connectors for OFSAA - Oracle Banking Platform
Interface for Oracle Data Relationship Management
DIH Connectors for OFSAA - Oracle Data Relationship Management
Oracle Financial Services Data Foundation Integration With Fusion Accounting Hub Cloud
DIH Connectors for OFSAA Banking - Oracle Financial Services Fusion Accounting Hub Cloud
Oracle Insurance Data Foundation Integration With Fusion Accounting Hub Cloud
DIH Connectors for OFSAA Insurance - Oracle Insurance Fusion Accounting Hub Cloud

NOTE You must be mapped to any one of the AHC user groups.

5 Component Details

The following components are included in this interface:

- Pre-packaged Subledger Application Information
- Template to Register Source System
- Data Integration Hub (DIH) Connectors

5.1 Packaged Subledger Application Information

The following Subledger Applications are predefined as follows:

- Retail Banking
- Commercial Banking
- Investment Banking
- Treasury
- Islamic Banking
- Core Banking PT
- Banking Adjustments

The list of product types covered under each Subledger Application is provided in the SLA Coverage file.

Each Subledger Application is represented as an event class, which can have one or more instruments such as Loans, Annuities, and Investments, and so on. For each event class, there are predefined accounting events based on expected activities for the Subledger Application.

Sub ledgers can have two types of templates:

- Transactions-based: This indicates that AHC receives transaction information, which is not preformatted for accounting. There are no debits or credits.
- Based on Pass-through accounting entries: There are debits and credits from the source system.

The rules vary based on the templates used.

Table 2: Subledger Applications

Subledger Application	Type
Retail Banking	Transaction based
Commercial Banking	Transaction based
Investment Banking	Transaction based
Treasury	Transaction based
Islamic Banking	Transaction based
Core Banking PT	Based on Pass-through accounting entries
Banking Adjustments	Transaction based

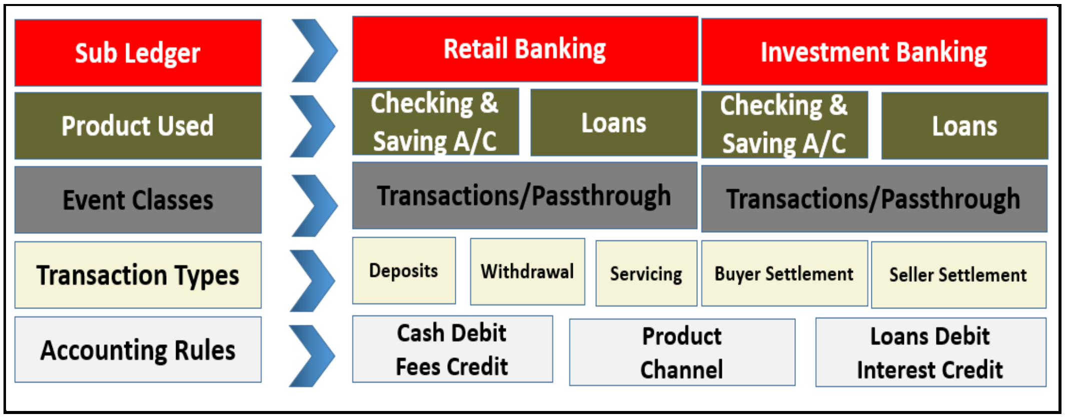
Subledger Information has transaction types mapped, along with the list of transaction and line attributes defined for each Subledger Application.

The FSDF data model is registered with AHC using the Excel template. The attributes of the transactions pass through entries, and the templates use the product processors. They are also available for creating new rules configurations to tailor accounting treatments.

You can use these pre-packaged Subledgers or create custom Subledger Applications in the Oracle Financial Services Data Foundation Integration with Accounting Hub Cloud Service.

The following illustration displays an example of the structure for accounting templates for OFSAA integration.

Figure 9: Structure for Accounting Templates



The Retail Banking Subledger Application has two process products involved: Checking and Savings Accounts and Loans. Each of these products can have one of the two event classes: Transactions or Pass-through entries. Note that pre-packaged Subledger Applications use only one event class per Subledger.

Various transaction types are registered for each of the event classes. For checking and savings accounts, transaction types include deposits, withdrawals, and servicing. For loans, transaction types include originations, payments, and closings. These transaction types and templates can be later used to configure rules in Accounting Hub Cloud Service.

Supporting references, which provide the ability to create accounting balances at a more detailed level than is provided by the general ledger chart of accounts, are available across all the Subledger Applications provided for the integration. Product and channel are examples of potential supporting references.

5.2 Registering with AHCS

AHCS uses a macro-enabled XLS template to facilitate registration of transaction or event types and the creation of a catalog of transaction information, towards preparing accounting rules. The template is defined by AHCS and made available as an XLSM file from your AHCS instance. FSDF Integration with AHCS populates it with SLA specifications in an automated manner.

On initial use (that is, when you have freshly installed FSDF Integration with AHCS) and after user-configured changes to any aspect of Subledger Applications or their events information (event types, header information, line information, or SLA packaging), the following actions must be undertaken for all or all affected SLAs:

1. Access your instance of AHCS and download the template file

   `XlaSourceSystemSetup.xlsm`. See the Oracle ERP Cloud documentation (**Learn More** > **Documentation Library**) on how this can be performed.

The sheets in this XLSM file are as follows:

a. **Instructions**: This sheet provides a brief overview of the template structure alongside instructions to fill in details and create the final ZIP file that is to be uploaded to your AHCS instance.

b. **Source System**: This sheet has sections that capture the source system name and transaction types, which are used to register the source system in AHC.

   - Source System: Name and short name are mandatory. This goes as the name for Source, Subledger, Journal Source, Journal Category, and Process Category.
   - Transaction Types: Name and short name for all the event or transaction types are mandatory in this field.
   - There are **Validate** and **Generate ZIP** buttons available on the Source System sheet. These support functions are detailed in the Registration of SLAs with Accounting Hub Service section.

c. **Transaction Information**: This sheet captures the attributes that you wish to use as header attributes in AHCS. You must enter the Name, Short Name, Domain, and Journal Display. By default, three mandatory attributes are displayed, which cannot be removed. These are highlighted in grey.

d. **Line Information**: This sheet captures the attributes that you wish to use as line attributes in AHCS. You must enter Name, Short Name, Domain, and Chart of Accounts Display. By default, three mandatory attributes are displayed, which cannot be removed. These are highlighted in grey.

2. See the Oracle ERP Cloud documentation (**Learn More** > **Documentation Library**) for further details on AHCS.

5.3 DIH Connectors

This section explains Extract and Insert Connectors.

5.3.1 Extract Connectors

The DIH connectors extract data from OFSAA staging tables, which must be created as part of creating Subledger in the application. These connectors extract data from OFSAA staging tables in ASCII file format per SLA definitions.

There are two External Data Descriptors (EDD) that are created for each Subledger that is saved in FSDF Integration with AHCS. One EDD is for the Header file and another for the Line file. Each EDD has the same structure as the transaction and line attributes that are configured for a Subledger, respectively. The attributes of the Subledger, which are defined in these EDDs, are mapped to the attributes of the staging tables.

There are predefined connectors that are created as part of pre-packaged Subledgers. EDDs and connectors are already provided for pre-packaged Subledgers.

All **Header** Connectors accept the following input parameters:

- **FIC_MIS_DATE or MIS_DATE** is the Extraction Date.
- **LEDGER_NAME:** The name of the Primary Ledger is a runtime parameter populated automatically from Subledger details.

- **SLA_CODE:** Subledger Code or Short Name is a runtime parameter populated automatically from Subledger details.
- **AH_GAAP_CODE:** is a constant Parameter to be updated in the DIH Parameters section before executing a batch.
- **AH_LOAD_RUN_ID:** Unique task ID that is auto-picked while execution of the task.
- **AH_FILE_DATE** is the Extraction Date whose file format must not be altered.

All **Line** Connectors accept the following input parameters:

- **FIC_MIS_DATE or MIS_DATE** is the Extraction Date.
- **SLA_CODE:** Subledger Code or Short Name is a runtime parameter populated automatically from Subledger details (for example, RETAIL_SLA for Retail Banking).
- **AH_LOAD_RUN_ID:** Unique task ID that is auto-picked while execution of the task.
- **AH_FILE_DATE:** is the Extraction Date whose file format must not be altered.

For Extract Connectors and mapping details, see the Extract Connectors Structure file.

5.3.2 Insert Connectors

The connectors have prebuilt mappings between AHC source views for GL or SR balances and OFSAA staging entities meant to store ledger data.

The following connectors are provided by AHC for GL and SR balances:

- **AH GL Balances Insert Con**: Connector to load only one ledger specific data for a given execution into the STG_GL_DATA table in OFSAA. This connector accepts these input parameters:
  - **PERIOD_NAME** Period for which GL data is loaded (for example, 9-Sep).
  - **FIC_MIS_DATE or MIS_DATE** is the Extraction Date.
  - **AH_CONSOLIDATION_FLAG** Runtime parameter and is a single-digit flag (for example, C or S).
  - LEDGER_ID for which GL balances must be filtered (for example, LEDGER_ID='1234').
- **AH SR Balances Insert Con**: Connector to load thick ledger or supporting reference balances into the STG_MANAGEMENT_LEDGER tables in OFSAA. This connector accepts these input parameters:
  - **PERIOD_NAME** Period for which GL data is loaded.
  - **FIC_MIS_DATE or MIS_DATE** is the Extraction Date.

For Connectors and mapping details, see the Insert Connectors Structure file.

6 Implementation

6.1 Background

Transaction objects in AHC are tables defined for each event class, capturing source transaction data for accounting events. The Create Accounting process gets the source transaction data from the transaction objects to generate journal entries. There are different types of transaction objects indicating whether they are used at the header or line level.

Header sources have the same value for all transaction lines or distributions associated with an accounting event. These sources are associated with a transaction header or with transaction reference data. Line sources have values that can vary by the transaction lines or distributions associated with an accounting event. They must be stored in the transaction objects at the line level.

Two separate header tables, one for each flow type is introduced in the OFSAA staging area for defining header and one Line table is introduced for defining line sources in AHC:

- Stage Transaction Header (STG_TXN_HEADER) for event-based accounting.
- Stage Accounting Entries Header (STG_ACCT_ENTRIES_HEADER) for pass-through accounting.
- Associated Transaction Line Table for Line accounting.

  For example, STG_ANNUITY_TXNS for event-based Accounting and STG_ACCOUNTING_ENTRIES for pass-through accounting.

6.2 Implementation Guidelines

The implementation guidelines are as follows:

1. Transaction numbers must match in the header and line files of import data files in AHC. You must carefully create connectors to extract the transaction numbers that have to match accordingly.
2. Ensure to provide the correct ledger name in the FSDF Integration with AHCS screens while configuring the Subledger.
3. Do not execute the connector’s batch in FSDF Integration with AHCS until the accounting rules and other required setup are configured.
4. Create connectors with appropriate filters, to extract only the required data from a table. After a set of transaction numbers are processed in AHC, it does not accept the same transaction numbers again for a Subledger. You must carefully extract the required data for appropriate accounting.
5. If the SLA template must be reloaded to AHC after changes, you must cautiously replace the old configuration and again create the accounting rules accordingly.
6. Only those event type names, which are configured in AHC, can be processed for accounting. If any other event type names come as part of data files imported to AHC, the whole file is not processed.
7. After the SLA template is uploaded and the source system is registered, for any further changes, ensure the following steps are performed in both FSDF Integration with AHCS and AHC:
   - If you add, edit, or delete any of the attribute or transaction types in the AHC screens, but do not update the respective changes in the FSDF Integration with AHCS application, integration may fail, as data extract files still have the unchanged structure of DIH.
   - If you add, edit, or delete any of the attribute or event type in the FSDF Integration with AHCS screens, but do not download the new template and re-upload it back to the AHC application, integration may fail. In this scenario, the AHC application has a different

Subledger configuration and the DIH extract connectors have a different configuration, which can lead to failure in data import.

- If there are any changes made in the source system or Subledger structure either in FSDF Integration with AHCS or AHC, other application structures must also be changed accordingly. If not, it may lead to integration failure.

8. Through FSDF Integration with AHCS, there is no mechanism to identify those transactions, which are not processed for accounting. You must identify them through AHC screens and re-upload them separately, after making the required corrections.

6.3 Transaction Object Population

Transaction objects are extracted into Header and Line CSV files and converted to zip format along with the Metadata text file.

1. In Subledger Applications, each Subledger have Header and Line Objects defined along with Transaction types.
2. Header and Line Connectors are created to extract Header and Line data, respectively.
3. Connectors are published and the auto-generated batch is executed.
4. After the successful execution of all Tasks in a batch, Header information is extracted to the `XlaTrxH.csv` file and Line information to the `XlaTrxL.csv` file.
5. The `Metadata.txt` file is auto-populated with Subledger details.
6. All three files: `XlaTrxH.csv`, `XlaTrxL.csv`, and `Metadata.txt` are auto-converted to zip format, which can be uploaded to AHC.

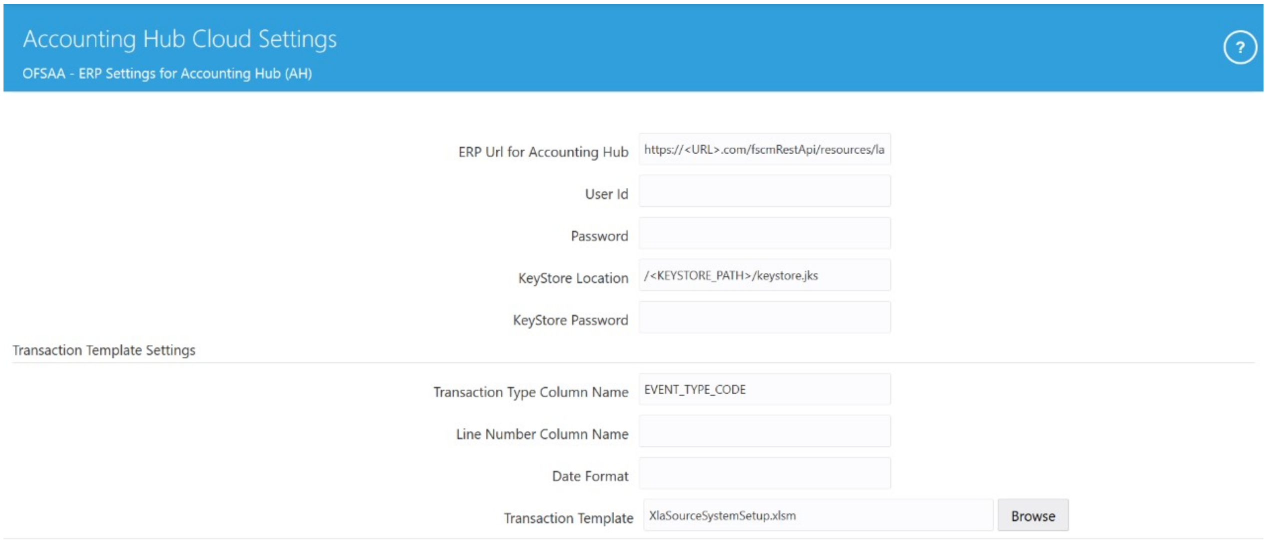
7 AHC Administration

7.1 AHC Settings

To complete the AHC setup, follow these steps

1. Log into the application and select **Oracle Financial Services Data Foundation Integration With Fusion Accounting Hub Cloud**.
2. Select the **AHC** Administration and then select **AHC Settings**. The OFSAA - ERP Settings for Accounting Hub (AH) window is displayed.

Figure 10: OFSAA - ERP Settings for Accounting Hub (AH) Window



Accounting Hub Cloud Settings

OFSAA - ERP Settings for Accounting Hub (AH)

ERP Url for Accounting Hub	https://<URL>.com/fscmRestApi/resources/la
User Id	
Password	
KeyStore Location	/<KEYSTORE_PATH>/keystore.jks
KeyStore Password	

Transaction Template Settings

Transaction Type Column Name	EVENT_TYPE_CODE
Line Number Column Name	
Date Format	
Transaction Template	XlaSourceSystemSetup.xlsm

3. Enter the required information in the **ERP Details for Accounting Hub** and **Transaction Template Settings** panes.

   a. Enter the following details in the **ERP Details for Accounting Hub** pane:

   — **URL for Accounting Hub Cloud Service**:

   Enter the URL assigned to you as part of your Oracle ERP Cloud AHCS subscription here. This information is specific to your tenancy on Oracle Cloud and subscription to ERP Cloud AHCS instance.

   The URL is in the following form:

   ```
   https://<<YourInstance>>/fscmRestApi/resources/latest/erpintegrations
   ```

   Of this URL, the section

   "`/fscmRestApi/resources/latest/erpintegrations`" must be left as it is.

   Set `<<YourInstance>>` as assigned to you as part of your Oracle ERP Cloud AHCS subscription.

   The captured URL must resemble this example:

   ```
   https://abc.de.fg.oraclecloud.com//fscmRestApi/resources/latest/erpintegrations
   ```

- **User ID**:

  Enter User Identification assigned to you for AHCS instance here.

- **Password**:

  Enter the password for your AHCS instance here.

- **KeyStore Location**:

  Enter a fully resolved file-path for your KeyStore location here.

- **KeyStore Password**:

  Enter the password for your KeyStore here.

NOTE

- The certificate must be imported to KeyStore before connection with AHCS is attempted.
- See the Importing Certificate into KeyStore section for details on importing Certificate to KeyStore.

b. Enter the following details in the Transaction Template Settings pane:

- **Transaction Type Column Name**:

  This is prepopulated with the value TRANSACTION_TYPE, the auto-generated value for the Transaction Type column name in the Register Transaction Source System step.

  If the auto-generated value for Transaction Type column name in the Register Transaction Source System step is modified, edit the prepopulated text and capture the assigned column name here.

- **Line Number Column Name**:

  This is prepopulated with the value LINE NUMBER, the auto-generated value for Line Number column name in the Register Transaction Source System step.

  If the auto-generated value for the Line Number column name in the Register Transaction Source System step is modified, edit the prepopulated text and capture the assigned column name here.

- **Date Format**:

  Enter the date format in which your instance of AHCS will expect the date columns to be provided. If not entered, the date field uses the default format.

- **Transaction Template**:

  Transaction Template refers to the template contained in the XLSM file downloaded from your instance of AHCS to facilitate registration of transaction or event types. See the Registering with AHCS section, for details.

  Browse and select the transaction template file `XlaSourceSystemSetup.xlsm` from the downloaded location in the local machine.

c. Click **Save**.

d. After it is saved, the settings entered is used while producing the Transaction Template. This is a one-time setup activity. Do not change these settings unless required by changes to your instance of AHCS.

e. If settings are altered and saved, the new settings applies to all SLA Header and Line files processed for publishing to AHCS from then on.

7.2 Refresh Interface

The steps under this section must be performed to complete application setup by deploying out-of-the-box contents such as ADIs, EDDs, and Connectors.

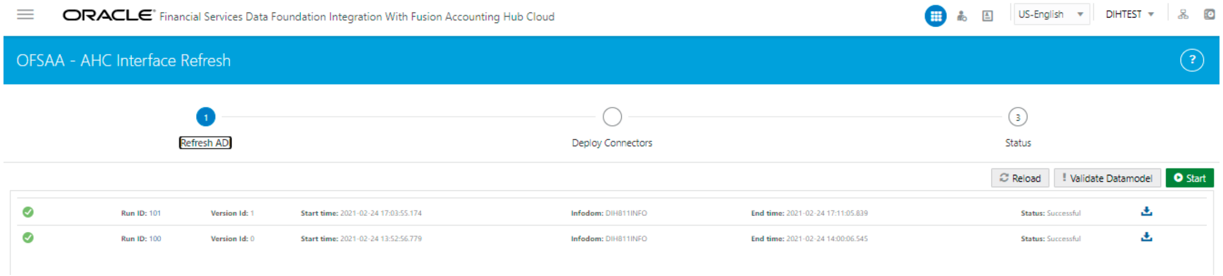
The Refresh Interface window contains the following steps:

- Refreshing the ADIs
- Deploying the Connectors
- Reviewing the Status

7.2.1 Refreshing ADIs

The first logical step in setting up the application for use is to refresh the Application Data Interface information. Application Data Interface (ADI) is a logical abstraction of the OFSAA Data Foundation, data model that you interact with as you use them for data exchanges for the purpose of integration with Fusion Accounting Hub. The "Refresh ADI" step reads the data model uploaded into your instance of OFSAA and creates ADIs. This step is relevant both when the application is being setup after fresh installation and when the data model is subsequently uploaded with incremental changes. The "ADI Refresh" process compares the existing ADIs with those based on the updated model and reflects changes appropriately with additional ADIs, additional attributes for existing ADIs, or new data types for existing attributes, as applicable.

Figure 11: OFSAA – AHC Interface Refresh


ORACLE Financial Services Data Foundation Integration With Fusion Accounting Hub Cloud
US-English
DIHTEST
OFSAA - AHC Interface Refresh
1
Refresh ADI
Deploy Connectors
3
Status
Reload
Validate Datamodel
Start

✓ Run ID: 101 Version Id: 1 Start time: 2021-02-24 17:03:55.174 Infodom: DIH811INFO End time: 2021-02-24 17:11:05.839 Status: Successful

✓ Run ID: 100 Version Id: 0 Start time: 2021-02-24 13:52:56.779 Infodom: DIH811INFO End time: 2021-02-24 14:00:06.545 Status: Successful

1. Click **Start** to start the refresh of ADIs.
2. The ongoing ADI refresh is displayed. This refreshes all the Application Data Interfaces and creates the Application Data Interfaces for all the staging tables present in the model, which is being uploaded in the same Infodom.

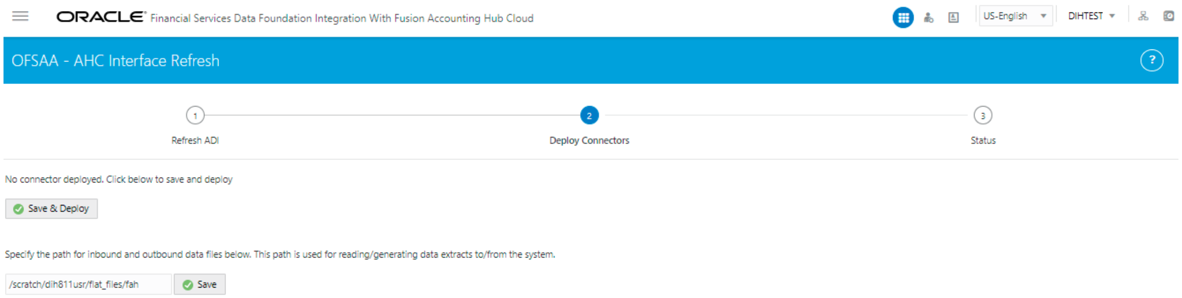
> **NOTE** See the DIH User Guide for detailed information.

7.2.2 Deploying Connectors

After the Refresh ADI is complete, External Data Descriptors and Connectors linking them to Application Data Interfaces, as used for data exchange with AHCS, are ready for deployment through the Deploy Connectors section. These External Data Descriptors and Connectors are listed under the AHC Data Mapping menu after this step is completed.

To deploy OFSAA-AHC Connectors, follow these steps:

Figure 12: OFSAA – AHC Interface Refresh **– Deploy Connectors**


ORACLE Financial Services Data Foundation Integration With Fusion Accounting Hub Cloud
US-English
DIHTEST
OFSAA - AHC Interface Refresh
?
1
Refresh ADI
2
Deploy Connectors
3
Status
Save & Deploy
Save

No connector deployed. Click below to save and deploy

Specify the path for inbound and outbound data files below. This path is used for reading/generating data extracts to/from the system.

/scratch/dih811usr/flat_files/fah

1. In the **File Location** field, specify the location where data files for AHCS integration must be generated and used subsequently for data exchange.

   Note that this also refers to AHCS as a source system for GL and SR Balance information, relevant to the Insert Connectors.

2. Click **Save & Deploy**.

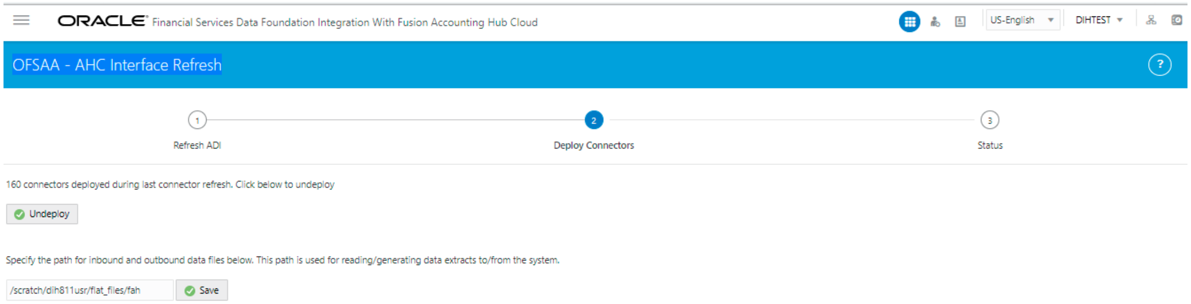
   A confirmation message is displayed "Are you sure you want to Deploy?".

3. Click **Yes**.

   After successful completion of the process, a confirmation message is displayed "Successfully Deployed".

4. If there is a subsequent requirement to re-deploy Connectors due to application upgrade, they must be Undeployed, and Deployed again. The Undeploy button can also be used for the purpose of restoration of out-of-box system data mappings should the need arises. The **Undeploy** option is available only after successful deployment of connectors. Click on Undeploy button to proceed.

Figure 13: OFSAA – AHC Interface Refresh – UnDeploy Connectors


ORACLE Financial Services Data Foundation Integration With Fusion Accounting Hub Cloud
US-English
DIHTEST
OFSAA - AHC Interface Refresh
?
1
Refresh ADI
2
Deploy Connectors
3
Status
Undeploy
Save

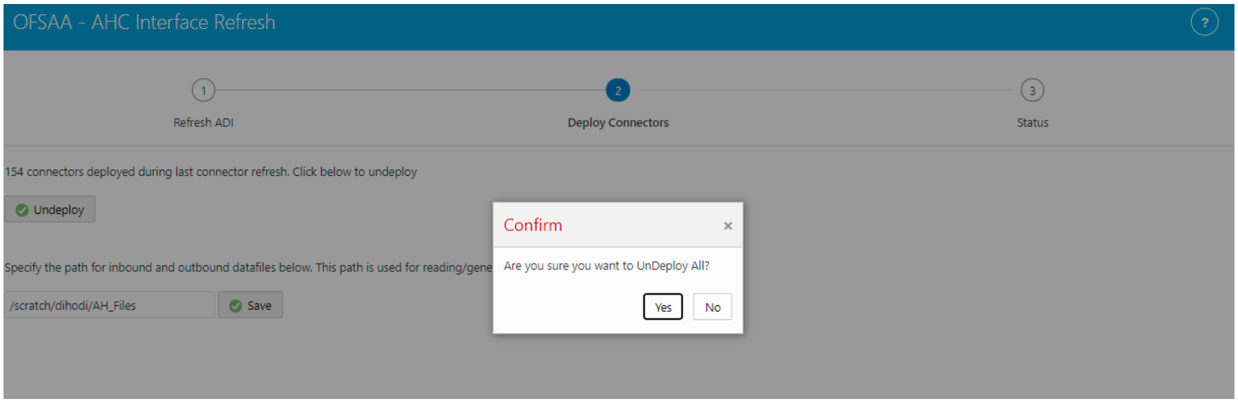
160 connectors deployed during last connector refresh. Click below to undeploy

Specify the path for inbound and outbound data files below. This path is used for reading/generating data extracts to/from the system.

/scratch/dih811usr/flat_files/fah

A confirmation dialog box is displayed with the message: "Are you sure you want to Undeploy All?".

Figure 14: OFSAA – AHC Interface Refresh **– UnDeploy Connectors Confirmation Message**



OFSAA - AHC Interface Refresh

154 connectors deployed during last connector refresh. Click below to undeploy

Specify the path for inbound and outbound datafiles below. This path is used for reading/gene

/scratch/dihodi/AH_Files

Confirm

Are you sure you want to UnDeploy All?

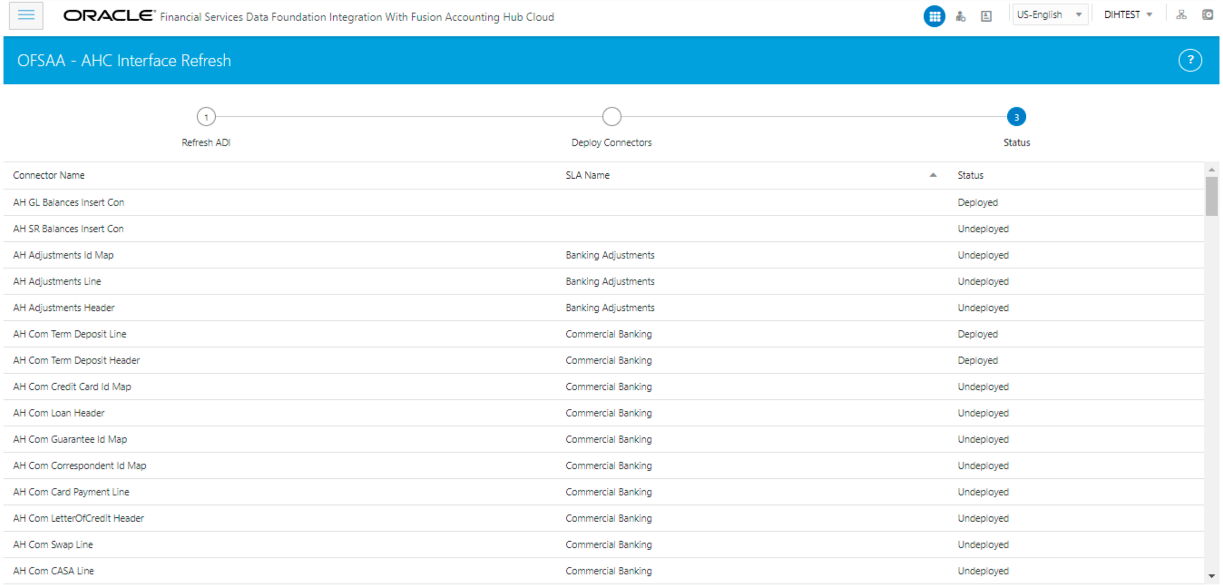
5. Click **Yes** to undeploy all the Connectors.

> **NOTE** Interface Connectors can be undeployed only if:
>
> 1. None of the Interface Connectors is in a Published state.
> 2. The EDDs deployed through the interface version are not used in any Custom Connectors.

7.2.3 Reviewing the Status

You can view the list of out-of-box Connectors available as part of the application and their deployment status. All the connectors must be in deployed status. Subsequently the detailed view of the deployed connectors is accessible under **Data Mapping** section.

Figure 15: OFSAA – AHC Interface Refresh – **Status**



ORACLE Financial Services Data Foundation Integration With Fusion Accounting Hub Cloud
US-English
DIHTEST

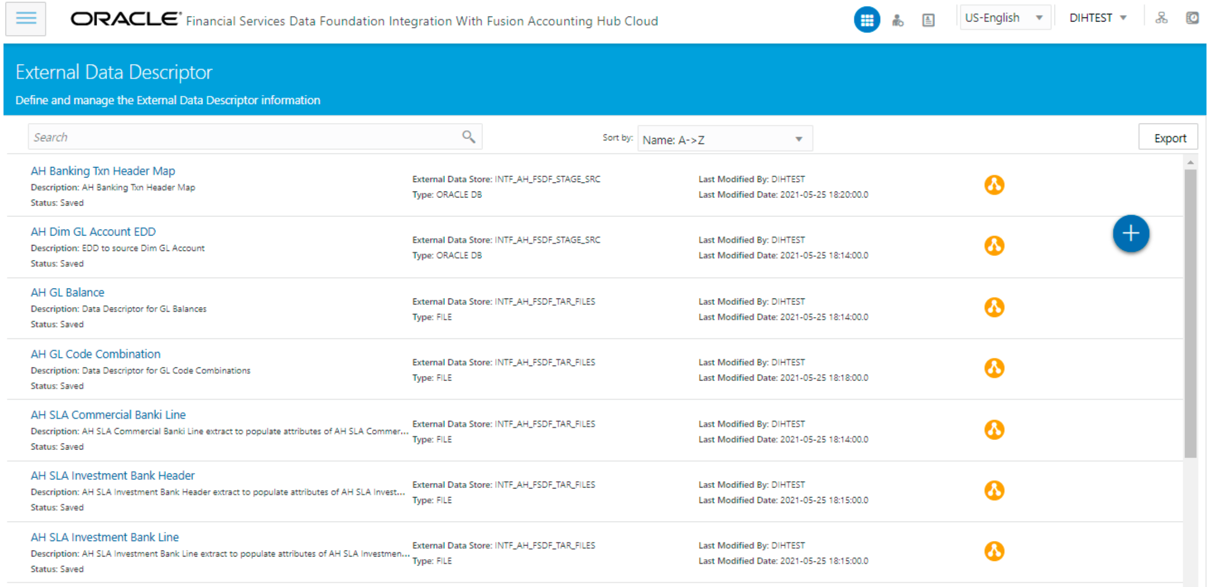
OFSAA - AHC Interface Refresh

1 Refresh ADI
Deploy Connectors
3 Status

Connector Name	SLA Name	Status
AH GL Balances Insert Con		Deployed
AH SR Balances Insert Con		Undeployed
AH Adjustments Id Map	Banking Adjustments	Undeployed
AH Adjustments Line	Banking Adjustments	Undeployed
AH Adjustments Header	Banking Adjustments	Undeployed
AH Com Term Deposit Line	Commercial Banking	Deployed
AH Com Term Deposit Header	Commercial Banking	Deployed
AH Com Credit Card Id Map	Commercial Banking	Undeployed
AH Com Loan Header	Commercial Banking	Undeployed
AH Com Guarantee Id Map	Commercial Banking	Undeployed
AH Com Correspondent Id Map	Commercial Banking	Undeployed
AH Com Card Payment Line	Commercial Banking	Undeployed
AH Com LetterOfCredit Header	Commercial Banking	Undeployed
AH Com Swap Line	Commercial Banking	Undeployed
AH Com CASA Line	Commercial Banking	Undeployed

1. To check deployed EDDs and Mappings, select **Data Mapping** from the **Oracle Financial Services Data Foundation Integration with Accounting Hub Cloud Service** window, and then select **External Data Descriptor.**

Figure 16: External Data Descriptor Summary Window



ORACLE Financial Services Data Foundation Integration With Fusion Accounting Hub Cloud
US-English
DIHTEST
Search
Export

External Data Descriptor

Define and manage the External Data Descriptor information

Sort by: Name: A->Z

AH Banking Txn Header Map
Description: AH Banking Txn Header Map
Status: Saved
External Data Store: INTF_AH_FSDF_STAGE_SRC
Type: ORACLE DB
Last Modified By: DIHTEST
Last Modified Date: 2021-05-25 18:20:00.0

AH Dim GL Account EDD
Description: EDD to source Dim GL Account
Status: Saved
External Data Store: INTF_AH_FSDF_STAGE_SRC
Type: ORACLE DB
Last Modified By: DIHTEST
Last Modified Date: 2021-05-25 18:14:00.0

AH GL Balance
Description: Data Descriptor for GL Balances
Status: Saved
External Data Store: INTF_AH_FSDF_TAR_FILES
Type: FILE
Last Modified By: DIHTEST
Last Modified Date: 2021-05-25 18:14:00.0

AH GL Code Combination
Description: Data Descriptor for GL Code Combinations
Status: Saved
External Data Store: INTF_AH_FSDF_TAR_FILES
Type: FILE
Last Modified By: DIHTEST
Last Modified Date: 2021-05-25 18:18:00.0

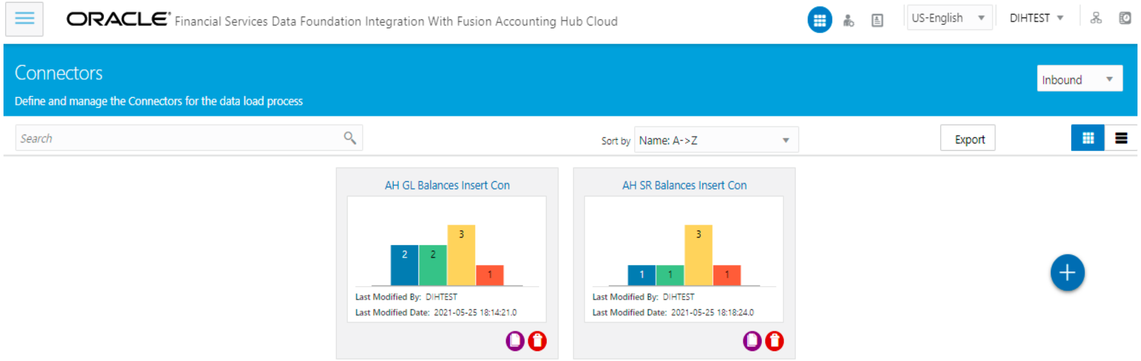
AH SLA Commercial Banki Line
Description: AH SLA Commercial Banki Line extract to populate attributes of AH SLA Commer...
Status: Saved
External Data Store: INTF_AH_FSDF_TAR_FILES
Type: FILE
Last Modified By: DIHTEST
Last Modified Date: 2021-05-25 18:14:00.0

AH SLA Investment Bank Header
Description: AH SLA Investment Bank Header extract to populate attributes of AH SLA Invest...
Status: Saved
External Data Store: INTF_AH_FSDF_TAR_FILES
Type: FILE
Last Modified By: DIHTEST
Last Modified Date: 2021-05-25 18:15:00.0

AH SLA Investment Bank Line
Description: AH SLA Investment Bank Line extract to populate attributes of AH SLA Investmen...
Status: Saved
External Data Store: INTF_AH_FSDF_TAR_FILES
Type: FILE
Last Modified By: DIHTEST
Last Modified Date: 2021-05-25 18:15:00.0

2. To check deployed Connectors and Mappings, select **Data Mapping** from the **Financial Services Data Foundation Integration with Accounting Hub Cloud Service** window, and then select **Connectors.**

Figure 17: Connectors Summary Window



ORACLE Financial Services Data Foundation Integration With Fusion Accounting Hub Cloud
US-English
DIHTEST
Inbound
Search
Export
2
2
3
1
1
1
3
1

Connectors

Define and manage the Connectors for the data load process

Sort by Name: A->Z

AH GL Balances Insert Con
Last Modified By: DIHTEST
Last Modified Date: 2021-05-25 18:14:21.0

AH SR Balances Insert Con
Last Modified By: DIHTEST
Last Modified Date: 2021-05-25 18:18:24.0

7.3 Transaction Types

Event or Transaction types refer to accounting events that are captured when transactions are committed or processed. While creating a Subledger, different transaction types are created so that all the transactions can be classified into one or the other event type and are used for creating journal lines. Examples of transaction types can be withdrawal, deposits, servicing, fees, charges, and so on.

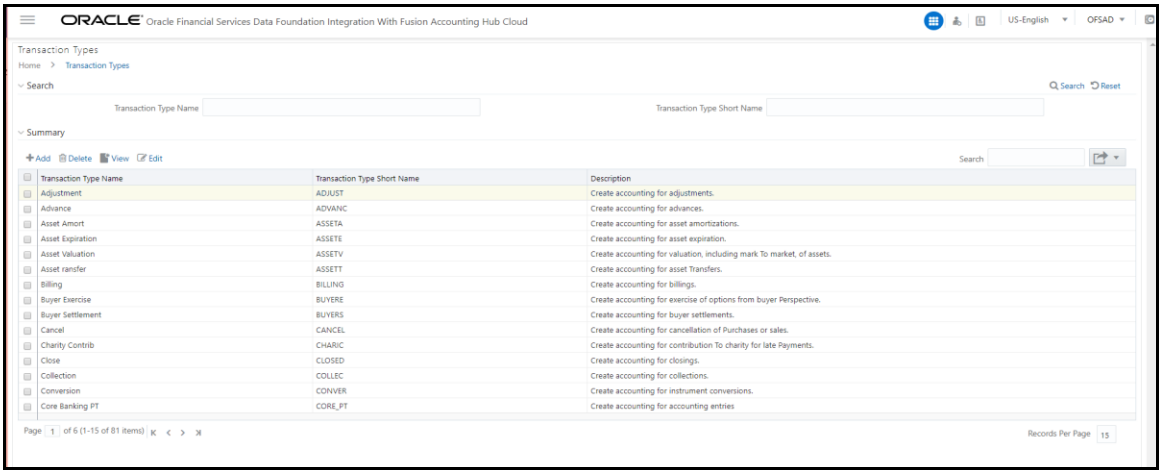
Transaction types are required to collect the transaction data and process it into accounting data necessary to form the Subledger information. Data from different source systems come together and are processed depending on these categories to form ledger information.

> **NOTE** To use pre-packaged Subledgers, you must use only the pre-packaged transaction types in the source system data.

To understand the transaction types, follow these steps:

1. From the **Oracle Financial Services Data Foundation Integration with Accounting Hub Cloud Service** window, select **Administration,** and then select **Transaction Types**. The **Transaction Types** window is displayed.

Figure 18: Transaction Types Summary window



ORACLE Oracle Financial Services Data Foundation Integration With Fusion Accounting Hub Cloud
US-English
OFSAD
Transaction Types
Home > Transaction Types
Search
Search
Reset
Transaction Type Name
Transaction Type Short Name
Summary
Add
Delete
View
Edit
Search
Page 1 of 6 (1-15 of 81 items)
Records Per Page 15

Transaction Type Name	Transaction Type Short Name	Description
Adjustment	ADJUST	Create accounting for adjustments.
Advance	ADVANC	Create accounting for advances.
Asset Amort	ASSETA	Create accounting for asset amortizations.
Asset Expiration	ASSETE	Create accounting for asset expiration.
Asset Valuation	ASSETV	Create accounting for valuation, including mark To market, of assets.
Asset ransfer	ASSETT	Create accounting for asset Transfers.
Billing	BILLING	Create accounting for billings.
Buyer Exercise	BUYERE	Create accounting for exercise of options from buyer Perspective.
Buyer Settlement	BUYERS	Create accounting for buyer settlements.
Cancel	CANCEL	Create accounting for cancellation of Purchases or sales.
Charity Contrib	CHARIC	Create accounting for contribution To charity for late Payments.
Close	CLOSED	Create accounting for closings.
Collection	COLLEC	Create accounting for collections.
Conversion	CONVER	Create accounting for instrument conversions.
Core Banking PT	CORE_PT	Create accounting for accounting entries

2. A list of pre-packaged transaction types is displayed on the window, which you can map with Subledgers accordingly. You cannot edit or delete these pre-packaged transaction types.

   You can search for Transaction Type Name or Short Name. Enter a part or full name and click the search to get the filtered result.

3. In the **Summary** pane, you can **Add**, **Delete**, **View,** or **Edit** the events.

4. Click ✚Add. The **Transaction Types** window appears.

5. Enter the following details to create a new event and click **Save**.

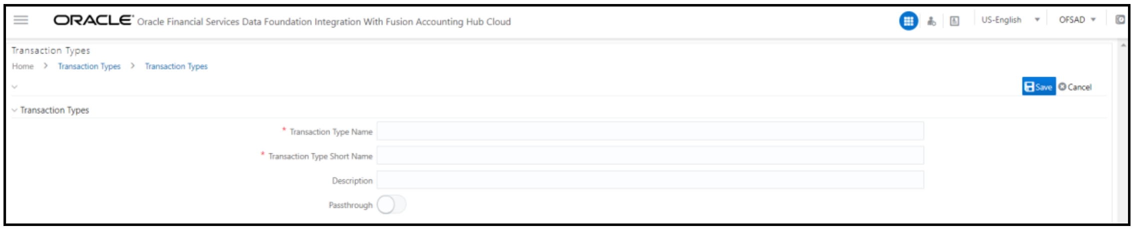
   a. Transaction Type Name

   b. Transaction Type Short Name

   c. Description

   d. Passthrough

Figure 19: Transaction Types window



ORACLE Oracle Financial Services Data Foundation Integration With Fusion Accounting Hub Cloud
US-English
OFSAD
Transaction Types
Home > Transaction Types > Transaction Types
Save
Cancel
Transaction Types
* Transaction Type Name
* Transaction Type Short Name
Description
Passthrough

The following are the validations for the fields:

Table 3: Validations

Field Name	Limitations/Validations
Transaction Type Name	• Must not have special characters. • Only alphanumeric characters and spaces are allowed. • Must not be more than 15 characters. • Must not start or end with space. • Must be unique.
Transaction Type Short Name	• Must have only Uppercase, numbers, and underscores. • Must start only with an uppercase or a number. • Must not have special characters and space. • Must not be more than 15 characters. • Must be unique.

6. Click Delete if you want to delete a transaction.
7. Click View to view the details of the transaction type.
8. Click Edit to edit the transaction type details.
9. Click **Export** to move the screen data to an Excel or CSV format for the reference.
10. During SLA configuration, the required transaction types from this list can be mapped to the defined SLA.

> **NOTE** It is not possible to edit or delete the transaction types, which are pre-packaged in the application as well as those, which are mapped to a Subledger. All transaction types lists are stored in the FSI_ACCNTING_EVENT_TYPE_MASTER table.

7.4 Subledger Attributes

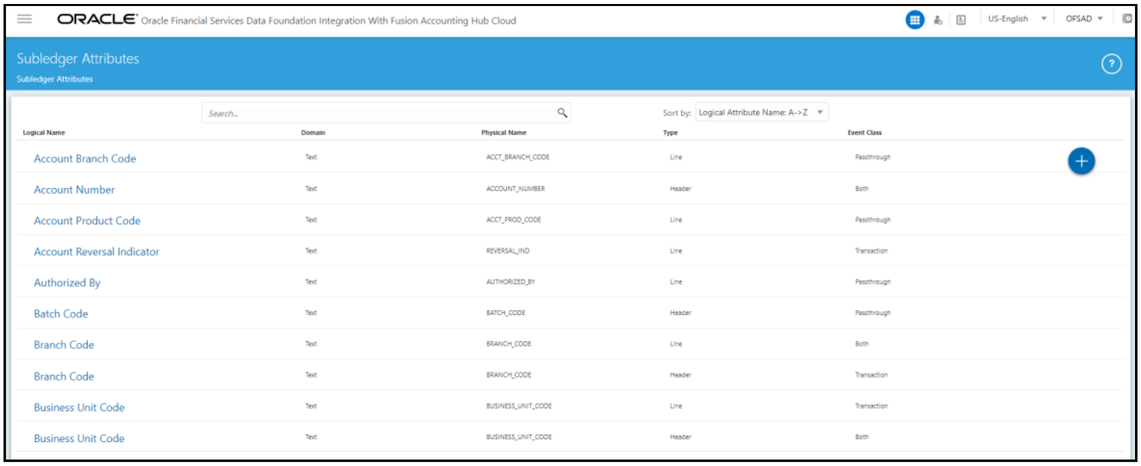
The Subledger Attributes user interface provides a list of seeded header and line attributes, which can be mapped to an SLA while defining it. These attributes cannot be edited. You can also add more attributes to the list of attributes, which can be mapped to an SLA during SLA configuration. The ledger balances can be calculated based on these Subledger attributes. To accommodate custom attributes, there is an option to add custom attributes that can vary from user-to-user to configure custom Subledger. These custom attributes can help in creating rules and extracting ledger balances in a required way.

7.4.1 Adding a Subledger Attribute

To add a Subledger Attribute, follow these steps:

1. From the **Oracle Financial Services Data Foundation Integration with Accounting Hub Cloud Service** window, select **Administration,** and then select **Subledger Attributes**. The **Subledger Attributes** window is displayed with a list of seeded data with Logical and Physical Name.

Figure 20: Subledger Attributes window

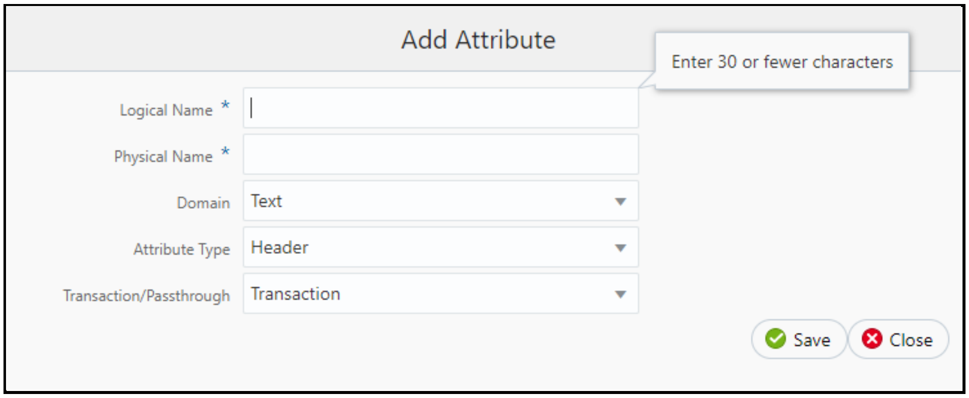


ORACLE Oracle Financial Services Data Foundation Integration With Fusion Accounting Hub Cloud
US-English
OFSAD
Subledger Attributes
Subledger Attributes
Search...
Sort by: Logical Attribute Name: A->Z

Logical Name	Domain	Physical Name	Type	Event Class
Account Branch Code	Text	ACCT_BRANCH_CODE	Line	Passthrough
Account Number	Text	ACCOUNT_NUMBER	Header	Both
Account Product Code	Text	ACCT_PROD_CODE	Line	Passthrough
Account Reversal Indicator	Text	REVERSAL_IND	Line	Transaction
Authorized By	Text	AUTHORIZED_BY	Line	Passthrough
Batch Code	Text	BATCH_CODE	Header	Passthrough
Branch Code	Text	BRANCH_CODE	Line	Both
Branch Code	Text	BRANCH_CODE	Header	Transaction
Business Unit Code	Text	BUSINESS_UNIT_CODE	Line	Transaction
Business Unit Code	Text	BUSINESS_UNIT_CODE	Header	Both

2. Click + to add a new attribute. The **Add Attribute** window is displayed.

Figure 21: Add Attribute window



Logical Name *
Physical Name *
Domain: Text
Attribute Type: Header
Transaction/Passthrough: Transaction

3. Enter the **Logical Name** and **Physical Name**. Physical Name can have only numbers and alphabets in upper case.
4. Select the **Domain**, **Attribute Type,** and **Transaction/Passthrough** details from the drop-down list. **Transaction or Passthrough** is the type of Subledger. Use this attribute for transaction type of Subledger or passthrough type of Subledger.
5. If the new attribute added is a **Header Type**, it appears in the **Transaction Information**. If the new attribute added is a **Line Type**, it appears in the **Line Information in SLA**.
6. Click **Save**.

> **NOTE** You can delete only a new attribute added but not a seeded attribute. If the added attribute is already mapped in SLA and saved, this cannot be deleted.

7.5 Subledger Applications

The goal of Subledger accounting is to generate journal entries for transactions that occur in Subledgers. To generate general ledger balances, it is very important to create journal lines and Subledger information. Subledger configuration helps transactional data to be transformed into GL account-level data by defining rules and thus loading the ledger balances. The total of the transactions in the Subledger rolls up into the general ledger.

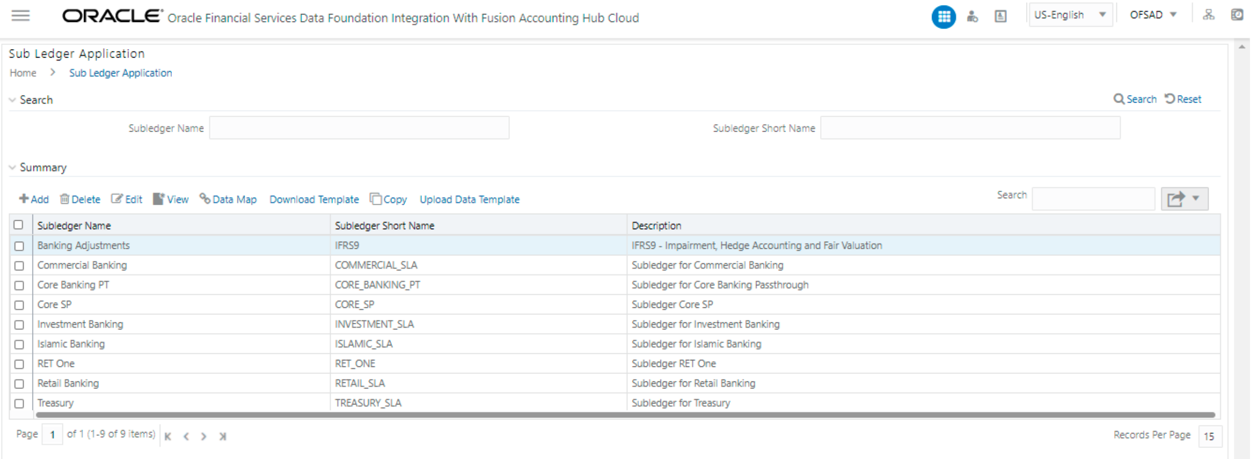
The AHC application has the SLA template through which a Subledger and its details are loaded.

7.5.1 Subledger Application Summary

To understand the Subledger Application summary window, follow these steps:

1. From the **Oracle Financial Services Data Foundation Integration with Accounting Hub Cloud Service** window, select **Administration** and then select **Subledger Applications**. The **Subledger Application** window is displayed.

Figure 22: Subledger Application Summary window



□	Subledger Name	Subledger Short Name	Description
□	Banking Adjustments	IFRS9	IFRS9 - Impairment, Hedge Accounting and Fair Valuation
□	Commercial Banking	COMMERCIAL_SLA	Subledger for Commercial Banking
□	Core Banking PT	CORE_BANKING_PT	Subledger for Core Banking Passthrough
□	Core SP	CORE_SP	Subledger Core SP
□	Investment Banking	INVESTMENT_SLA	Subledger for Investment Banking
□	Islamic Banking	ISLAMIC_SLA	Subledger for Islamic Banking
□	RET One	RET_ONE	Subledger RET One
□	Retail Banking	RETAIL_SLA	Subledger for Retail Banking
□	Treasury	TREASURY_SLA	Subledger for Treasury

2. You can search for **Subledger Name** or **Subledger Short Name**. A list of pre-packaged Subledgers appears. For more information, see the SLA Coverage file.

> **NOTE** Pre-packaged Subledgers are not editable through the AHC Subledger window.

3. In the **Subledger Application** window, in the **Summary pane**, you can **Add**, **Delete**, **Edit**, **View Data Map,** and **Download the Sub Ledger**.

4. Click + Add to create a new Subledger.

5. Select one Sub Ledger and click Delete if you wish to delete a Subledger.

6. Select one Sub Ledger and click Edit the Subledger details.

7. Select one Sub Ledger and click View to view the details of the Subledger.

8. Select one Subledger and click Download Template to download the source template with SLA information for uploading to AHCS instance

9. Select a Subledger and click Upload Data Template to upload SLA files from AHCS "Download Data Template" for said SLA.

10. Select one Sub Ledger and click Data Map to create and map connectors to the selected Subledger.

11. Select one Sub Ledger and click Copy a Subledger.
12. After the details are filled in, save the Subledger.
13. Download the SLA template that must be uploaded to the AHC application.

> **NOTE** The SLA source template from Step 8, above has to be manually validated and uploaded to your AHCS cloud environment. This template file is uploaded into your AHCS environment as a ZIP (.zip) file.
>
> If there are errors during validation, make required changes in Data Foundation Integration with Fusion Accounting Hub Cloud user interface for SLA / SLA attributes configuration and perform Step 8, above, again. See the SLA Template Validations section for more information. Subledgers saved are stored in the FSI_SLA_MASTER table.

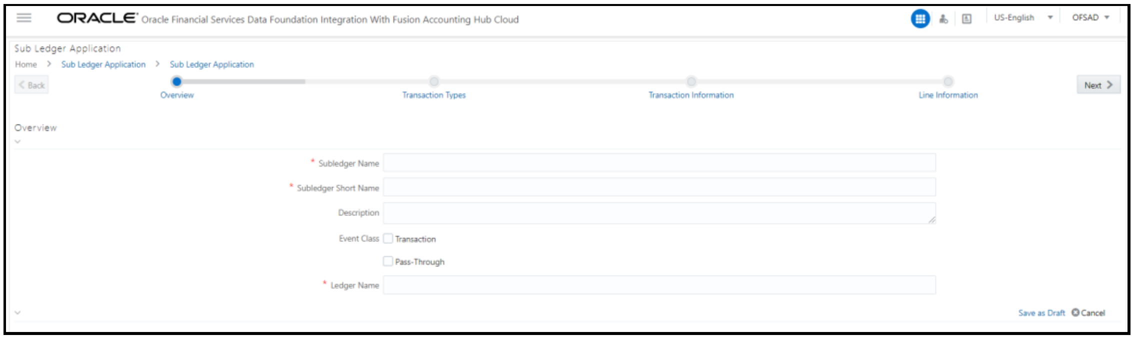
14. Click **Export** to move the summary data to an Excel or CSV format.
15. See Registration of SLAs with Accounting Hub Service, for more details on registering Sub Ledger Applications configured in this application with AHCS.

7.5.2 Adding a Subledger Application

To add a Subledger:

1. Click Add to create a new Subledger. The **Subledger Application** window is displayed.

Figure 23: Subledger Application window



2. Enter the following details:
   a. Subledger Name
   b. Subledger Short Name
   c. Description
   d. Event Class
   e. Ledger Name

   Note: The following are the validations for the fields:

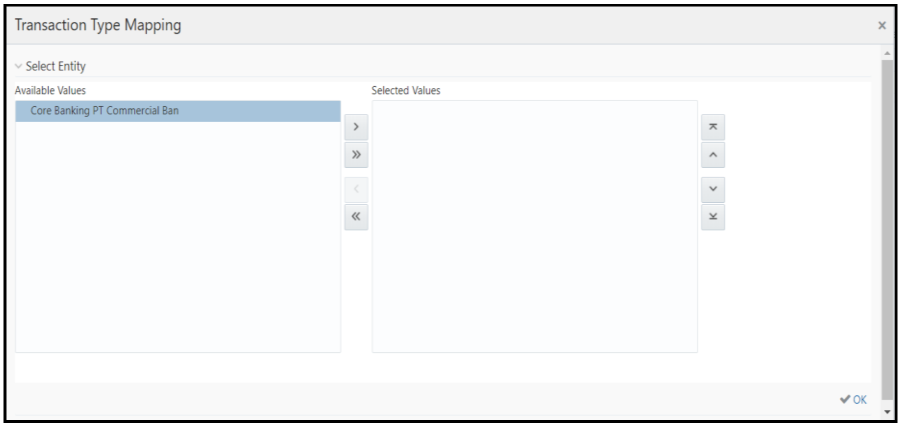
Table 4: Validations

Field Name	Limitations/Validations
Subledger Name	• Must not have special characters. • Only alphanumeric characters and spaces are allowed. • Must not have more than 15 characters. • Must not start with space. • Must be unique.
Subledger Short Name	• Must have only Uppercase, numbers, and underscores. • Must start only with an uppercase or a number. • Must not have special characters and space. • Must not have more than 15 characters. • Must be unique.
Event Class	• Can select one or both the options.
Ledger Name	• Must not exceed 100 characters. • Change the Ledger Name to the required name, for the pre-packaged Subledgers, which by default has Default Ledger as the input name.

3. Click **Save as Draft**.
   The message, "*Subledger draft saved successfully*" is displayed.
4. Click **OK**.
   The Transaction types window is displayed.
5. In the **Transaction Types** pane, perform the following steps:
   a. Click Transaction Type Mapping. All the transaction types that are created as part of Transaction Types, along with the pre-packaged list of **Transaction Types** (**Available Values**) window are displayed on the LHS.

> **NOTE** Transaction Types are filtered based on the Event Class selected for the Subledger. Transaction Type to Subledger mappings is stored in the FSI_SLA_EVENT_TYPE_MAP table.

Figure 24: Transaction Type Mapping window



b. Click › to move the required entities to the list of the **Selected Values**.

c. Click » to move all the entities to the list of the **Selected Values**.

d. Click ⤒ ˄ ˅ ⤓ to move the entities up and down.

e. After the entities are selected, click **OK**.

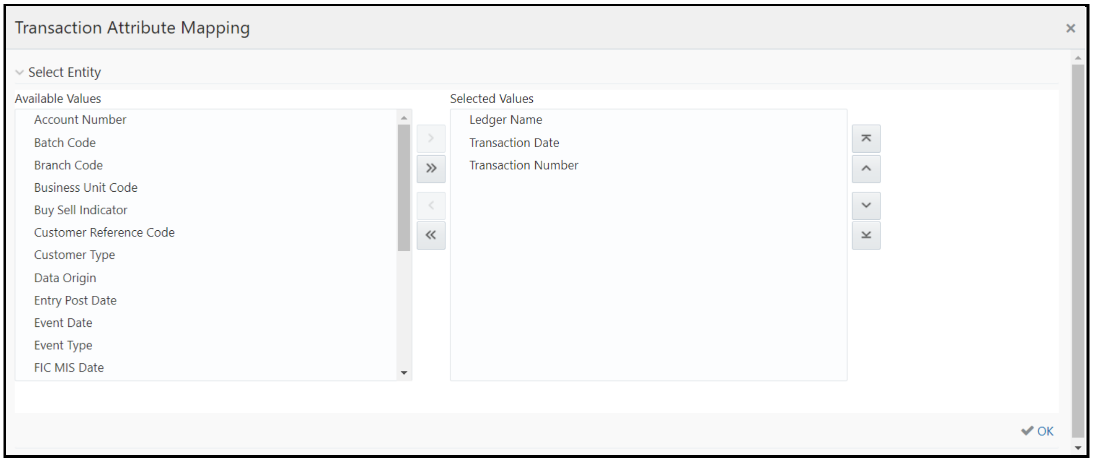
The selected values are now displayed in the **Transaction Type Mapping** window.

6. In the **Transaction Information** pane, perform the following steps:

a. Three mandatory attributes: **Ledger Name**, **Transaction Date**, and **Transaction Name** are selected by default and cannot be edited.

b. Click + Transaction Attribute Mapping. The **Transaction Attribute Mapping** window is displayed.

Figure 25: Transaction Attribute Mapping window



a. The list of transaction attributes is displayed. These are seeded from the FSI_SLA_ATTR_MASTER table.

b. Ledger Name, Transaction Date, and Transaction Name are selected by default.

c. Click › to move the required entities to the list of the **Selected Values**.

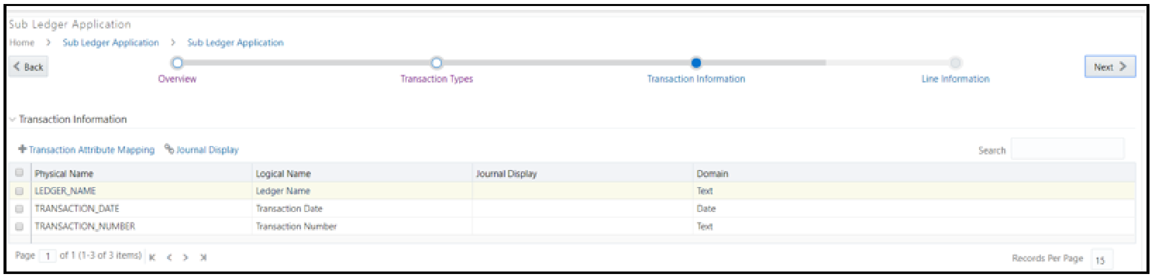
d. Click » to move all the entities to the list of the **Selected Values**.

e. Click ⤒ ˄ ˅ ⤓ to move the entities up and down.

f. After the entities are selected, click **OK**.

g. The selected values are now displayed in the **Transaction Information** pane.

Figure 26: Transaction Information window



Sub Ledger Application

Home > Sub Ledger Application > Sub Ledger Application

< Back | Overview | Transaction Types | Transaction Information | Line Information | Next >

Transaction Information

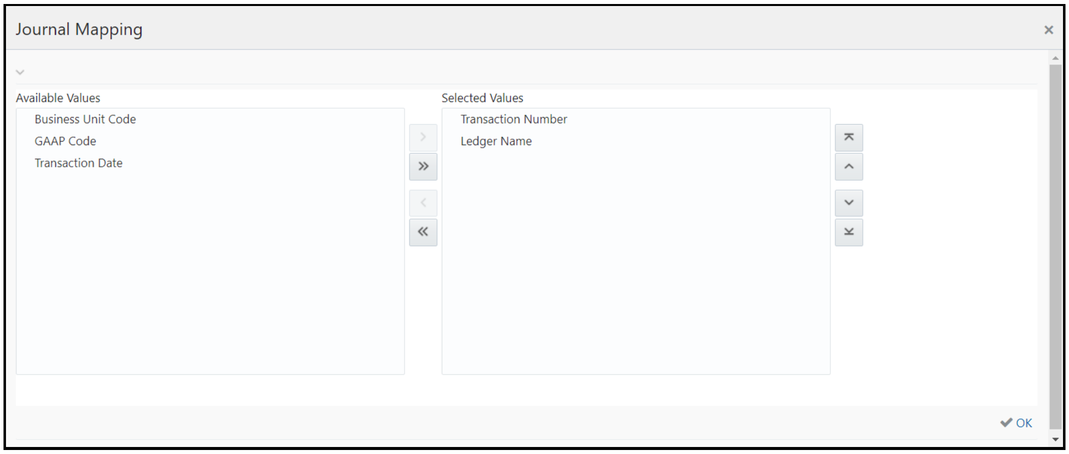
+ Transaction Attribute Mapping | Journal Display | Search

	Physical Name	Logical Name	Journal Display	Domain
□	LEDGER_NAME	Ledger Name		Text
□	TRANSACTION_DATE	Transaction Date		Date
□	TRANSACTION_NUMBER	Transaction Number		Text

Page 1 of 1 (1-3 of 3 items) | Records Per Page 15

h. If you wish to display these attributes as part of journal lines, click Journal Display . The **Journal Mapping** window is displayed.

Figure 27: Journal Mapping window



Journal Mapping

Available Values

Business Unit Code
GAAP Code
Transaction Date

Selected Values

Transaction Number
Ledger Name

OK

NOTE You cannot assign a journal display as YES to Ledger's name.

a. Click > to move the required entities to the list of the **Selected Values**.

b. Click » to move all the entities to the list of the **Selected Values**.

c. Click ⤒ ˄ ˅ ⤓ to move the entities up and down.

d. After the entities are selected, click **OK**.

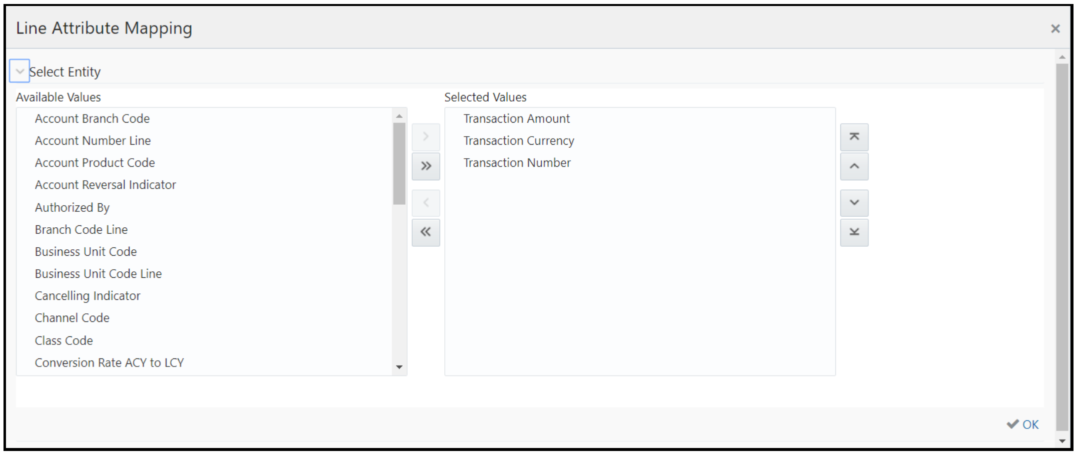
e. Click **Next**.

7. In the **Line Information** pane, perform the following steps:

   a. Three mandatory attributes: **Transaction Amount**, **Transaction Currency**, and **Transaction Number** are selected by default and cannot be edited.

   b. Click +Line Attribute Mapping . The **Line Attribute Mapping** window is displayed.

Figure 28: Line Attribute Mapping window



Line Attribute Mapping ×

Select Entity

Available Values

- Account Branch Code
- Account Number Line
- Account Product Code
- Account Reversal Indicator
- Authorized By
- Branch Code Line
- Business Unit Code
- Business Unit Code Line
- Cancelling Indicator
- Channel Code
- Class Code
- Conversion Rate ACY to LCY

Selected Values

- Transaction Amount
- Transaction Currency
- Transaction Number

✓ OK

a. The list of line attributes is displayed. These are seeded from the FSI_SLA_ATTR_MASTER table.

b. **Transaction Amount**, **Transaction Currency**, and **Transaction Number** are selected by default.

c. Click › to move the required entities to the list of the **Selected Values**.

d. Click » to move all the entities to the list of the **Selected Values**.

e. Click ⤒ ˄ ˅ ⤓ to move the entities up and down.

f. After the entities are selected, click **OK**. The selected values are now displayed in the Line Information pane.

g. If you wish to display these attributes as part of the chart of account, click Chart Of Account Value. The **Chart of Account Mapping** window is displayed.

Figure 29: Chart of the Account Mapping window

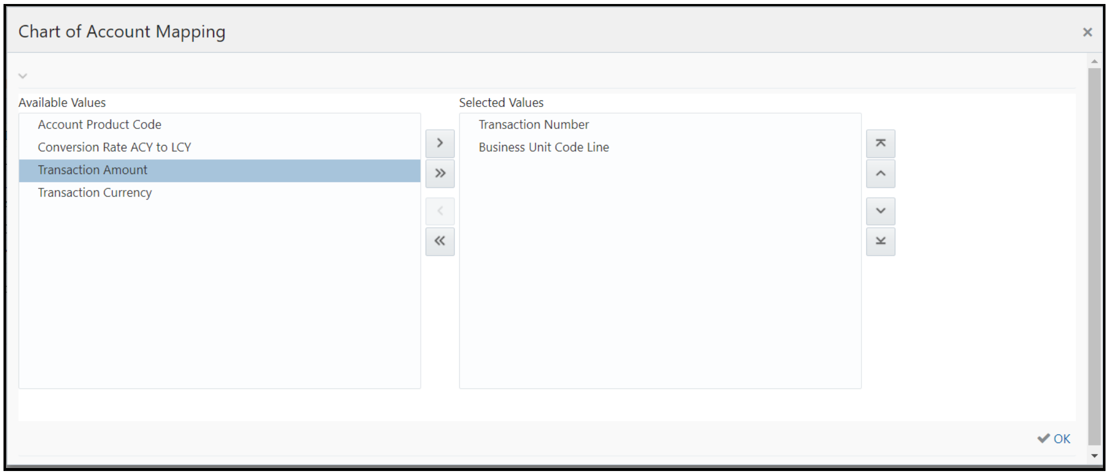


Chart of Account Mapping ×

Available Values

- Account Product Code
- Conversion Rate ACY to LCY
- Transaction Amount
- Transaction Currency

Selected Values

- Transaction Number
- Business Unit Code Line

✓ OK

NOTE You cannot assign a chart of accounts display as Yes to the three mandatory columns.

a. Click › to move the required entities to the list of the **Selected Values**.

b. Click » to move all the entities to the list of the **Selected Values**.

c. Click ⤒ ˄ ˅ ⤓ to move the entities up and down.

d. After the entities are selected, click **OK**.

8. Click **Save**. An entity is now saved in the summary window.

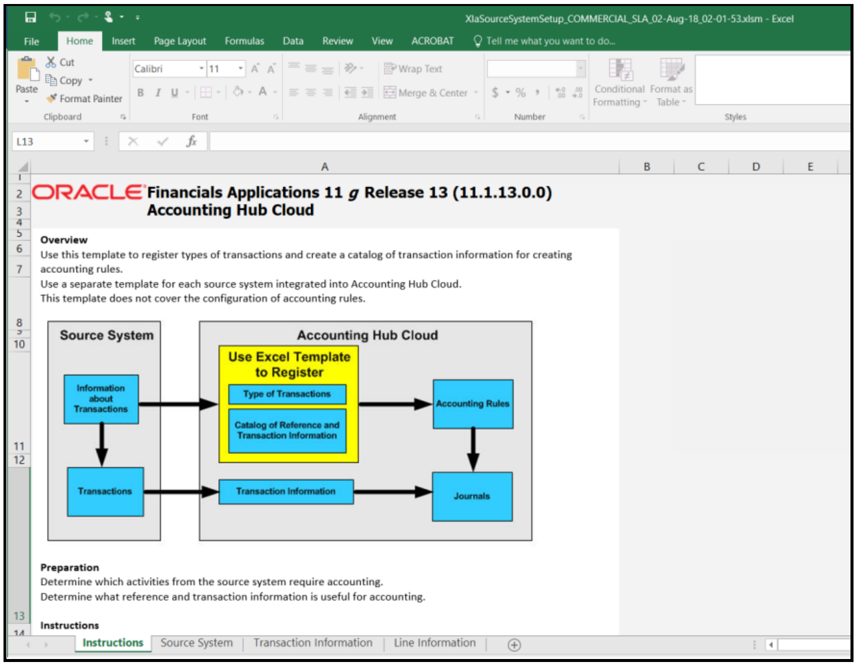
NOTE After the Transaction Information and Line, Information is mapped and saved, they are stored in the FSI_SLA_TRANSACTION_ATTR and FSI_SLA_LINE_ATTR tables, respectively. The Transaction and Line information is already configured for the pre-packaged Subledgers. See the Pre-packaged Subledger Configurations file.

7.5.3 Registration of SLAs with Accounting Hub Service

To register the SLAs with Accounting Hub Service, follow these steps:

1. From the **Oracle Financial Services Data Foundation Integration with Accounting Hub Cloud Service** window, select **Subledger Application**. The **Subledger Applications** summary window is displayed with a list of SLAs.
2. From this list of SLAs, select the SLA that must be registered.
3. Click **Download Template**. A file with the XLSM extension is downloaded to the client machine. This XLSM file maintains the template in `XlaSourceSystemSetup.xlsm` with Subledger Application details filled in.

Figure 30: XlaSourceSystemSetup.xlsm



ORACLE Financials Applications 11 *g* Release 13 (11.1.13.0.0)
Accounting Hub Cloud

Overview
Use this template to register types of transactions and create a catalog of transaction information for creating accounting rules.
Use a separate template for each source system integrated into Accounting Hub Cloud.
This template does not cover the configuration of accounting rules.

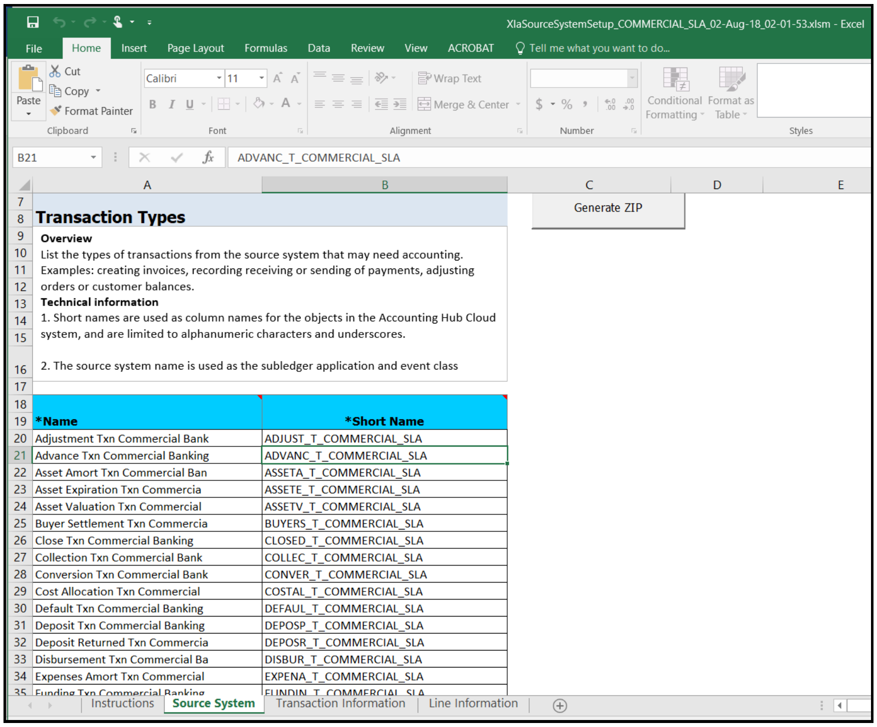
Preparation
Determine which activities from the source system require accounting.
Determine what reference and transaction information is useful for accounting.

Instructions

4. In the **Instructions** sheet, all the details are explained. Note that, the XLSM file must not be manually modified. The details that follow are to aid with users' understanding of the content of the files and how it participates in the integration process.

5. In the **Source System** sheet, the **Name** and **Short Name** are given in the Subledger Application window are displayed.

Figure 31: XlaSourceSystemSetup.xlsm – Source System



Transaction Types

Overview
List the types of transactions from the source system that may need accounting.
Examples: creating invoices, recording receiving or sending of payments, adjusting orders or customer balances.
Technical information
1. Short names are used as column names for the objects in the Accounting Hub Cloud system, and are limited to alphanumeric characters and underscores.

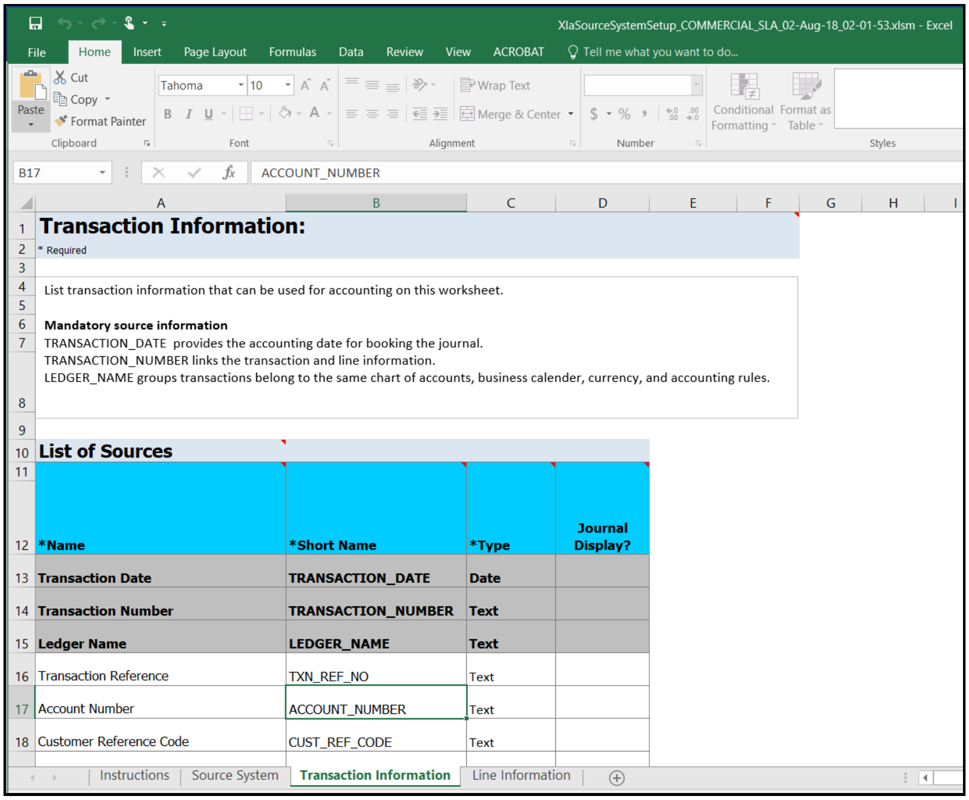
2. The source system name is used as the subledger application and event class

*Name	*Short Name
Adjustment Txn Commercial Bank	ADJUST_T_COMMERCIAL_SLA
Advance Txn Commercial Banking	ADVANC_T_COMMERCIAL_SLA
Asset Amort Txn Commercial Ban	ASSETA_T_COMMERCIAL_SLA
Asset Expiration Txn Commercia	ASSETE_T_COMMERCIAL_SLA
Asset Valuation Txn Commercial	ASSETV_T_COMMERCIAL_SLA
Buyer Settlement Txn Commercia	BUYERS_T_COMMERCIAL_SLA
Close Txn Commercial Banking	CLOSED_T_COMMERCIAL_SLA
Collection Txn Commercial Bank	COLLEC_T_COMMERCIAL_SLA
Conversion Txn Commercial Bank	CONVER_T_COMMERCIAL_SLA
Cost Allocation Txn Commercial	COSTAL_T_COMMERCIAL_SLA
Default Txn Commercial Banking	DEFAUL_T_COMMERCIAL_SLA
Deposit Txn Commercial Banking	DEPOSP_T_COMMERCIAL_SLA
Deposit Returned Txn Commercia	DEPOSR_T_COMMERCIAL_SLA
Disbursement Txn Commercial Ba	DISBUR_T_COMMERCIAL_SLA
Expenses Amort Txn Commercial	EXPENA_T_COMMERCIAL_SLA
Funding Txn Commercial Banking	FUNDIN_T_COMMERCIAL_SLA

> **NOTE** The Name or Short Name, which appears in the row, must have the event type name along with the Subledger Application name, following EVENT_TYPE_NAME_SLA_NAME pattern, as in the image, above. The Transaction Type name is appended with the SLA name to maintain the uniqueness of transaction types across all Subledger Applications. Short Name is limited by AHCS specifications to 30 characters and those employed by FSDF Integration for AHCS consider this.

6. In the **Transaction Information** sheet, those Transaction Type Names and Short Names against which Journal Display is set to Yes is displayed through the AHCS user interface.

Figure 32: XlaSourceSystemSetup.xlsm – Transaction Information



Transaction Information:

* Required

List transaction information that can be used for accounting on this worksheet.

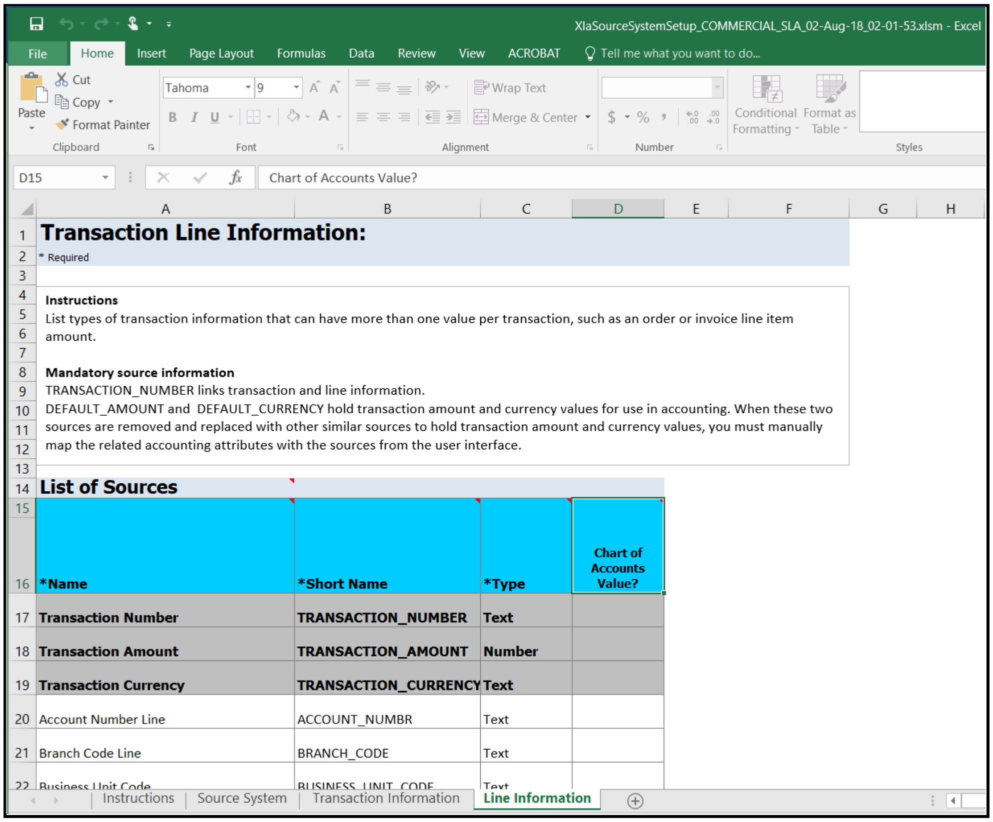
Mandatory source information
TRANSACTION_DATE provides the accounting date for booking the journal.
TRANSACTION_NUMBER links the transaction and line information.
LEDGER_NAME groups transactions belong to the same chart of accounts, business calender, currency, and accounting rules.

List of Sources

*Name	*Short Name	*Type	Journal Display?
Transaction Date	**TRANSACTION_DATE**	**Date**	
Transaction Number	**TRANSACTION_NUMBER**	**Text**	
Ledger Name	**LEDGER_NAME**	**Text**	
Transaction Reference	TXN_REF_NO	Text	
Account Number	ACCOUNT_NUMBER	Text	
Customer Reference Code	CUST_REF_CODE	Text	

7. In the **Line Information** sheet, those Transaction Type Names, and Short Names against which Journal Display is set to Yes are displayed through the AHCS user interface.

Figure 33: XlaSourceSystemSetup.xlsm – Line Information



Transaction Line Information:

* Required

Instructions
List types of transaction information that can have more than one value per transaction, such as an order or invoice line item amount.

Mandatory source information
TRANSACTION_NUMBER links transaction and line information.
DEFAULT_AMOUNT and DEFAULT_CURRENCY hold transaction amount and currency values for use in accounting. When these two sources are removed and replaced with other similar sources to hold transaction amount and currency values, you must manually map the related accounting attributes with the sources from the user interface.

List of Sources

*Name	*Short Name	*Type	Chart of Accounts Value?
Transaction Number	**TRANSACTION_NUMBER**	**Text**	
Transaction Amount	**TRANSACTION_AMOUNT**	**Number**	
Transaction Currency	**TRANSACTION_CURRENCY**	**Text**	
Account Number Line	ACCOUNT_NUMBR	Text	
Branch Code Line	BRANCH_CODE	Text	
Business Unit Code	BUSINESS_UNIT_CODE	Text	

NOTE The three mandatory columns are listed in a greyed area for both Transaction Information and Line Information.

8. Click **Validate**, in the Source System tab of the XLSM file, to check if there are any errors in the template. Note that, this step must be performed manually.

NOTE The XLSM files must not be manually modified. If you want to make changes to the files, make the required changes using the DIH user interface and follow the steps to download the template again.

9. Click **Generate ZIP**, in the Source System tab of the XLSM file, to compress required content and prepare the ZIP file. Note that, this step must be performed manually.

10. Upload the ZIP file thus generated to your instance of AHCS. See AHCS user-documentation for details on how this must be performed.

NOTE In AHCS, map Entered Amount and Entered Currency in Manage Accounting Attributes window to Transaction Amount and Transaction Currency, before proceeding with the configuration of accounting rules.

11. The following steps are required only if you are making changes to use this application with Oracle Financials Cloud 20B or later (with Flexible Configuration enabled) or when you are setting up and using this application for the first time with release 8.1.1 of the application and onwards, with Oracle Financials Cloud 20B or later.

12. Once the generated ZIP file is uploaded to your instance of AHCS, download the data template generated by AHCS corresponding to the SLA registered through the **Download Data Template** function in AHCS. See Oracle Financials documentation for details on how this must be performed.

13. Select the SLA again, as in step 2 of this process, above.

14. Click Upload Data Template .

15. Drag the file obtained from your instance of AHCS in step 11 above into the window. Alternatively, you can click **Browse** and select the file.

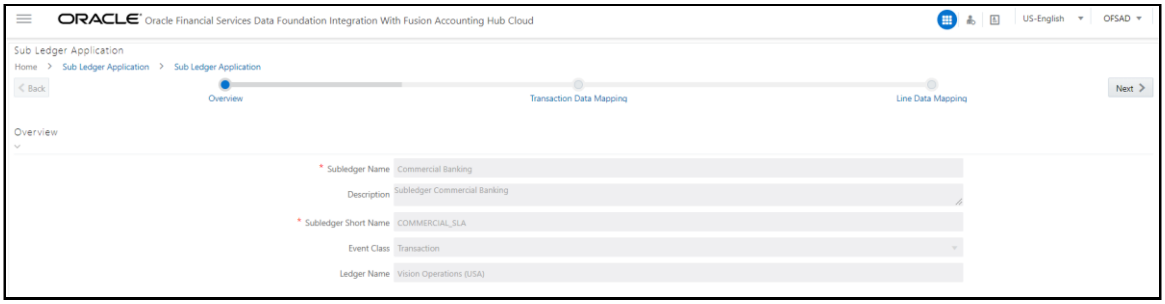
16. After successful completion, the SLA definition is now complete.

7.5.4 Using Data Map

To create a Connector and map it to a Subledger Application, follow these steps:

1. From the **Subledger Application** summary window, select the desired SLA, and click **Data Map**. The fields are non-editable in the Overview pane.

Figure 34: Subledger Application Window

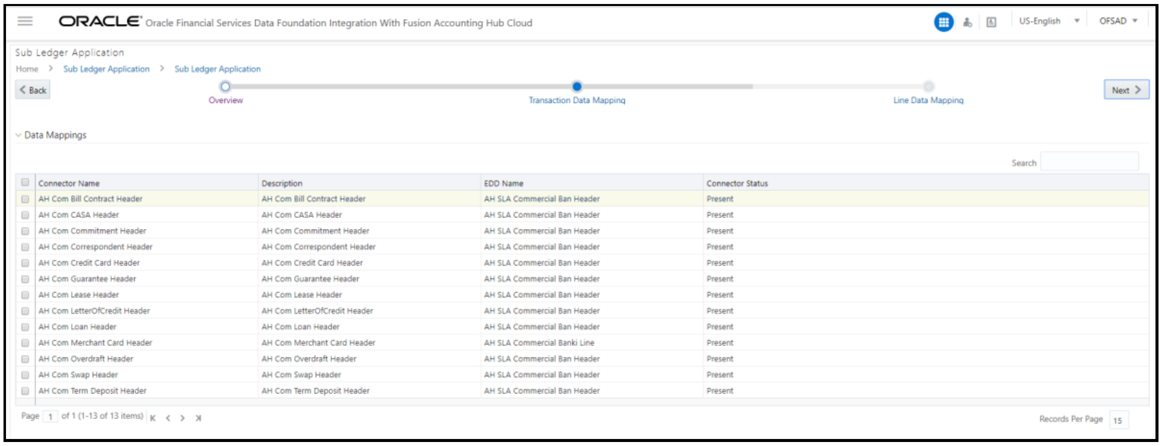

ORACLE Oracle Financial Services Data Foundation Integration With Fusion Accounting Hub Cloud
US-English
OFSAD
Sub Ledger Application
Home > Sub Ledger Application > Sub Ledger Application
Back
Overview
Transaction Data Mapping
Line Data Mapping
Next

Overview

* Subledger Name	Commercial Banking
Description	Subledger Commercial Banking
* Subledger Short Name	COMMERCIAL_SLA
Event Class	Transaction
Ledger Name	Vision Operations (USA)

2. Click **Next**. The **Transaction Data Mapping** window is displayed.

Figure 35: Transaction Data Mapping Window


ORACLE Oracle Financial Services Data Foundation Integration With Fusion Accounting Hub Cloud
US-English
OFSAD
Sub Ledger Application
Home > Sub Ledger Application > Sub Ledger Application
Back
Overview
Transaction Data Mapping
Line Data Mapping
Next
Data Mappings
Search

□	Connector Name	Description	EDD Name	Connector Status
□	AH Com Bill Contract Header	AH Com Bill Contract Header	AH SLA Commercial Ban Header	Present
□	AH Com CASA Header	AH Com CASA Header	AH SLA Commercial Ban Header	Present
□	AH Com Commitment Header	AH Com Commitment Header	AH SLA Commercial Ban Header	Present
□	AH Com Correspondent Header	AH Com Correspondent Header	AH SLA Commercial Ban Header	Present
□	AH Com Credit Card Header	AH Com Credit Card Header	AH SLA Commercial Ban Header	Present
□	AH Com Guarantee Header	AH Com Guarantee Header	AH SLA Commercial Ban Header	Present
□	AH Com Lease Header	AH Com Lease Header	AH SLA Commercial Ban Header	Present
□	AH Com LetterOfCredit Header	AH Com LetterOfCredit Header	AH SLA Commercial Ban Header	Present
□	AH Com Loan Header	AH Com Loan Header	AH SLA Commercial Ban Header	Present
□	AH Com Merchant Card Header	AH Com Merchant Card Header	AH SLA Commercial Banki Line	Present
□	AH Com Overdraft Header	AH Com Overdraft Header	AH SLA Commercial Ban Header	Present
□	AH Com Swap Header	AH Com Swap Header	AH SLA Commercial Ban Header	Present
□	AH Com Term Deposit Header	AH Com Term Deposit Header	AH SLA Commercial Ban Header	Present

Page 1 of 1 (1-13 of 13 items)   Records Per Page 15

a. Click **Create** to add a new data mapping. The **New Connector** window is displayed.

b. In the **Target** pane, the required EDD is automatically created following the **AHC SLA <<SLA NAME>> Header** pattern. For example, when Subledger Application is named Commercial Banking, EDD created is **AH SLA Commercial Banking Header**. EDD structure will have all the attributes that are mapped to the Subledger Application under Transaction Information.

c. Map the required source and target and save the connector.

> **NOTE** You must carefully apply filters if any while creating connectors to fetch the correct data from a product for a Subledger. Extraction Date filter is already enabled with pre-packaged connectors for executing them.

d. Event Type names must also be filtered depending on the Subledger you configure. If not configured, all transaction types are processed in the extract data for a Subledger and the accounting process fails.

e. Double-click **Mapping** to map the source and target.

> **NOTE** For more information on mapping, see the *Connectors* section in the DIH User Guide.

f. Click **Properties** or >.
The **Connector Details** window is displayed.

g. Enter the **Name** and **Description** and click **Save**, or **Save As Draft**.

h. Click **Transaction Data Mapping**.
The **Transaction Data Mapping** is displayed with all the header-related connectors, which are saved under AHC. You can add the required connector for the Subledger listed here.

i. Click > to move the required entities to the list of the **Selected Values**.

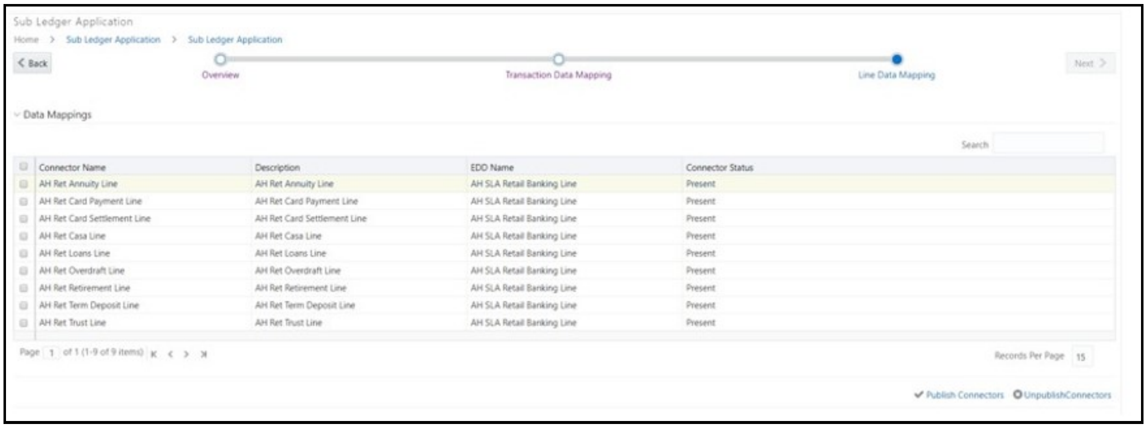
ii. Click » to move all the entities to the list of the **Selected Values**.

iii. You click ⤒ ˄ ˅ ⤓ to move the entities up and down.

iv. After the entities are selected, click **OK**.

3. Click **Next**.
The **Line Data Mapping** window is displayed.

Figure 36: Line Data Mapping Window



□	Connector Name	Description	EDD Name	Connector Status
□	AH Ret Annuity Line	AH Ret Annuity Line	AH SLA Retail Banking Line	Present
□	AH Ret Card Payment Line	AH Ret Card Payment Line	AH SLA Retail Banking Line	Present
□	AH Ret Card Settlement Line	AH Ret Card Settlement Line	AH SLA Retail Banking Line	Present
□	AH Ret Casa Line	AH Ret Casa Line	AH SLA Retail Banking Line	Present
□	AH Ret Loans Line	AH Ret Loans Line	AH SLA Retail Banking Line	Present
□	AH Ret Overdraft Line	AH Ret Overdraft Line	AH SLA Retail Banking Line	Present
□	AH Ret Retirement Line	AH Ret Retirement Line	AH SLA Retail Banking Line	Present
□	AH Ret Term Deposit Line	AH Ret Term Deposit Line	AH SLA Retail Banking Line	Present
□	AH Ret Trust Line	AH Ret Trust Line	AH SLA Retail Banking Line	Present

a. Click **Create** to add a new data mapping.
The **New Connector** window is displayed.

In the **Target** pane, the EDD is automatically created. For example, Subledger's name is Commercial Banking. The EDD is created with AHC SLA as the prefix of the name and Line as the suffix. The full name is **AH SLA Commercial Banking Line**. EDD structure has all the attributes that are mapped to the Subledger under the Line Information.

b. Map the required source and target and save the connector.

c. Double-click **Mapping** to map the source and target.

> **NOTE** You must apply filters if any while creating connectors to fetch the correct data from a product for a Subledger. Extraction Date filter is already enabled with pre-packaged connectors for executing them.
>
> For more information on mapping, see the *Connectors* section in the DIH User Guide.

d. Click **Properties** or >. The **Connector Details** window is displayed.

e. Enter the **Name** and **Description** and click Save, or **Save As Draft**.

f. Click **Line Data Mapping**. The **Line Data Mapping** window is displayed. It displays all the line-related connectors, which are saved under AHC. You can add the required connector for the Subledger listed here.

   i. Click > to move the required entities to the list of the **Selected Values**.

   ii. Click » to move all the entities to the list of the **Selected Values**.

   iii. You click ⤒ ^ ˅ ⤓ to move the entities up and down.

   iv. After the entities are selected, click **OK**.

4. Click **Save**.
   The message, "*SLA and Connector saved successfully*" is displayed.

5. Click ✓ Publish Connectors to publish all the Transaction and Line EDD-related connectors.
   After publishing successfully, a batch is created automatically, and the batch name is displayed in a message (<INFODOM>_DIH_AH_<SLA_CODE>).

6. If you want to unpublish the connector, click ⊗ UnpublishConnectors .
   After unpublishing the connectors, the SLA batch created for published SLA is removed.

7. After unpublishing a connector, select the required connector and click ✎ Edit Data Mapping to edit a connector.

7.5.5 Copying a Subledger Application

To copy a Subledger, follow these steps:

1. From the **Subledger Application** window, select one Subledger and click Copy.

   The **Save Subledger** window is displayed.

2. Enter the **Name** and **Short Name** and click **Save**. This copies all the properties except the Name and Short Name as it must be unique.

   For example,

Table 5: Connector Details

Connector Name Before Copy	Connector Name After Copy
AH Com Bill Contract Header AH Com Casa Header	AH (copied SLA name) succeeded by Hdr1, Hdr2, and Hdr3
AH Com Commitment Header	

3. Similarly, for line mapping, the name after copy shows as AH <SLAname> Ln1.
4. The EDD name is also replaced with the copied name in the connector window in the format AH SLA <SLA name> Header and AH SLA <SLA name> Line.

7.6 Standardization of Product Codes and Mapping with Subledger Applications

7.6.1 Filtering Data Based on Product Codes

Data for each Subledger Application is obtained from common Header tables and specific transaction tables corresponding to the types of products relevant to the SLA. Some of these transaction tables or Product Processor Entities contain data relevant to more than one SLA.

In such instances, appropriate filters must be applied as data is extracted from the Product Processor Entities for purposes of each SLA. Subledger Applications packaged with FSDF Integration with AHCS already contain such filters, referring to standard product types and codes thereof. See the seeded product codes file for details on product processor entities and seeded product codes.

Product Processor Entities that contain data relevant to more than a single SLA are listed along with corresponding SLAs as follows.

Table 6: Product Processor Entities

Product Processor Entity	Subledger Applications		
STG_CASA	Retail Banking	Commercial Banking	
STG_LOAN_CONTRACTS	Retail Banking	Commercial Banking	
STG_CARDS	Retail Banking	Commercial Banking	
STG_TD_CONTRACTS	Retail Banking	Commercial Banking	
STG_OD_ACCOUNTS	Retail Banking	Commercial Banking	
STG_SWAPS_CONTRACTS	Investment Banking	Commercial Banking	Treasury

Product Processor Entity	Subledger Applications		
STG_MM_CONTRACTS	Investment Banking	Treasury	
STG_INVESTMENTS	Investment Banking	Treasury	
STG_FUTURES	Investment Banking	Treasury	
STG_FX_CONTRACTS	Investment Banking	Treasury	
STG_CREDIT_DERIVATIVES	Investment Banking	Treasury	
STG_OPTION_CONTRACTS	Investment Banking	Treasury	
STG_REPO_CONTRACTS	Investment Banking	Treasury	
STG_COMMITMENT_CONTRACTS	Commercial Banking	Treasury	
STG_CORRESPONDENT_ACCOUNT	Commercial Banking	Treasury	

A set of product codes must be established in Managing Product Code Filters such that multiple product or instrument codes in use across your institution are standardized for use across analysis and accounting. FSDF Integration with AHCS provides a seeded list of product codes, each of which is mapped to a Subledger Application in Managing Product Code Filters.

For example, STG_CASA is designed to host data relating to both commercial and retail current and savings accounts, through seeded product codes RETACC and CORPACC. These product codes are in turn, mapped to Retail Banking and Commercial Banking SLAs. Packaged SLAs have filters on product code predefined.

7.6.2 Managing Product Code Filters

FSDF Integration with AHCS uses a combination of filter clauses explicitly defined in Connectors and additional information maintained in SETUP_MASTER, a configuration entity. While defining SLAs and related information, you can use entries recorded in this entity to facilitate filtered extraction of data from product processor entities to SLAs. You can also use this mechanism to affect such filtered extraction, if packaged product codes, SLA definitions, related information, or a combination of these are modified.

The following tabulation depicts sample entries in SETUP_MASTER to address the following scenario:

- STG_CASA has data with four product codes – Retail Product 1, Retail Product 2, Commercial Product 1, and Commercial Product 2.
- Data corresponding to Retail Product 1 and Retail Product 2 must be mapped to Retail Banking SLA.
- Data corresponding to Commercial Product 1 and Commercial Product 2 must be mapped to Commercial Banking SLA.
- Standardized product codes are RETACC and CORPACC, corresponding to Retail Banking and Commercial Banking SLAs, respectively.

Entries in the following table facilitate the appropriate filtering of data from the STG_CASA product processor entity to the correct SLAs.

Table 7: Sample Entries in SETUP_MASTER

V_COMPONENT_CODE (Actual Product Code)	V_COMPONENT_VALUE (Seeded Product Code)	V_COMPONENT_DESC (Description)
Retail Product 1	RETACC	STG_CASA to Retail Banking
Retail Product 2	RETACC	STG_CASA to Retail Banking
Commercial Product 1	CORPACC	STG_CASA to Commercial Banking
Commercial Product 2	CORPACC	STG_CASA to Commercial Banking

7.7 OFSAA Chart of Account (COA) Mapping

COA segments are predefined in the cloud environment before loading data to the STG_GL_DATA and management ledger table through connectors. You must map the relevant COA segments to the FSDF columns for loading data, as defined in the Accounting Hub Cloud Service environment. For example, if COA segment 1 is mapped to the Legal Entity attribute in a cloud environment while defining account rules, the same must be mapped in the DIH screens Legal Entity to COA_SEGMENT1. The mapping window for the COA segment displays different dimensions to allow you to map them to respective COA segments.

The ledger balance data from AHC appear in the form of COA segment columns. COA segments are defined differently for different users and therefore, there is an option to select which dimension represents which segment.

The Subledger COA Mapping window displays the seeded dimension names, which are a part of the data model, and it is possible to select dimensions against each COA segment as required.

The list of predefined dimensions required in the application and which are mapped to COA segments are as follows:

- General Ledger Account Code
- Legal Entity Code
- Account Branch Code
- GAAP Code
- Currency Code
- Product Code
- Organization Unit Code
- Business Unit Code
- Customer Class Code

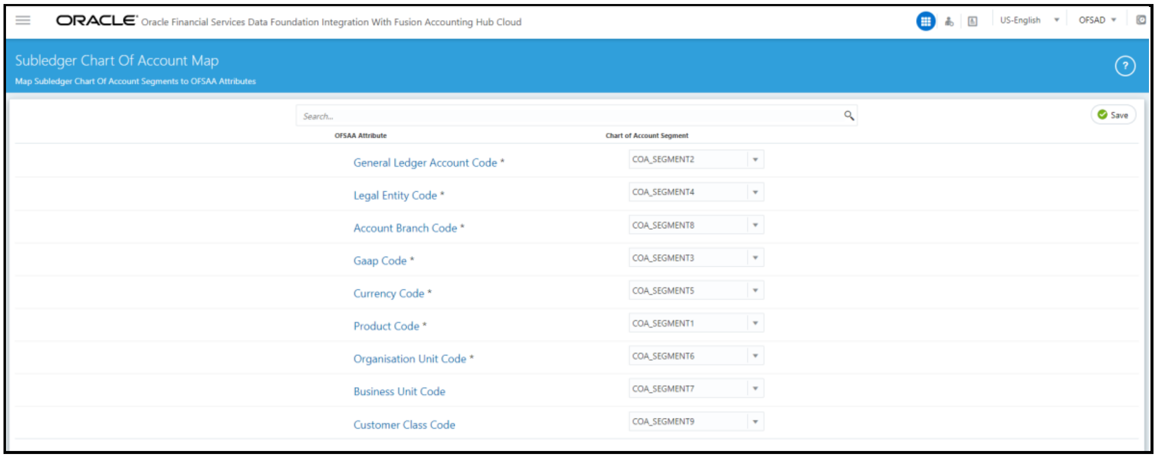
7.7.1 Mapping Segments

Depending on the COA or accounting principles followed, the OFSAA predefined dimensions can be mapped with the list of COA segments.

To map the segments, follow these steps:

1. From the **Oracle Financial Services Data Foundation Integration with Accounting Hub Cloud Service** window, select **AHC Administration** and then select **OFSAA Chart of Account Mapping**. The **Subledger Chart of Account Map** window is displayed.

Figure 37: Subledger Chart of Account Map Window


ORACLE Oracle Financial Services Data Foundation Integration With Fusion Accounting Hub Cloud
US-English
OFSAD
Search...
Save

Subledger Chart Of Account Map

Map Subledger Chart Of Account Segments to OFSAA Attributes

OFSAA Attribute	Chart of Account Segment
General Ledger Account Code *	COA_SEGMENT2
Legal Entity Code *	COA_SEGMENT4
Account Branch Code *	COA_SEGMENT8
Gaap Code *	COA_SEGMENT3
Currency Code *	COA_SEGMENT5
Product Code *	COA_SEGMENT1
Organisation Unit Code *	COA_SEGMENT6
Business Unit Code	COA_SEGMENT7
Customer Class Code	COA_SEGMENT9

The **OFSAA Attribute** column displays the list of predefined dimensions and the **Chart Of Account Segment** displays the COA Segment value with a drop-down list of COA segments COA_SEGMENT1, COA_SEGMENT2, and so on.

2. Select the **NONE** option when the dimension is not required to be mapped or used.
3. Select the required COA segments and click **Save**.
4. These segment values are replaced with the dimension columns as mapped in the inbound connector and the data is loaded as per the mapping into the staging tables of the ledger loaded as mapped into Data Foundation.
5. After it is saved, the Insert Connectors automatically get mapped for both GL and SR.

NOTE

- Specify the COA segment as the attributes in OFSAA. For example, if you map the COA Segment 1 to GL Code, all the extracts from AHCS with column header as COA Segment 1 are loaded as GL Code in OFSAA.
- The predefined dimensions with * must be mapped to some of the other COA segments to get values, as they are not null columns.

7.8 Subledger Extract Files

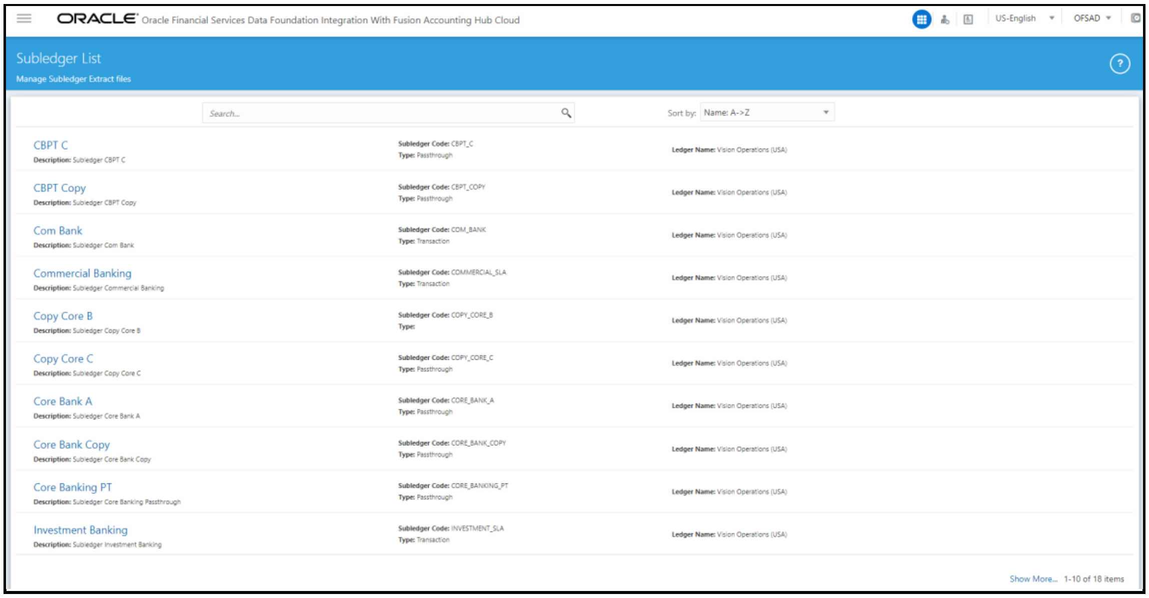
During batch execution of Subledger connectors, by default, the sixth parameter of the last task is set to N (for more information, see Batch Execution which generates a .zip file) after successful execution. When the parameter is set to Y, execution generates a .zip file, uploads it to UCM, and triggers the required process automatically.

In both cases, the **Subledger List** window can be used to manage the extracted zip files.

The **Subledger Extract Files** window has all the list of Subledgers, where you can select individual Subledger and upload required files to UCM or further process them if the files are already uploaded.

From the **Oracle Financial Services Data Foundation Integration with Accounting Hub Cloud Service** window, select **AHC Administration** and then select **Subledger List**. The **Subledger List** window is displayed.

Figure 38: Subledger List Window



ORACLE Oracle Financial Services Data Foundation Integration With Fusion Accounting Hub Cloud

Subledger List
Manage Subledger Extract files

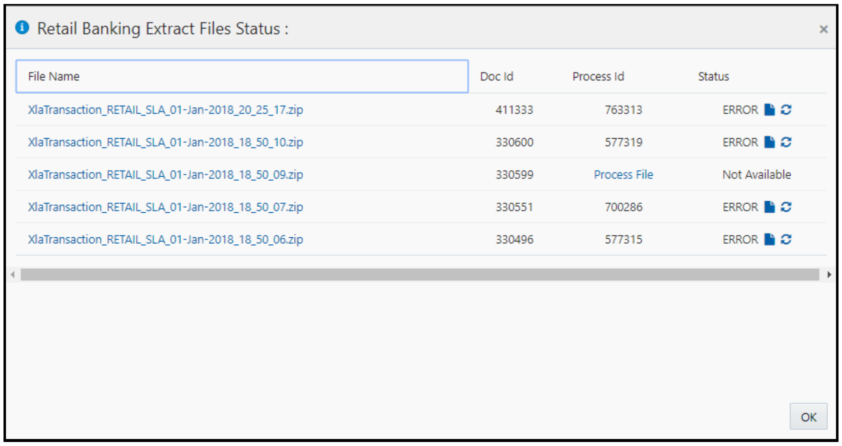
Sort by: Name: A->Z

CBPT C **Description:** Subledger CBPT C	**Subledger Code:** CBPT_C **Type:** Passthrough	**Ledger Name:** Vision Operations (USA)
CBPT Copy **Description:** Subledger CBPT Copy	**Subledger Code:** CBPT_COPY **Type:** Passthrough	**Ledger Name:** Vision Operations (USA)
Com Bank **Description:** Subledger Com Bank	**Subledger Code:** COM_BANK **Type:** Transaction	**Ledger Name:** Vision Operations (USA)
Commercial Banking **Description:** Subledger Commercial Banking	**Subledger Code:** COMMERCIAL_SLA **Type:** Transaction	**Ledger Name:** Vision Operations (USA)
Copy Core B **Description:** Subledger Copy Core B	**Subledger Code:** COPY_CORE_B **Type:**	**Ledger Name:** Vision Operations (USA)
Copy Core C **Description:** Subledger Copy Core C	**Subledger Code:** COPY_CORE_C **Type:** Passthrough	**Ledger Name:** Vision Operations (USA)
Core Bank A **Description:** Subledger Core Bank A	**Subledger Code:** CORE_BANK_A **Type:** Passthrough	**Ledger Name:** Vision Operations (USA)
Core Bank Copy **Description:** Subledger Core Bank Copy	**Subledger Code:** CORE_BANK_COPY **Type:** Passthrough	**Ledger Name:** Vision Operations (USA)
Core Banking PT **Description:** Subledger Core Banking Passthrough	**Subledger Code:** CORE_BANKING_PT **Type:** Passthrough	**Ledger Name:** Vision Operations (USA)
Investment Banking **Description:** Subledger Investment Banking	**Subledger Code:** INVESTMENT_SLA **Type:** Transaction	**Ledger Name:** Vision Operations (USA)

Show More... 1-10 of 18 items

When an SLA is selected, a new window opens which contains the list of files uploaded for that SLA.

Figure 39: Subledger Extract Files window



Retail Banking Extract Files Status :

File Name	Doc Id	Process Id	Status
XlaTransaction_RETAIL_SLA_01-Jan-2018_20_25_17.zip	411333	763313	ERROR
XlaTransaction_RETAIL_SLA_01-Jan-2018_18_50_10.zip	330600	577319	ERROR
XlaTransaction_RETAIL_SLA_01-Jan-2018_18_50_09.zip	330599	Process File	Not Available
XlaTransaction_RETAIL_SLA_01-Jan-2018_18_50_07.zip	330551	700286	ERROR
XlaTransaction_RETAIL_SLA_01-Jan-2018_18_50_06.zip	330496	577315	ERROR

It has four columns:

- **File Name**: The files uploaded are displayed under the **File Name** column. Click the file name to download the file to a local system.
- **Doc ID**: For the extracted .zip files, where batch execution is completed with the sixth parameter as N has **Upload to UCM** option in this field for the first time. You can click this, and the application uploads the created .zip file to UCM. After it is uploaded, a unique Doc ID number is created, and it is displayed in this column. For files executed with parameter 6 as Y and parameter 7 as 1, a unique Doc ID number is displayed.
- **Process ID**: For the extracted zip files, where batch execution is completed with the 6th parameter as Y and the 7th parameter as 2, it has a unique Process ID number in this column. The files which are executed with the **sixth parameter as N or Y** (along with the 7th parameter as 1), has the Process File option in this field is displayed for the first time.

  The Process File either uploads the .zip file to UCM and triggers the process or job as ‘Import Accounting Transactions’.

or

It directly triggers the uploaded file with the Import Accounting Transactions process.

Click **Process File** and the following actions occur:

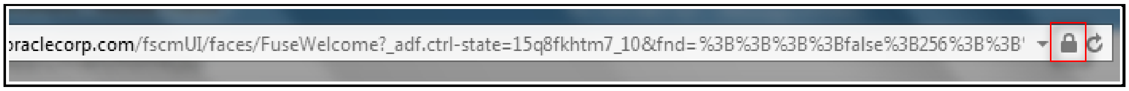
- If Doc ID number exists: .zip file, which is already uploaded to UCM under the existing Doc ID, is triggered for processing and a unique Process ID is created, which appears under this column after a successful trigger.
- If Doc ID says Upload to UCM: .zip file that is generated is automatically uploaded to UCM and triggers the required process. This updates the Doc ID and Process ID with a unique number after successful upload and process, respectively.

- **Status**: The status is updated in a couple of seconds, which reflects the status of the 'Import Accounting Transactions' job.

7.9 Import Certificate into Keystore

Export the certificate from the browser to the file using the following steps:

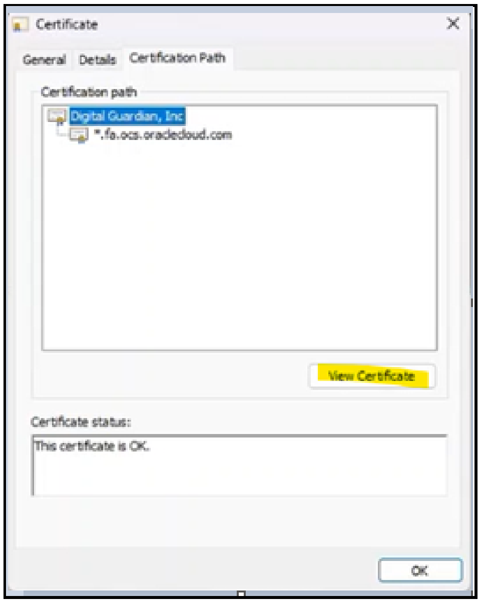
1. Access the AHC SSL URL from any web service using Microsoft Internet Explorer or Mozilla Firefox
2. Click the **Security Lock** symbol and then click **View Certificates**.



›raclecorp.com/fscmUI/faces/FuseWelcome?_adf.ctrl-state=15q8fkhtm7_10&fnd=%3B%3B%3B%3Bfalse%3B256%3B%3B'

3. From the **Certification Path** tab, the certificate hierarchy is displayed. Export the certificate (**Digital Guardian, Inc** in the following case) to KEYSTORE file.

Figure 40: Certificate window



Digital Guardian, Inc
*.fa.ocs.oraclecloud.com

This certificate is OK.

4. From the **Certification Path** tab, select **Digital Guardian, Inc** and then click **View Certificate**.
5. From the **Details** tab, select **Copy to File**. The **Certificate Export Wizard** is displayed.
6. Click **Next** and enter a name and location for the file you want to export.
7. Change the encoding to **Base-64** and click **Next**.

8. Enter the file name and click **Finish**.

9. When using other web browsers, perform similar steps. The navigation path can differ in the web browsers used.

10. Run the following command to import a certificate into KeyStore:

```
keytool -import -trustcacerts -file <filename> -alias <aliasname> -
keystore keystore.jks -storepass welcome1
```

11. Run the following command to verify if the trust store contains the imported certificates:

```
keytool -list -v -Keystore <filename> -storepass welcome1 | grep -i Verizon
```

7.10 Subledger Event Grouping

Subledger Event Grouping functionality is introduced to the group data, based on a few seeded dimensions while extracting and then posting them to AHC. Grouping is based on dimensions and significance. Similarly, while loading the data from AHC to STG_GL_DATA, aggregation based on the dimensions and the basic primary key columns of the GL data table helps in maintaining the uniqueness of the table without any errors.

Grouping functionality includes combining data based on some preseeded columns in both header and line-level data. When data is grouped on some dimensions and transaction numbers are not considered for grouping, there is a requirement to regenerate the transaction numbers. As aggregate columns are different in header and line level, aggregation happens separately for header and line data. However, the only connection between the line and header data is the transaction number. Therefore, to maintain the connection, pair up the corresponding header and line connectors along with a mapping table, which maintains actual transaction numbers with a map to newly generated or replaced transaction numbers.

Grouping functionality includes the following components:

- Event Group Summary
- Defining an Event Group
- Managing Group Attributes

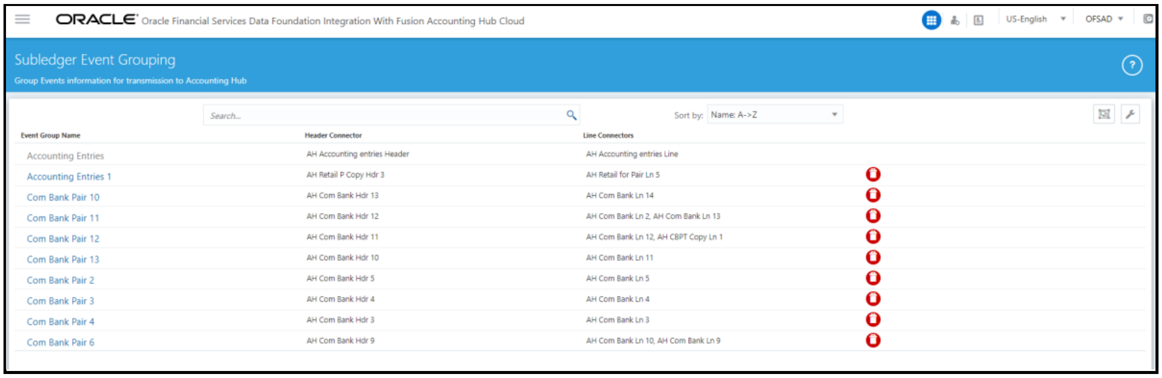
7.10.1 Subledger Event Grouping Summary

To understand the Sub Ledger Event Grouping summary window, follow these steps:

1. From the **Oracle Financial Services Data Foundation Integration with Accounting Hub Cloud Service** window, select **AHC Administration** and then select **Subledger Event Grouping**.

   The **Subledger Event Grouping** window is displayed with a list of pre-packaged event groups information for transmission to the Accounting Hub.

Figure 41: Subledger Event Grouping


ORACLE Oracle Financial Services Data Foundation Integration With Fusion Accounting Hub Cloud
US-English
OFSAD
Subledger Event Grouping
Group Events information for transmission to Accounting Hub
Search...
Sort by: Name: A->Z

Event Group Name	Header Connector	Line Connectors
Accounting Entries	AH Accounting entries Header	AH Accounting entries Line
Accounting Entries 1	AH Retail P Copy Hdr 3	AH Retail for Pair Ln 5
Com Bank Pair 10	AH Com Bank Hdr 13	AH Com Bank Ln 14
Com Bank Pair 11	AH Com Bank Hdr 12	AH Com Bank Ln 2, AH Com Bank Ln 13
Com Bank Pair 12	AH Com Bank Hdr 11	AH Com Bank Ln 12, AH CBPT Copy Ln 1
Com Bank Pair 13	AH Com Bank Hdr 10	AH Com Bank Ln 11
Com Bank Pair 2	AH Com Bank Hdr 5	AH Com Bank Ln 5
Com Bank Pair 3	AH Com Bank Hdr 4	AH Com Bank Ln 4
Com Bank Pair 4	AH Com Bank Hdr 3	AH Com Bank Ln 3
Com Bank Pair 6	AH Com Bank Hdr 9	AH Com Bank Ln 10, AH Com Bank Ln 9

2. The **Subledger Event Grouping** window displays the **Event Group Name**, **Header Connector**, and **Line Connectors**.

   a. **Event Group Name**: It is the generic name given to identify a Header and Line Connector mapping. The pre-packaged Subledger connectors and names are displayed.

   b. **Header Connector**: Displays the Header Connector's name.

   c. **Line Connectors**: Displays the Line Connector's name.

3. You can search for Sub Ledger Event Grouping Name.

4. You can define an Event Group, Manage Group Attributes, and Delete user-defined group events.

5. Click 🗑 if you want to delete a Sub Ledger Event Group.

> **NOTE** It is not possible to delete a pre-packaged Sub Ledger Event Group.

7.10.2 Defining an Event Group

To define an event group, follow these steps:

1. From the **Subledger Event Grouping** window, click ⧉ to define the event group. The **Define Event Group** window is displayed.

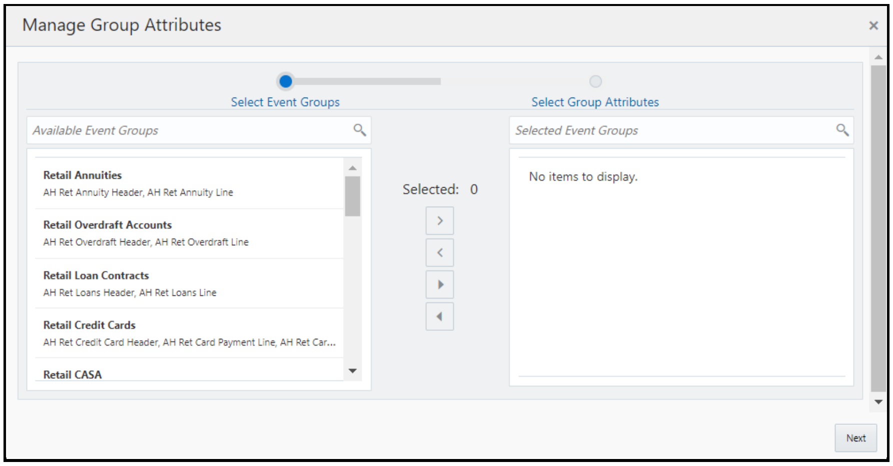
2. Enter the required details and click **Save**.

7.10.3 Managing Group Attributes

To manage group attributes, follow these steps:

1. From the **Subledger Event Grouping** window, click 🔧 to manage group attributes. The **Manage Group Attributes** window is displayed with the **Available Event Groups**.

Figure 42: Manage Group Attributes



Retail Annuities
AH Ret Annuity Header, AH Ret Annuity Line

Retail Overdraft Accounts
AH Ret Overdraft Header, AH Ret Overdraft Line

Retail Loan Contracts
AH Ret Loans Header, AH Ret Loans Line

Retail Credit Cards
AH Ret Credit Card Header, AH Ret Card Payment Line, AH Ret Car...

Retail CASA

No items to display.

2. Select the Event Groups that are created previously, for which you want to define the group attributes.

   a. Click > to move the required event groups to the list of the Select Group Attributes.

   b. Click ▶ to move all the event groups to the list of the Select Group Attributes.

   c. Click < or ◀ to remove the event groups back one by one or remove all, respectively.

3. After the event groups are selected, click **Next** to move to the **Manage Group Attributes** window.

   Here, by default, all the columns, which are a part of mappings in the selected event groups are listed under the selected Group Attributes, that is, they are a part of the group by clause for grouping.

   A list of grayed-out attributes cannot be removed from the group attribute list, as they are the basic dimensions on which ledger data is required.

4. You can move the non-mandated columns to LHS and remove them from aggregation or group by clause.

   a. Click > to move the non-mandated columns to LHS.

   b. Click ▶ to move the non-mandated columns to LHS.

   c. Click < or ◀ to remove the entities back one by one or remove all, respectively.

5. After all the required columns are confirmed, click **Save**. A confirmation message is displayed "*Are you sure you want to remove*?".

6. Click **Yes**.
   The **Subledger Group Attributes** are updated successfully.

8 Obtaining Balance Information from AHCS

This chapter provides details of the process that facilitates balance information against GL Chart of Accounts and with Support Reference information to be extracted from AHCS and ingested into FSDF.

8.1 Configuration and Data Files

This section details the configuration files and data files required before the execution of AHC Connectors.

8.1.1 Chart of Account to OFSAA Dimension Mapping

See the section OFSAA Chart of Account (COA) Mapping for more information on how this mapping must be done.

8.1.2 GL and SR Balance Extraction

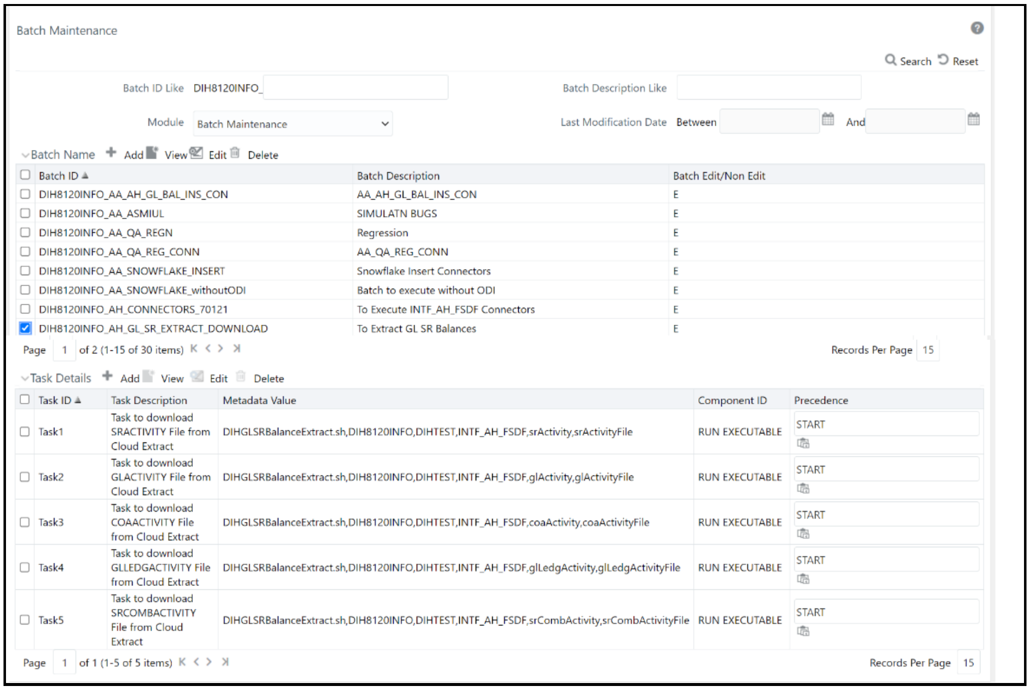
The following batches can be executed to extract and load the GL and Supporting Reference data into OFSAA.

1. AH_GL_SR_EXTRACT_DOWNLOAD

   Execution of this batch performs the following tasks such as it creates a BICC job or updates a BICC job if it already exists, schedules a job, extracts files from BICC and downloads the following files to EDS path.

   - ah_gl_balance.csv
   - ah_sr_balance.csv
   - ah_gl_code_combination.csv
   - ah_sr_ref_combination.csv
   - ah_gl_ledgers.csv

Figure 43: Batch Maintenance FSDF Extract Download



Batch Maintenance

Batch ID Like: DIH8120INFO_

Batch Description Like:

Module: Batch Maintenance

Last Modification Date: Between … And …

☐	Batch ID ▲	Batch Description	Batch Edit/Non Edit
☐	DIH8120INFO_AA_AH_GL_BAL_INS_CON	AA_AH_GL_BAL_INS_CON	E
☐	DIH8120INFO_AA_ASMIUL	SIMULATN BUGS	E
☐	DIH8120INFO_AA_QA_REGN	Regression	E
☐	DIH8120INFO_AA_QA_REG_CONN	AA_QA_REG_CONN	E
☐	DIH8120INFO_AA_SNOWFLAKE_INSERT	Snowflake Insert Connectors	E
☐	DIH8120INFO_AA_SNOWFLAKE_withoutODI	Batch to execute without ODI	E
☐	DIH8120INFO_AH_CONNECTORS_70121	To Execute INTF_AH_FSDF Connectors	E
☑	DIH8120INFO_AH_GL_SR_EXTRACT_DOWNLOAD	To Extract GL SR Balances	E

Page 1 of 2 (1-15 of 30 items) Records Per Page 15

☐	Task ID ▲	Task Description	Metadata Value	Component ID	Precedence
☐	Task1	Task to download SRACTIVITY File from Cloud Extract	DIHGLSRBalanceExtract.sh,DIH8120INFO,DIHTEST,INTF_AH_FSDF,srActivity,srActivityFile	RUN EXECUTABLE	START
☐	Task2	Task to download GLACTIVITY File from Cloud Extract	DIHGLSRBalanceExtract.sh,DIH8120INFO,DIHTEST,INTF_AH_FSDF,glActivity,glActivityFile	RUN EXECUTABLE	START
☐	Task3	Task to download COAACTIVITY File from Cloud Extract	DIHGLSRBalanceExtract.sh,DIH8120INFO,DIHTEST,INTF_AH_FSDF,coaActivity,coaActivityFile	RUN EXECUTABLE	START
☐	Task4	Task to download GLLEDGACTIVITY File from Cloud Extract	DIHGLSRBalanceExtract.sh,DIH8120INFO,DIHTEST,INTF_AH_FSDF,glLedgActivity,glLedgActivityFile	RUN EXECUTABLE	START
☐	Task5	Task to download SRCOMBACTIVITY File from Cloud Extract	DIHGLSRBalanceExtract.sh,DIH8120INFO,DIHTEST,INTF_AH_FSDF,srCombActivity,srCombActivityFile	RUN EXECUTABLE	START

Page 1 of 1 (1-5 of 5 items) Records Per Page 15

2. AH_CONNECTORS_70121

NOTE For some unknown reasons, when deployed in WebLogic, not all batches are created. Follow these steps to avoid the issue:

1. Log in to the WebLogic server admin console.
2. Navigate to **Console** and select **Services** and then select **Data Sources.**
3. Select **Atomic Schema.DS** and then select **Connection Pool**.
4. Update **Statement-Cache-Size** value to 0.
5. Restart the WebLogic server and redeploy connectors from the **AHC Refresh Interface** window.

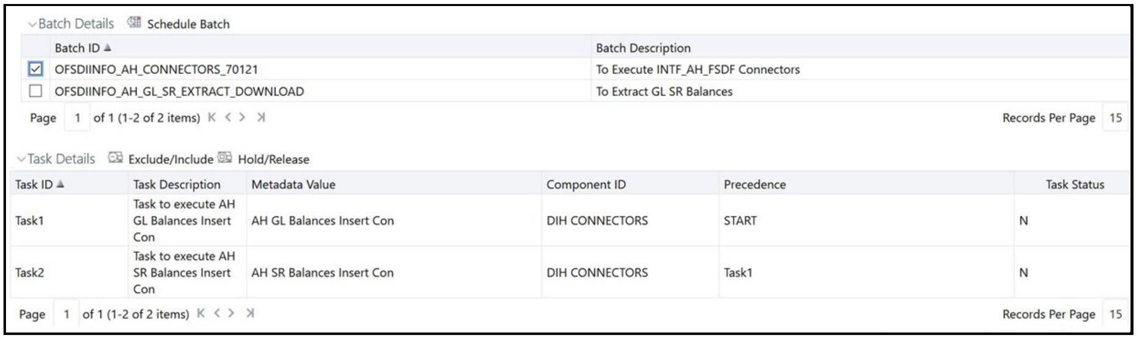
See the DIH User Guide for more information on executing the DIH Connectors.

8.2 Executing AHC Connectors

After completing the steps in the Chart of Account to OFSAA Dimension Mapping and Obtaining Balance Information from AHCS sections, execute the following steps to ingest balance information extracted from your instance of AHCS:

1. From the **Financial Services Data Foundation Integration with Fusion Accounting Hub Cloud Service** window, select **Execution,** and then select **Batch Execution**.
2. Publish **AH GL Balances Insert Con** and **AH SR Balances Insert Con Connectors**.
3. In each execution sequence or period, navigate to **Batch Orchestration** and search for auto-generated batch AH_CONNECTORS.

Figure 44: Batch Maintenance Window



	Batch ID ▲	Batch Description
☑	OFSDIINFO_AH_CONNECTORS_70121	To Execute INTF_AH_FSDF Connectors
□	OFSDIINFO_AH_GL_SR_EXTRACT_DOWNLOAD	To Extract GL SR Balances

Task ID ▲	Task Description	Metadata Value	Component ID	Precedence	Task Status
Task1	Task to execute AH GL Balances Insert Con	AH GL Balances Insert Con	DIH CONNECTORS	START	N
Task2	Task to execute AH SR Balances Insert Con	AH SR Balances Insert Con	DIH CONNECTORS	Task1	N

> **NOTE** For some unknown reasons, when deployed in WebLogic, not all batches are created. Perform the following steps to avoid the issue:
>
> 1. Log in to the WebLogic server admin console.
> 2. Navigate to **Console** and select **Services** and then select **Data Sources.**
> 3. Select **Atomic Schema.DS** and then select **Connection Pool**.
> 4. Update **Statement-Cache-Size** value to 0.
> 5. Restart the WebLogic server and redeploy connectors from the **AHC Refresh Interface** window.

4. Select **AH GL Balances Insert Con Task** and click **Edit**. The Batch Parameters window is displayed.
5. In **Batch Parameters** specify,
   a. PERIOD_NAME for which GL Balances must be extracted (for example, PERIOD_NAME='Aug-18').
   b. AH_CONSOLIDATION_FLAG (for example, AH_CONSOLIDATION_FLAG='C').
   c. LEDGER_ID for which GL balances must be filtered (for example, LEDGER_ID='1234').
6. Repeat steps 3 through 5 for **AH SR Balances Insert Con**.

> **NOTE** In Batch Parameters step, the PERIOD_NAME parameter alone is applicable.

7. Navigate to **Batch Execution**, search for **AH_Connectors**, specify **FIC_MIS_DATE**, and execute the batch.

   See the DIH User Guide for more information on executing the DIH Connectors.

9 Publishing Events Data to AHCS Using Extract Connectors

9.1 Configuration of Metadata

See the Managing Product Code Filters section for details.

9.2 Batch Execution

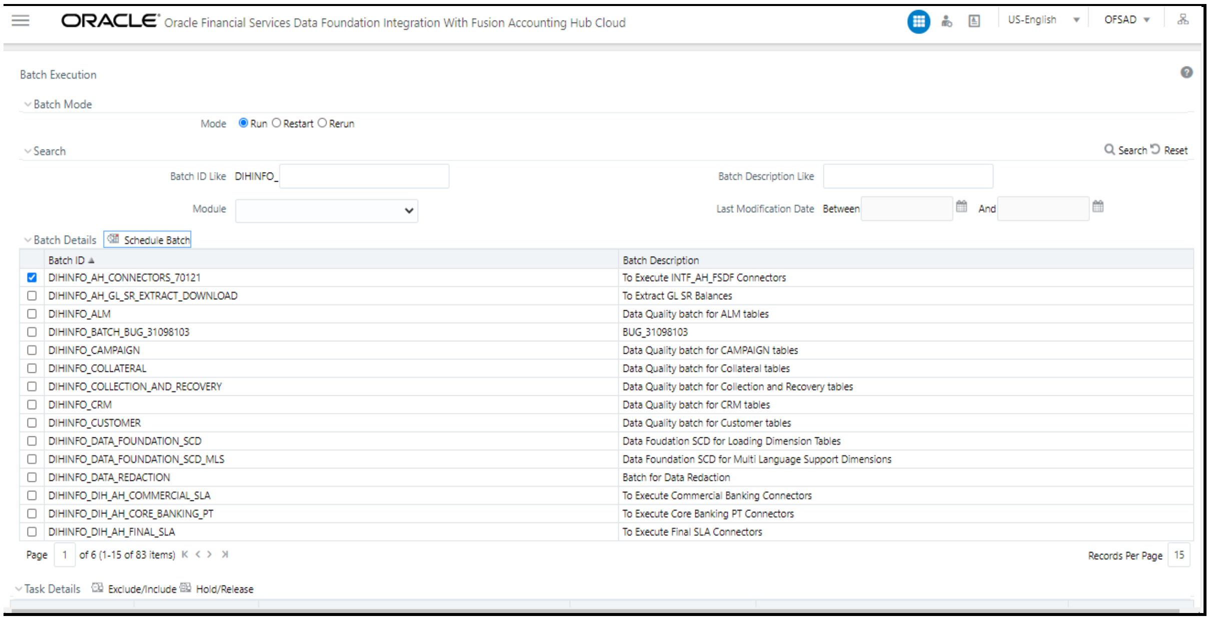
> **NOTE** You must have all the required configuration, including accounting rules, completed in your instance of AHCS before attempting to send events data using this process.
>
> See the Batch Re-Execution section before you proceed with Batch Execution.

In the **Batch Execution** window, there are batches created for each Subledger. These batches are created when you publish SLAs, as detailed in the Subledger section of this guide.

To execute a batch, follow these steps:

1. From the **Financial Services Data Foundation Integration with Fusion Accounting Hub Cloud Service** window, select **Execution,** and then select **Batch Execution**.

Figure 45: Batch Execution Window



ORACLE Oracle Financial Services Data Foundation Integration With Fusion Accounting Hub Cloud
US-English
OFSAD
Batch Execution
Batch Mode

Mode ● Run ○ Restart ○ Rerun

Batch ID Like DIHINFO_

Batch Description Like

Module

Last Modification Date Between And

Batch Details
Schedule Batch

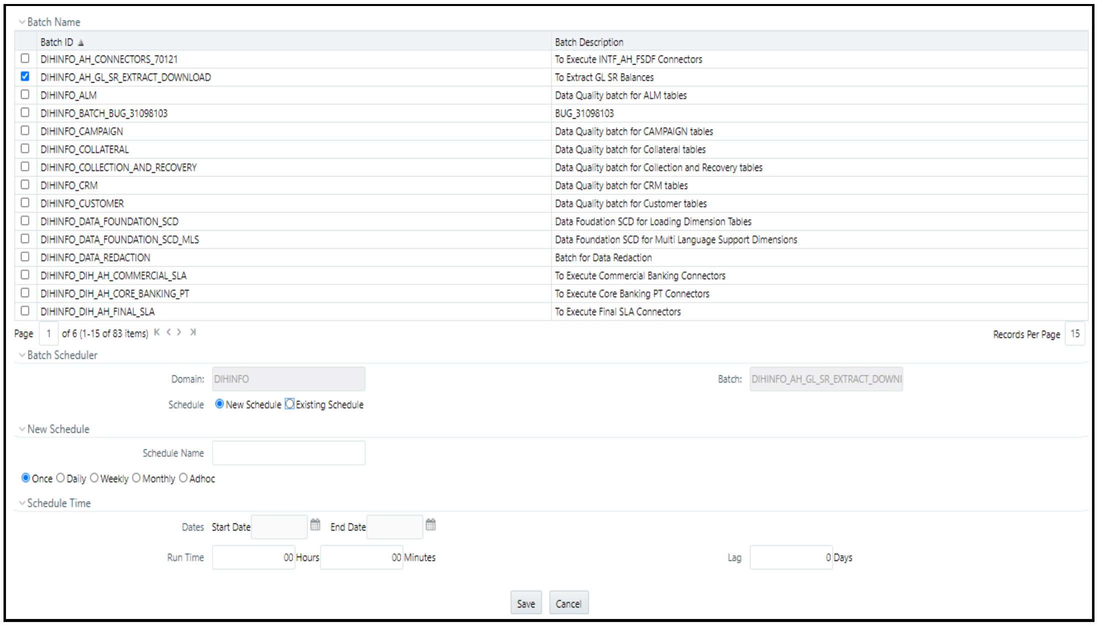
	Batch ID ▲	Batch Description
☑	DIHINFO_AH_CONNECTORS_70121	To Execute INTF_AH_FSDF Connectors
□	DIHINFO_AH_GL_SR_EXTRACT_DOWNLOAD	To Extract GL SR Balances
□	DIHINFO_ALM	Data Quality batch for ALM tables
□	DIHINFO_BATCH_BUG_31098103	BUG_31098103
□	DIHINFO_CAMPAIGN	Data Quality batch for CAMPAIGN tables
□	DIHINFO_COLLATERAL	Data Quality batch for Collateral tables
□	DIHINFO_COLLECTION_AND_RECOVERY	Data Quality batch for Collection and Recovery tables
□	DIHINFO_CRM	Data Quality batch for CRM tables
□	DIHINFO_CUSTOMER	Data Quality batch for Customer tables
□	DIHINFO_DATA_FOUNDATION_SCD	Data Foudation SCD for Loading Dimension Tables
□	DIHINFO_DATA_FOUNDATION_SCD_MLS	Data Foundation SCD for Multi Language Support Dimensions
□	DIHINFO_DATA_REDACTION	Batch for Data Redaction
□	DIHINFO_DIH_AH_COMMERCIAL_SLA	To Execute Commercial Banking Connectors
□	DIHINFO_DIH_AH_CORE_BANKING_PT	To Execute Core Banking PT Connectors
□	DIHINFO_DIH_AH_FINAL_SLA	To Execute Final SLA Connectors

Page 1 of 6 (1-15 of 83 items)

Records Per Page 15

2. Select a batch from the **Batch Details** table and then click **Schedule Batch**. The **Batch Scheduler** window is displayed.

Figure 46: Batch Scheduler Window



Batch Name

	Batch ID ▲	Batch Description
□	DIHINFO_AH_CONNECTORS_70121	To Execute INTF_AH_FSDF Connectors
☑	DIHINFO_AH_GL_SR_EXTRACT_DOWNLOAD	To Extract GL SR Balances
□	DIHINFO_ALM	Data Quality batch for ALM tables
□	DIHINFO_BATCH_BUG_31098103	BUG_31098103
□	DIHINFO_CAMPAIGN	Data Quality batch for CAMPAIGN tables
□	DIHINFO_COLLATERAL	Data Quality batch for Collateral tables
□	DIHINFO_COLLECTION_AND_RECOVERY	Data Quality batch for Collection and Recovery tables
□	DIHINFO_CRM	Data Quality batch for CRM tables
□	DIHINFO_CUSTOMER	Data Quality batch for Customer tables
□	DIHINFO_DATA_FOUNDATION_SCD	Data Foudation SCD for Loading Dimension Tables
□	DIHINFO_DATA_FOUNDATION_SCD_MLS	Data Foundation SCD for Multi Language Support Dimensions
□	DIHINFO_DATA_REDACTION	Batch for Data Redaction
□	DIHINFO_DIH_AH_COMMERCIAL_SLA	To Execute Commercial Banking Connectors
□	DIHINFO_DIH_AH_CORE_BANKING_PT	To Execute Core Banking PT Connectors
□	DIHINFO_DIH_AH_FINAL_SLA	To Execute Final SLA Connectors

Page 1 of 6 (1-15 of 83 items) Records Per Page 15

Batch Scheduler

Domain: DIHINFO Batch: DIHINFO_AH_GL_SR_EXTRACT_DOWNI

Schedule ● New Schedule ○ Existing Schedule

New Schedule

Schedule Name

● Once ○ Daily ○ Weekly ○ Monthly ○ Adhoc

Schedule Time

Dates Start Date End Date

Run Time 00 Hours 00 Minutes Lag 0 Days

3. Enter the details as of which data must be processed and click **Save**.
4. Click **Execute Batch** in the **Batch Execution** window.
5. See the OFSAAI User Guide for details on batch execution, runtime parameters, and monitoring.

   Each SLA batch consists of:

   - Connector execution task
   - Processes Event Grouping
   - Prepares the Header file
   - Prepares the Line file
   - Runs an executable file

   All tasks specified within the batch must be executed.

   The Run Executable task performs the following actions:

1. Identifies the extracted Header and Line CSV files.
2. Formats data by removing duplicate Header rows in both Header and Line files.
3. Generates the `Metadata.txt` file with SLA details.
4. Creates a .zip file including the Header, Line, and Metadata files.
5. See the following structure to locate and identify the .zip files generated:

   ```
   /<EDS_PATH>/<SLA_CODE>/XlaTransaction_<SLA_CODE>_<MIS_DATE>_<TIMESTAMP>.
   zip
   ```

6. For every execution, intermediate files are copied to the `temp` folder along with a log file.

   By default, AHCS disables automated triggering of events data processing after files are uploaded through its API to UCM. This can be enabled by updating the sixth runtime parameter of the final Run Executable task to Y before executing the batch.

With this enabled, the seventh parameter of this task can be set to 1 (where the .zip file is uploaded to UCM and no further action is taken) or 2 (where the .zip file is uploaded to UCM and import task in AHCS is triggered to process uploaded file) as appropriate for the purposes.

9.3 Batch Re-Execution

FSDF Integration with AHCS does not currently support incremental processing. Each execution causes all events data relevant for the MIS Date specified while executing the aforementioned batches, subject only to filters defined while setting up SLAs or related Connectors, to be extracted from Staging entities and published to AHCS.

In other words, batch processing can be performed only once per SLA for any given MIS Date. Re-execution can cause duplicated events data to be sent to your instance of AHCS, with no provision for automated rollback.

10 Metadata Object Migration

Metadata objects created by AHCS may also be migrated leveraging this mechanism, with the list including SLA parameters, Transcation attribute, Line attribute, Transcation Data Mapping, and Line Data Mapping, and batches.

Topics:

- Overview
- Performing Object Migration
- Exporting Object from Source Environment
- Importing Object into Target Environment

For more information on Connector Migration, see the DIH User Guide.

10.1 Overview

Using the object migration utility, you can migrate (export/ import) AHCS metadata objects across different setups. You can specify one or more objects within an object type or multiple object types. You can choose from where the Object Migration utility reads the data, that is, from CSV files or the OBJECTMIGRATION.xml file. For migrating objects using CSV files and OBJECTMIGRATION.xml file, see section Command Line Utility to Migrate Objects from OFSAAI User Guide.

The following sections detail how such Metadata Object migration can be achieved.

10.2 Performing Object Migration

Offline Object Migration is a two-step process:

- Exporting Objects from Source Environment
- Importing Objects into Target Environment

For both of these steps, see the sample file **OBJECTMIGRATION.xml**, which is also present at `$MIGRATION_HOME/conf/` in the OFSAAI setup.

Assumptions and Pre-requisites for this process are listed as follows:

- Both the Source and Target environment must be on the same data model version.
- Ensure that the source and target are at the same AHCS application level as well as at the same model level before migrating objects from source to target.
- ADI refresh is completed before this process.
- If the exported object already exists in the target environment or an object with the same name already exists in the target environment, then it is overwritten with new definition details.

Dependent objects for a SLA are Transcation attribute, Line attribute, Transcation Data Mapping, and Line Data Mapping.

10.3 Exporting Objects from Source Environment

To export objects from the source environment, follow these steps:

1. Replace placeholders of UserID, Infodom with Source UserID, and Infodom.
2. For `$Folder` put segment name for the infodom provided. Enter locale as 'en_US'.
3. `$FILE_NAME`: Specify the file name which is created under "metadata/archive" folder. For example, mention 'rules' in place of $FILE_NAME and you get **rules.dmp** in the archive folder.

   Fail On error: Fail on any error that occurs while restoring metadata. Mention 'Y' for Yes and 'N' for No.

   OVERWRITE: If Metadata exists in the system then Overwrite while restoring metadata. Enter 'Y' for Yes and 'N' for No.

   In Mode tag: mention EXPORT.

   For **FAILONERROR** and **OVERWRITE**, it's recommended to mention 'Y'.
4. In the OBJECT tag, enter the Code as SLA code, to export all definitions. This SLA code is same as the Subledger Short Name. If not, for each definition put an equal number of OBJECT tags with map ID and version number in comma-separated format.

   Type: Use the following for definitions:

   - 80 for SLA Migration
5. The format for the All OBJECTS tag is as follows:

   ```
   <OBJECTS TargetFolder="DIHUSERS"><OBJECT Code="RETAIL_SLA" Type="80"
   /></OBJECTS>
   ```
6. For three definitions, the OBJECTS tag is:

   ```
   <OBJECTS TargetFolder="DIHUSERS">

   <OBJECT Code="RETAIL_SLA" Type="80" />

   <OBJECT Code="INVESTMENT_SLA" Type="80" />

   <OBJECT Code="COMMERCIAL_SLA" Type="80" />

   </OBJECTS>
   ```
7. Execute `$MIGRATION_HOME/bin/ObjectMigration.sh` after providing executable permissions.
8. A file `$FILE_NAME.dmp` is created in $MIGRATION_HOME/metadata/archive
9. Move this file to `$MIGRATION_HOME/metadata/restore` folder. You can copy the file in the Target environment by creating a "**restore**" folder under the "**metadata**" directory (if not available).
10. Exporting definitions from the Source environment is done successfully.

10.4 Importing Objects From Target Environment

To import objects to the target environment, follow these steps:

1. Repeat 1-3 steps as followed in export mode.
2. In the Mode tag: mention IMPORT.
3. In the OBJECT CODE property, mention Subledger Short Name.

NOTE Everything that is exported is imported. You cannot choose specific definitions to move.

4. The format for the OBJECTS Tag is:

   ```
   <OBJECTS TargetFolder="DIHUSERS">
   <OBJECT Code="COMMERCIAL_SLA" Type="80" /></OBJECTS>
   ```

5. Execute `$MIGRATION_HOME/bin/ObjectMigration.sh` after providing executable permissions.
6. Check **DIH.log** for logs. It provides details such as the number of definitions that have successfully moved and other errors. Importing objects to the target environment is done successfully.

10.5 Validation Scenarios

The following is the list of validation scenarios for SLA migration:

- Any Custom SLA can be migrated.
- Migration of SLA with New SLA Code and New SLA Name can be performed.
- If the migration of SLA is performed with New SLA Code and Existing SLA Name, then migration is not performed.
- If the migration of SLA already present in the target environment with Existing SLA Code and Existing SLA Name, then it will overwrite the existing SLA in target environment.
- Out of the box SLAs cannot be overwritten, if you are using the same SLA Name and SLA Code.
- If the migration of SLA is performed with Existing SLA Code and New SLA Name, then migration is not performed.

OFSAA Support

Raise a Service Request (SR) in in My Oracle Support (MOS) for queries related to the OFSAA applications.

Send Us Your Comments

Oracle welcomes your comments and suggestions on the quality and usefulness of this publication. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most about this manual?

If you find any errors or have any other suggestions for improvement, indicate the title and part number of the documentation along with the chapter/section/page number (if available) and contact the Oracle Support.

Before sending us your comments, you might like to ensure that you have the latest version of the document wherein any of your concerns have already been addressed. You can access My Oracle Support site that has all the revised/recently released documents.