Oracle Financial Services Data Integration Hub

User Manual

8.0.2.0.0

ORACLE
FINANCIAL SERVICES

TABLE OF CONTENTS

Preface

Audience

Following are the intended audience for the DIH user guide:

- **ETL Developers:** The ETL Developers from the IT Department of the financial services institution, who do the data sourcing.
- **Business Analysts:** The business analysts from the IT Department of the financial services institution, who do the mapping of the tables.

Prerequisites

- OFSAA application should be installed for the staging tables. This can be installed before or after DIH.
- Oracle Data Integrator (ODI) must be installed.

Acronyms

Acronym	Description
DIH	Data Integration Hub
UI	User Interface
ODI	Oracle Data Integrator
ADI	Application Data Interface
KM	Knowledge Module
EDD	External Data Descriptor
Apps	Application

Glossary of Icons

This manual may refer to all or some of the following icons.

Icons	Description
	To create a function
	To Edit the details of a function
	To View the details of a function
	To Delete a function
	To view Dependencies

Related Information Sources

Along with this user manual, you can also refer to the following documents in the OTN documentation library:

- Oracle Financial Services Analytical Application-FCUBS Interface User Guide Release 8.0.2.0.0
- Oracle Financial Services Analytical Application-OBP Interface User Guide Release 8.0.2.0.0
- Oracle Financial Services Analytical Application-DRM Interface User Guide Release 8.0.2.0.0
- Oracle Financial Services Analytical Application-FAH Interface 8.0.2.0.0
- Oracle Financial Services Data Integration Installation Manual Release 8.0.2.0.0

1 Introduction to Data Integration Hub

This chapter provides an overview to Data Integration Hub (DIH), its benefits, and the structure of remaining chapters in the user guide.

This chapter covers the following topics:

1.1 Overview of Data Integration Hub (DIH)

Data Integration Hub (DIH) enables to load the data from the source systems to the OFSAA staging tables, through logical interfaces, known as Application Data Interfaces (ADI). Additionally, it enables extraction of data from the OFSAA system to downstream systems such as file, and Oracle Database. DIH provides a set of User Interfaces (UI), which is used to define and maintain External Data Descriptors (EDD), Application Data Interfaces, and also map the EDDs and ADIs through Connectors. The mappings can be one to one, one to many, and many-to-many.

The source systems that supply data include, the core banking systems, rating systems, modeling systems, and so on. In the absence of DIH, the data from the source systems are extracted, transformed, and loaded (ETL process) to the physical tables in Oracle Data Integrator (ODI). With DIH, the ETL activity is not replaced; but DIH serves as an abstract, logical layer to the physical tables in Oracle Data Integrator (ODI). In the logical layer, the entity and attribute names are mentioned in Business terms in English. In the physical layer, the table and column names are mentioned in technical terms.

The process is explained using the following example:

Let an attribute name in the logical layer be, Customer Account Number. The equivalent column name in the physical layer is V_Account_Number. Using the Connectors in DIH, the logical layer in ADI and physical layer in EDD are mapped. That is, in this case, Customer Account Number is mapped to V_Account_Number. The advantage of having DIH is, when the physical name of a column in an OFSAA table changes in the future, the financial institution need not change their data extracts. This also results in saving time, effort, and money for the migration activity.

Another major functionality of DIH is the support for Big Data. DIH supports source data in formats like HDFS and Hive. Additionally, it supports DB2, Sybase, SQL Server and Teradata.

1.2 Benefits of DIH

The benefits of DIH are as follows:

- Provides a level of abstraction and insulates upstream ETL processes from staging model changes.

One of the examples is when the name of the table or column changes in the physical layer, only the mapping with logical layer needs to be updated. This results in cost, time and effort savings.

- Support of Big Data.
- Provides a simplified mapping screen for loading data into OFSAA staging via an abstract layer.
- Removes ETL technicalities by prepackaging & predefining ODI parameters based on different use cases.
- Download Specification is a system managed meta object for upstream data mapping.
- Process/execution optimization for OFSAA data and known use cases. For example, in a direct one-to-one mapping, there won't be any hopping.
- It also hosts pre-defined integration with Oracle Products that play role of data providers to Analytical Applications.
- Helps in getting lineage right from source data to staging till results.

1.3 Summary of Chapters

This user guide provides information on mapping of a External Data Descriptor (EDD) with Application Data Interface (ADI). It begins with information on how a Parameter is defined, how a Source System is defined in DIH, how an EDD and ADI are set, and how they are connected. The metadata is then published to ODI. The execution part does not happen in DIH as it depends on batches that are scheduled in the Financial Services Institutions. However, details about that are mentioned in the last chapter.

The guide has the following chapters:

- **Data Flow in DIH:** This chapter explains the flow of data in DIH.
- **Logging into DIH and common functionalities:** This chapter explains how to login to DIH and common functions such as modifying, viewing and deleting, using Parameter as an example.
- **Setting up Platform and Parameters:** This chapter explains how to configure and edit the platform and parameters.
- **Defining Source Systems:** This chapter explains how to define the External Data Store information.
- **External Data Descriptor:** This chapter explains how to define the source table/file, and adding the different components
- **Setting ADI:** This chapter explains how to configure the ADI for viewing an OFSAA Data Interface
- **Mapping EDD between ADI:** This chapter explains how to map one or more EDDs to ADI and how to create and view a connector
- **Refresh:** This chapter explains about the Refresh feature in DIH.

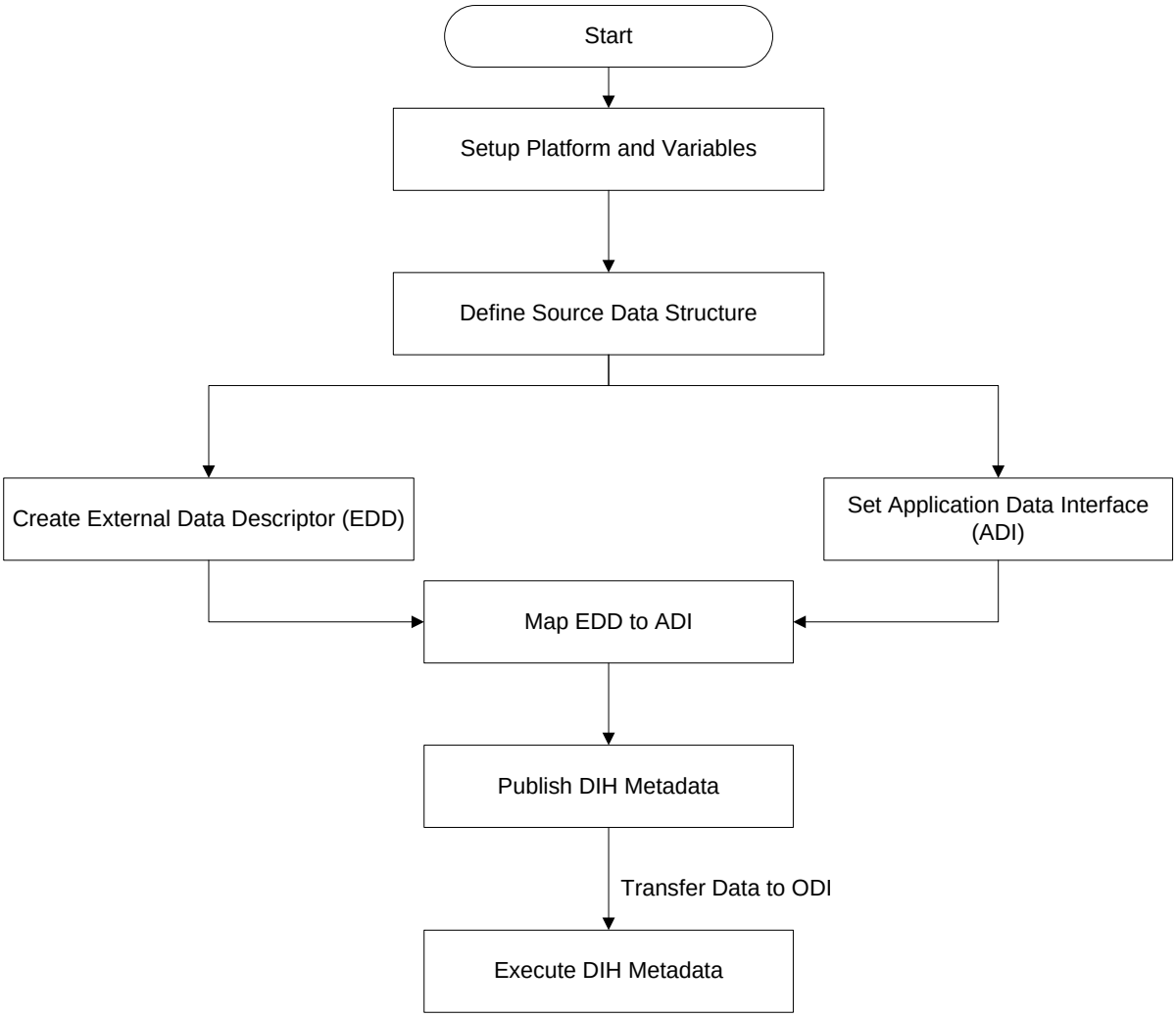
- **Publishing DIH Metadata:** This chapter explains how the DIH metadata is transferred to Oracle Data Integrator (ODI).
- **Executing DIH Metadata:** This chapter explains as to how after publishing, the scheduled batches are executed.
- **Execution History:** This chapter provides information about the execution history.
- **Metadata Browser:** This chapter details the metadata browser.

2 Data Flow in DIH

This chapter provides high-level information of the data flow used in DIH.

2.1 DIH Dataflow Diagram

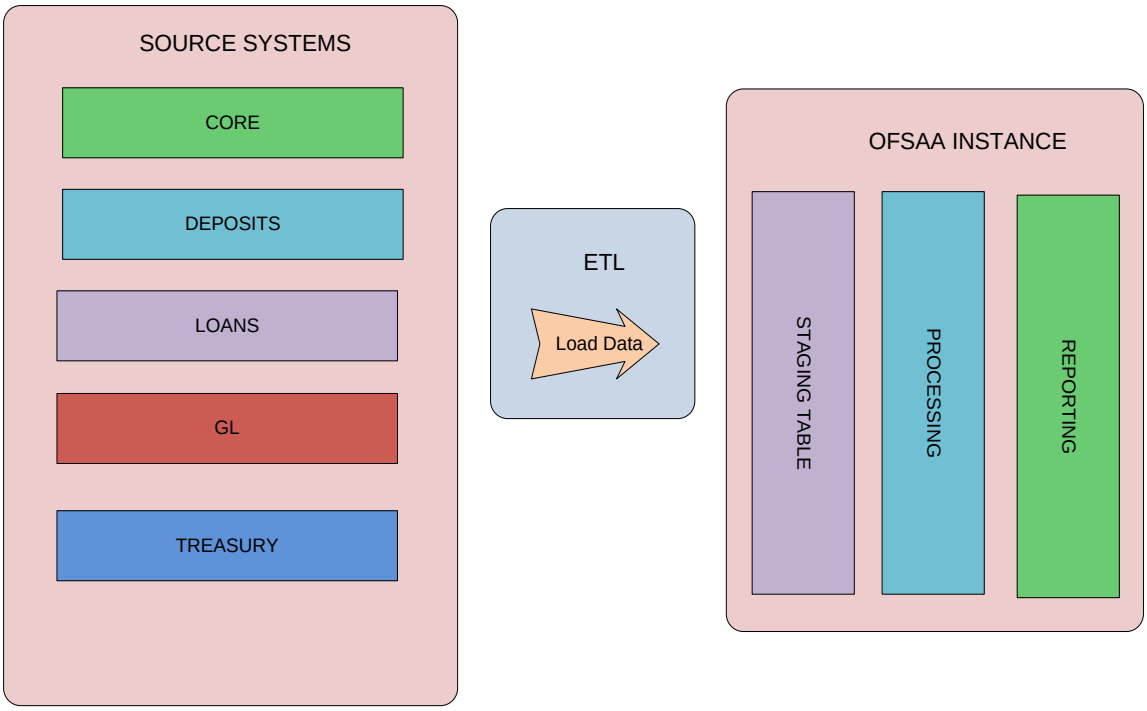
The following is the DIH data flow diagram:



For mapping between the EDD and ADI, you need to define the parameter place holder (optional), set up the source, define the EDD, and then connect the EDD and ADI. ADI is auto seeded from the OFSA Applications data models that are installed.

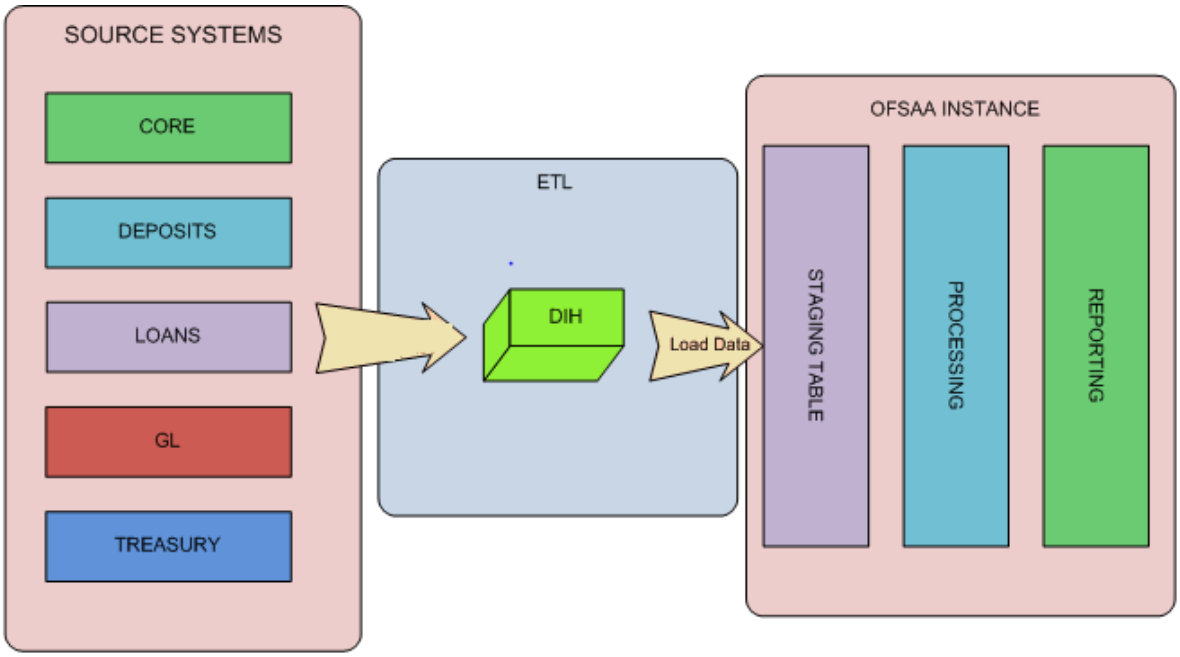
2.2 Data Flow Environment without DIH

In this data flow environment diagram without DIH, the data is loaded from the source systems, to the server in which OFSA Applications are installed (OFSAA Instance). In the OFSAA Instance, data is loaded into the staging table.



2.3 Data Flow Environment with DIH

The DIH is then used to pull data directly from the source tables, to the OFSAA staging tables.


SOURCE SYSTEMS
CORE
DEPOSITS
LOANS
GL
TREASURY
ETL
DIH
Load Data
OFSAA INSTANCE
STAGING TABLE
PROCESSING
REPORTING

3 Logging into DIH and Common Functionalities

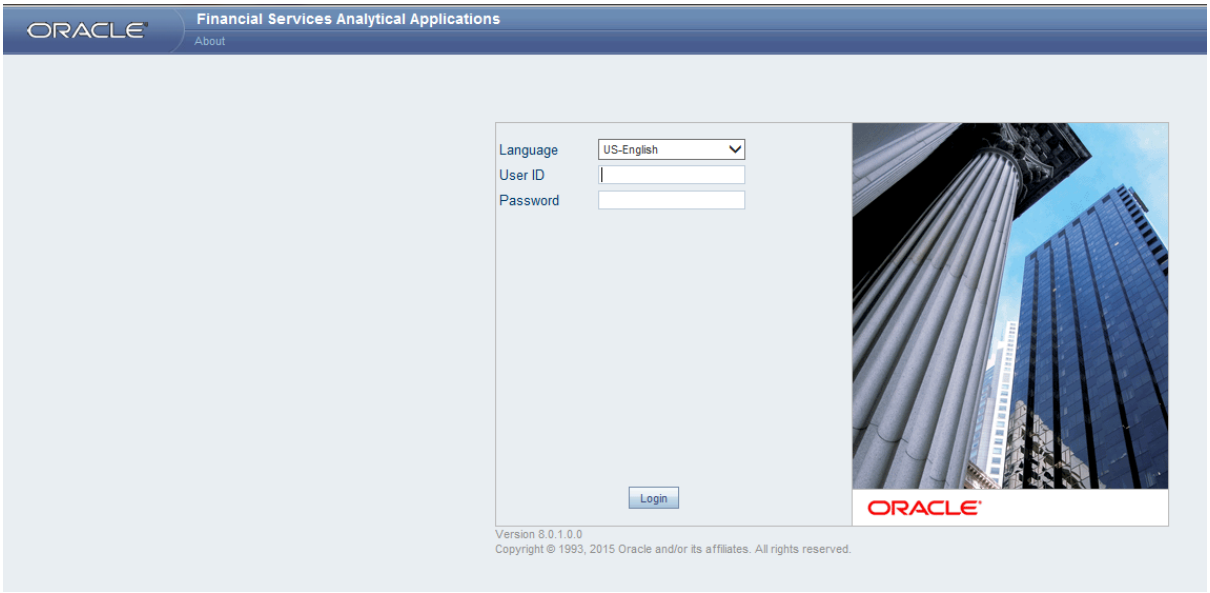
In this chapter, details on how to log into the DIH application are explained. There are a few common functionalities which are used in different modules of DIH. To avoid repetition, these functionalities are explained under Common Functionalities section, taking Parameter as an example.

This chapter has the following sections:

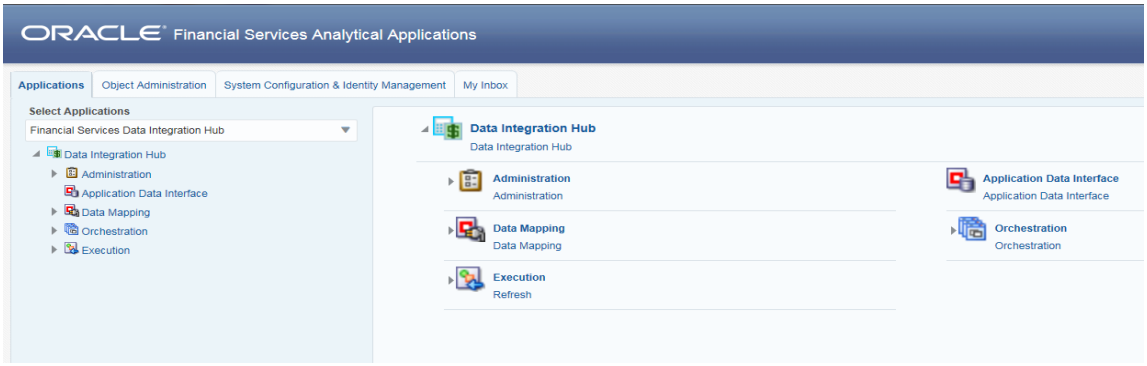
- **Logging into DIH**
- **Common Functionalities**

3.1 Logging into DIH

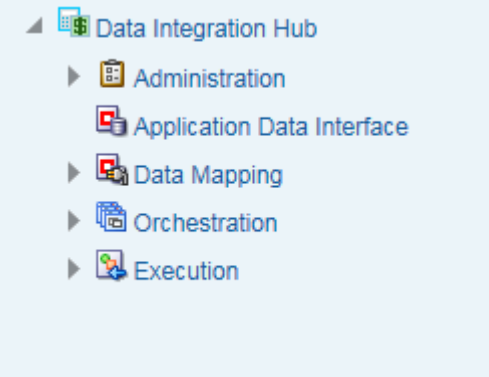
Access the DIH application using the login credentials (User ID and password). The built-in security system ensures that you are permitted to access the window and actions based on the authorization only.



After logging into the application, the following window appears:



Click the **Data Integration Hub** option in the Left Hand Side (LHS) of the menu. The **DIH** window appears.



The DIH has the below five sub links for data mapping:

- Administration
- Application Data Interface
- Data Mapping
- Orchestration
- Execution

3.2 Common functionalities

You can modify, view, delete, and find dependencies of a Parameter, Source, EDD, ADI, and Connectors. Following sections explains how this is done, taking the example of a parameter.

3.2.1 Modifying a Parameter

You can edit an existing Parameter, other than the Parameter name.

To edit a Parameter:

1. Select the checkbox adjacent to the Parameter name.
2. Click Edit button from the tool bar. The *edit parameter* window appears.
3. Only the Parameter description, Parameter Type and the Value can be edited in this window. Update the required details.
4. Click **Submit** to save the changes made.

3.2.2 Viewing a Parameter

You can view a Parameter at any given point. To view an existing Parameter:

1. Select the checkbox adjacent to the Parameter name.
2. Click View button from the tool bar.

The *View parameter* window displays the details of the selected Parameter.

3.2.3 Deleting a Parameter

This option only checks the higher order object. That is, if the order has dependency, you cannot delete unless the dependency is removed.

Example: Assume Parameter is used in Connector. Then, unless that Connector is deleted, the used Parameter cannot be deleted.

To delete an existing parameter:

1. Select the checkbox adjacent to the Parameter name.
2. Click Delete button from the tool bar. A confirmation dialogue appears
3. Click **OK**. The Parameter details are deleted.

3.2.4 Dependency

As the name suggests, on clicking the Dependency icon, it lists where the entire parent Parameter has dependency. That is, you cannot delete a child file without deleting the parent file.

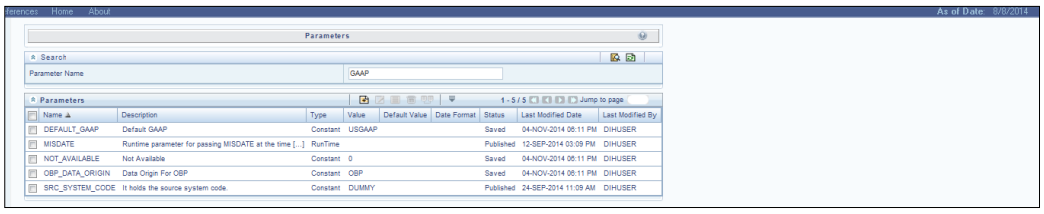
3.2.5 Search and Filter

The Search and Filter option in the user interface helps you to find the required information. You can enter the nearest matching keywords to search, and filter the results by entering information on the additional fields.

For example: enter the Parameter name as ‘GAAP’ in the search field.

Click the search icon to start the search or refresh icon to reset the search fields.

The entire Parameter name with GAAP is listed.



4 Setting up the Platform and Parameters

This chapter has the following sections:

4.1 Settings

Setting up the platform and parameters (place holder) is the initial activity in DIH.

The **Settings** option allows in maintaining the Oracle Data Integrator (ODI) setup information and application variables.

Click the **Administration** option from the LHS menu and select **Settings** option. The Settings window appears in the RHS.

This window captures the ODI set up information.

NOTE: Only the DIH administrator is authorized to edit this field.

4.1.1 Editing the Settings

While editing the Settings, the fields that are displayed are explained below.

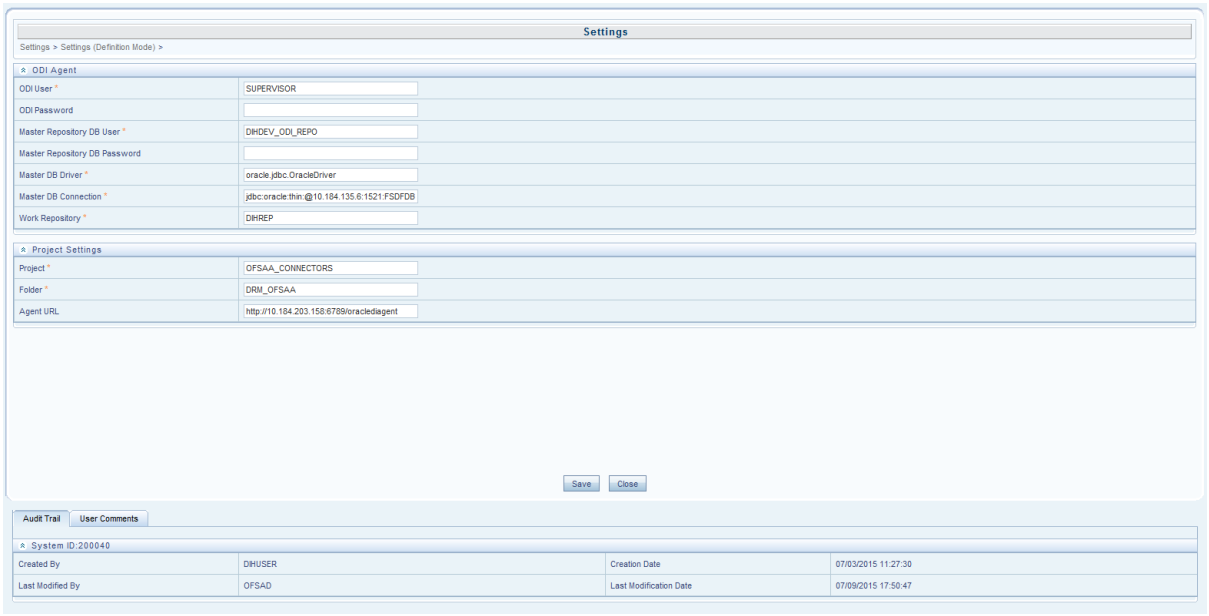
Fields and their descriptions

Fields	Description
Fields marked in red asterisk(*) are mandatory	

Fields	Description
Fields marked in red asterisk(*) are mandatory	
ODI User	The ODI supervisor user name you defined when creating the master repository or an ODI user name you defined in the Security Navigator after having created in the master repository.
ODI Password	The ODI supervisor password you defined when creating the master repository or an ODI user password you defined in the Security Navigator after having created the master repository.
Master Repository DB User	Database user ID/login of the schema (database, library) that contains the ODI master repository.
Master Repository DB Password	This is the user's password.
Master DB Driver	Specifies the driver required to connect to the RDBMS supporting the master repository created from the dropdown list. The default value is oracle.jdbc.OracleDriver. It need not be changed if it is on Oracle database.
Master DB Connection	The URL used to establish the JDBC connection to the database hosting the repository. The format is jdbc:oracle:thin:@<Hostname/IP Address>:<Port Number>:<Service Name>
Work Repository	The name of the work repository that has been created previously (Example: *WorkRep1*).
Project	Specify the Project Name created in ODI.
Folder	Specify the folder name under the project created in ODI so that all packages is created under this location.
Agent URL	Specify the agent URL where the ODI agent is running. This is used to execute a DIH connector from OFSAAI batch/RRF. This is not needed to do data mapping. The format is http://<Hostname/IP Address where ODI agent is running>:<PORT Number>/<Agent Context Name>

Procedure to Edit the Settings

1. Select the ODI User and click Edit button from the tool bar. The following window appears.

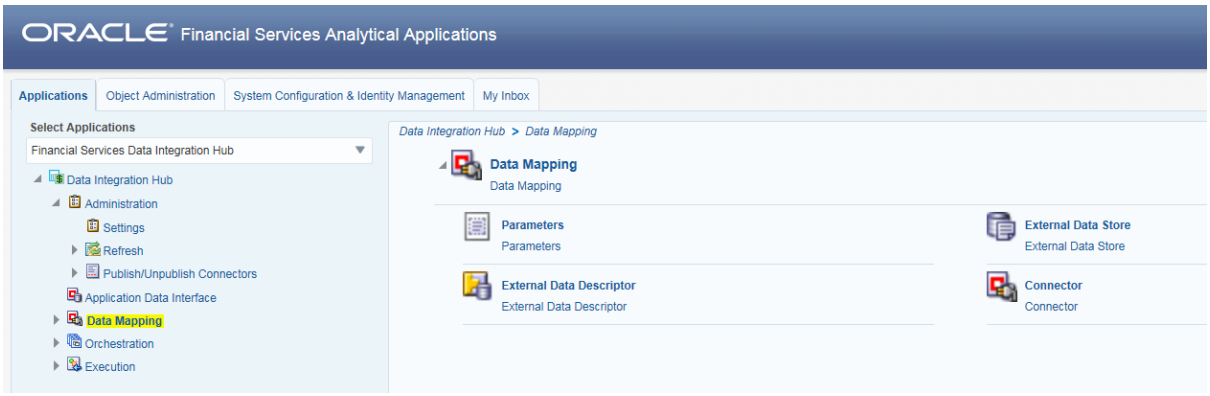


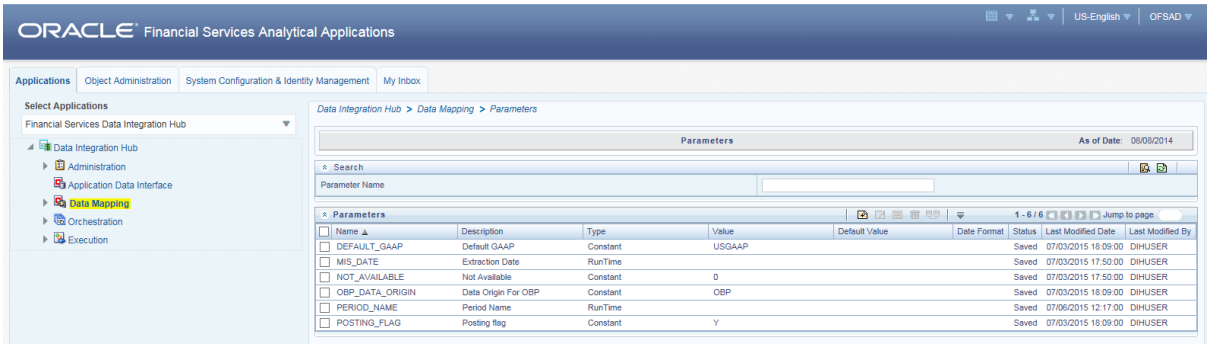
2. Enter the ODI User, Master Repository DB User, Master DB driver, Master DB Connection, Work Repository under the *ODI Agent* tab.
3. Under the *Project Settings* tab, enter the Project and Folder fields. The field details are explained as tabulated in the Fields and their Description section.
4. Enter the details and click **Submit**.

The *Audit Trail* section at the bottom of the window displays the information of the activities undertaken in this window. The *User Comments* section facilitates you to add or update additional information as comments.

4.2 Parameters

Parameters are place holders and constant values that have different uses in DIH. Click the **Data Mapping** option on the LHS. Select **Parameters** for the screen to appear.

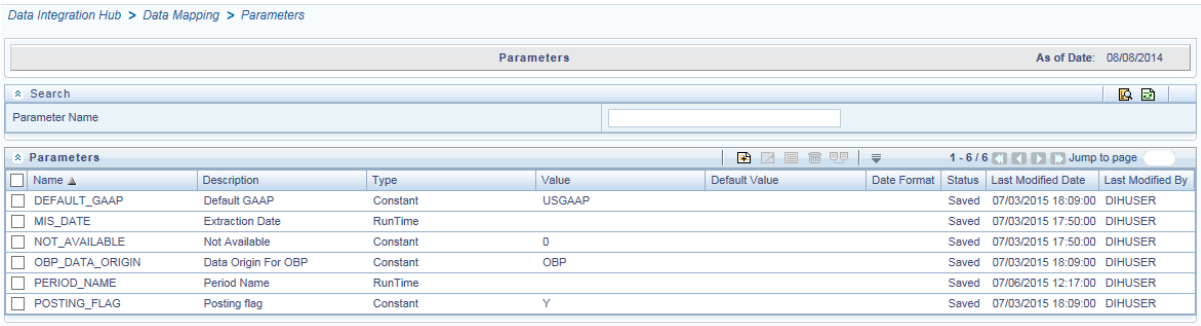




The uses of Parameters are explained in the following sections.

4.2.1 Parameters in EDD Definition

While defining an EDD, parameter can be used as a place holder in a data filename.



For example:

Consider a table with 2 columns, such as Account number and Balance.

Account Number	Balance
A1	1000
A2	1000
A3	1000
A1	1000

A2	1500
A3	1500

In the above example, a customer has 3 accounts (A1, A2 and A3).

The customer has deposited different amounts on January 1st and 2nd 2014. The csv data files can be created for those two dates as follows:

- The account transaction for January 1st 2014 is saved as td_contracts_/01012014/.csv
- The account transaction for January 2nd 2014 is saved as td_contracts_/01022014/.csv

If a parameter, MISDATE is defined as a runtime, this can be used as a place holder that substitutes date in mmddyyyy format. That is, the data filename can be mentioned as td_contracts_%#MISDATE%.csv. When this file is called, it substitutes the date in the file name, dynamically, in the run time.

Parameters Data Types need not always be RunTime. They can be Constants or values like Current Date, which can also be used to substitute a value in a data filename.

4.2.2 Parameters in Connector

Parameters are used while defining the mapping, between EDD to ADI. While mapping the ADI to EDD, the fields or columns within the ADI needs to be mapped to the fields in EDD. If there are no corresponding extracts in EDD, parameters can be used to identify the default values for certain ADI elements. Also, parameters can be used while defining derived columns during mapping.

For Example: If you want to use the Runtime mis date as parameter then, it needs to be converted to a date first. So, the expression would be following.

To_char(to_date(#DIHDEV.MIS_DATE,'dd-MON-YYYY'),'MM')

NOTE: Runtime batch mis date will be in string format. It needs to be converted to DATE. The date format used here should be a valid SQL date format.

4.2.3 Defining a Parameter

While defining a Parameter, the fields that are displayed are explained as tabulated.

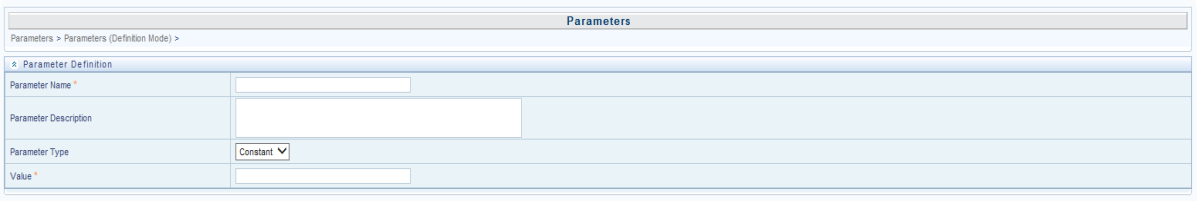
Fields and their descriptions

Fields	Description
Fields marked in red asterisk(*) are mandatory	
Parameter Name	The name for the place holder that you want to define. For example, MISDATE, which can be used as a place holder for Date.
Parameter Description	The description for the parameter you want to define. In this example, the description can be, “MISDATE can be used to substitute the date values for each day, dynamically, in mmddyyyy format.”
Parameter Type	There are 3 parameter data types: ▪ Constant: Constant data type is selected, for substituting a constant value. ▪ RunTime: RunTime data type is selected for substituting a value dynamically, in run time. In the example that is used here, MISDATE can be selected as Run Time, because, it is used to make a substitution, dynamically. ▪ CurrDate: CurrDate data type is selected for substituting a value as Current System Date.
Value	Only for constant types. Holds the actual value that of the parameter

Procedure to define a Parameter

To define a new Parameter:

1. Click the Add ( ) button from the tool bar menu. The following window appears.



2. Enter the Parameter Name and description.

   Example: Parameter Name: MISDATE

3. Select the Parameter Type from the drop down list and enter the Value in its respective field.

   The fields are explained as tabulated in the Fields and their Description section

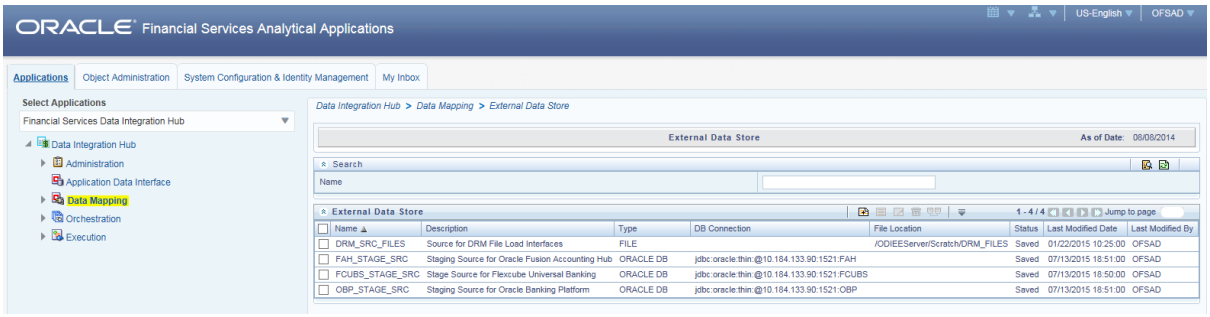
4. Enter the above details and click **Submit** to save the changes made.

The *Audit Trail* section at the bottom of the window displays the information of the parameter created.
The *User Comments* section facilitates you to add or update additional information as comments.

5 Defining Source Systems

This option enables to define the External Data Store information. DIH supports Oracle database, XML, EBCDIC, Flat file data sources, DB2, SQL server, Sybase, Teradata, and Big data sources such as HDFS and Hive.

Click the **Data Mapping** option on the LHS and select **External Data Store.** The *External Data Store* window appears in the RHS.



In the **Source Systems** section of the *External Data Store* window, you can define, edit, and delete a source.

You can make use of the Search option to search for a specific Source.

This chapter has a section named Creating a Source.

5.1 Creating an External Data Store

While creating a Source, the fields that are displayed are explained as tabulated.

Fields and their descriptions

Fields	Description
Fields marked in red asterisk(*) are mandatory	
Source Name	Is the name of the Source we are going to create. Example: USG_FILE_SRC. This must be in uppercase.

Fields	Description
Fields marked in red asterisk(*) are mandatory	
Source Description	A description for the Source is to be mentioned. Example: The landing zone where all the required files will be arrived for loading data into OFSAA.
Source Type	The available options are: ▪ EBCDIC: Extended Binary Coded Decimal Interchange Code (EBCDIC) File is a binary code for alphabetic and numeric characters. ▪ FILE: American Standard Code for Information Interchange (ASCII) File is a character-encoding scheme. ▪ HDFS: Hadoop Distributed File System (HDFS) is an open source and fundamentally a new way of storing and processing data. It enables distributed processing of huge amounts of data across industry-standard servers that both store and process the data, and can scale without any limits HDFS is Big Data in a raw format. ▪ HIVE: **Hive** provides a mechanism to project structure onto the data in Hadoop. Big data in tabulated format. On selecting this option, all the HIVE type files are displayed. ▪ ORACLE DB: An **Oracle database (DB)** is a collection of data treated as a unit. The purpose of a database is to store and retrieve related information. On selecting this option, all the ORACLE DB type files are displayed. ▪ XML: Extensible Markup Language (**XML**) is a markup language that defines a set of rules for encoding documents. ▪ DB2: IBM DB2 is a family of database server products. These products support the relational model. ▪ SQL Server: Microsoft SQL Server is a relational database management system. It is a software product with the primary function of storing and retrieving data as requested by other software applications which may run either on the same computer or on another computer across a network. ▪ Sybase: Sybase produces software to manage and analyze information in relational databases. ▪ Teradata: Teradata Corporation provides analytic data platforms, applications and related services. Its products enable users to consolidate data from different sources and make the data available for analysis. With respect to the above example, select the Source Type as File.

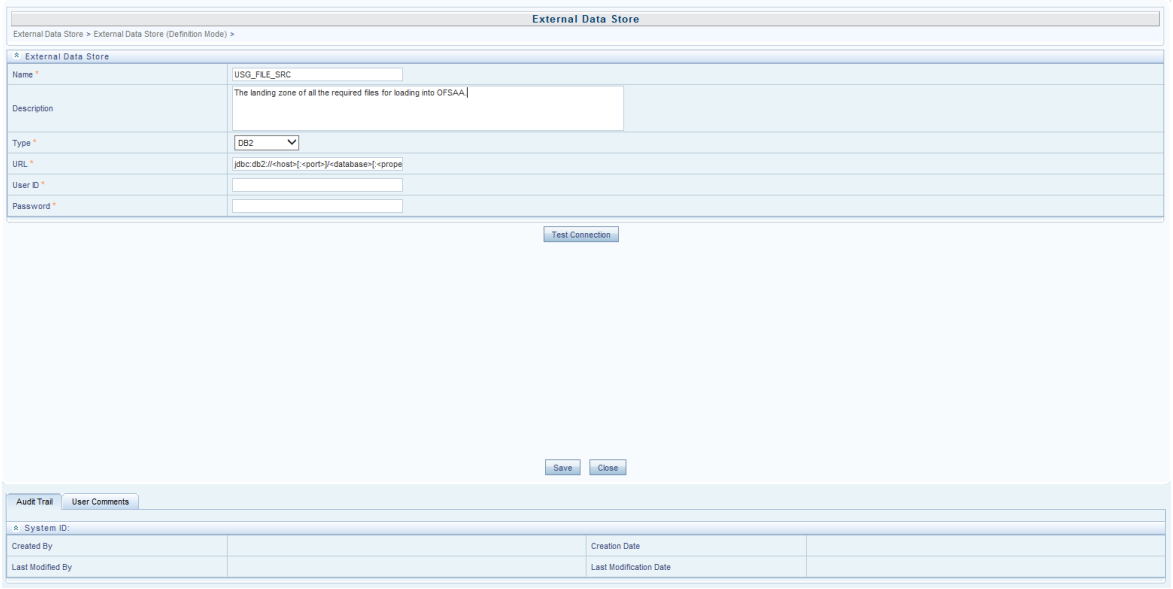
Fields	Description
Fields marked in red asterisk(*) are mandatory	
	Note: For Db2/SQL Server/Teradata/Sybase to work as source for DIH, Jar files with the following JDBC driver classes should be copied into the following paths: **1. <FIC**_HOME>/ficweb/webroot/WEB-INF/lib **2. <ODI_HOME>/odi/agent/lib** ▪ DB2: com.ibm.db2.jcc.DB2Driver ▪ SQL Server: com.microsoft.sqlserver.jdbc.SQLServerDriver ▪ Teradata: com.ncr.teradata.TeraDriver ▪ Sybase: com.sybase.jdbc3.jdbc.SybDriver These are third party libraries. Therefore, the names of the files may change for each version. Refer to the documentation of these databases, for the exact name of the file which contains these drivers. This field changes depending on the Source type you select. For example: If the Source type is selected as **File,** the next field is **File Location.** In this field, you need to mention the location of the file. Example: /landingzone/inputfiles If the Source type is selected as **HDFS**, in addition to **File Location**, the following field appears: **HDFS JDBC URL:** In this field, you need to mention the URL of the DB. Example: hdfs://<Host>:<Port> If the Source type is selected as **HIVE**, in addition to **File Location**, the following field appears: **JDBC URL:** In this field, you need to mention the URL of the DB. Example: jdbc:hive://<Host>:<Port>/<Schema> If the Source type is selected as **ORACLE DB**, the following Fields appear **URL**: In this field you need to mention the URL of the DB **Note**: The JDBC URL of the DB should be defined using the service name of the DB Instance. For example: jdbc:oracle:thin:@//<hostname>:<port>/<servicename> **User ID**: Enter the User ID or Schema Name **Note**: Only the tables that are a part of this Schema can be defined in the External Data Descriptor Page. **Password** : Enter a password If the Source type is selected as **SQL Server**, the following Fields appear **URL**: In this field, you need to mention the URL of the DB **Note**: The JDBC URL of the DB should be defined. For example: jdbc:sqlserver://<hostname>\SQLExpress

Fields	Description
Fields marked in red asterisk(*) are mandatory	
Location	**User ID**: Enter the User ID **Password** : Enter password. **Schema**: Enter the Schema name **Note**: Only the tables that are a part of this Schema can be defined in the External Data Descriptor Page. If the Source type is selected as **Sybase**, the following Fields appear **URL**: In this field, you need to mention the URL of the DB **Note**: The JDBC URL of the DB should be defined. For example: jdbc:sybase:Tds:<hostname>:<port> **User ID**: Enter the User ID **Password :** Enter password **Schema:** Enter the Schema name **Note**: Only the tables that are a part of this Schema can be defined in the External Data Descriptor Page. If the Source type is selected as **Teradata**, the following Fields appear **URL** : where you need to mention the URL of the DB **Note**: The JDBC URL of the DB should be defined. For example: jdbc:teradata://<hostname> **User ID**: Enter the User ID **Password :** Enter password **Schema:** Enter the Schema name **Note**: Only the tables that are a part of this Schema can be defined in the External Data Descriptor Page. If the Source type is selected as **DB2**, the following Fields appear **URL**: In this field, you need to mention the URL of the DB **Note**: The JDBC URL of the DB should be defined. For example: jdbc:db2://<hostname>[:<port>]/<database> **User ID**: Enter the User ID **Password:** Enter password **Database:** Enter the database name **Note**: Only the tables that are a part of this database can be defined in the External Data Descriptor Page.

Procedure to create an External Data Store

To create a new source from *External Data Store* window:

1. Click Add button from the tool bar. The *External Data Store* window appears.



2. Enter the Source Name. Example: USG_FILE_SRC and enter a description for the same.
3. Select the Source Type from the drop down list. Example: File.
4. The rest of the fields will change as per the option selected for Source Type. If Source Type is selected as File, The File Location field needs to me mentioned.

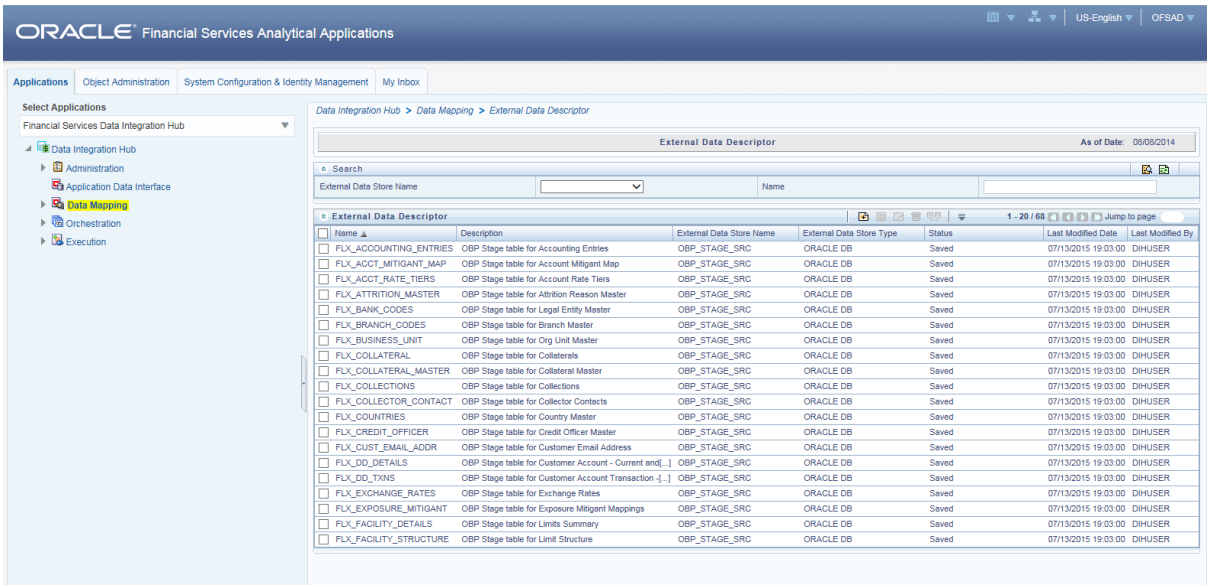
   The fields are explained as tabulated in the Fields and their Description section.
5. Click “Test Connection” to test the connection details (User ID/ Password) for the database types **DB2, HIVE, Oracle DB, SQL Server, Sybase,** and **Teradata.**
6. Enter the above details and click **Save**.

The *Audit Trail* section at the bottom of the window displays the information of the source created. The *User Comments* section facilitates you to add or update additional information as comments.

6 External Data Descriptor

The External Data Descriptor (EDD) allows defining or registering external source data structures in DIH. The EDD definition can be used for loading and extracting data into/from OFSAA. The control tab is not used for extraction. It is only used when reconciliation is required while loading data into OFSAA.

Click the **Data Mapping** option on the LHS and select **External Data Descriptor.** The *External Data Descriptor* window appears in the RHS.



This chapter has a section named creating an EDD.

6.1 Creating an EDD

While creating an EDD, the fields that are displayed are explained as tabulated.

Fields and their descriptions

In the Source Data tab:

Fields	Description
Fields marked in red asterisk(*) are mandatory	
Data File Name	You can add multiple data files to an EDD. For example, You need to add Term Deposits Contracts data file. There are Term Deposits Contracts data files for Retail as well as Corporate accounts. Therefore, to get both these details, you first add the Term Deposits Contracts data file for Retail accounts, say, td_contracts%#MISDATE%_1.csv and as the next record, add Term Deposits Contracts data file for Corporate accounts. Example: td_contracts%#MISDATE%_1.csv
Record Delimiter	The records are stored differently in different operating systems. The options available are: ▪ MS-DOS ▪ Unix ▪ No Record Delimiter ▪ Other With respect to the example, select Unix.
Read from template	A template contains all the values and is in Excel file format. If the template is not available, you need to create it manually by clicking the Add button, under *Source Data Elements* tab. If the template is available, you can browse for the template. Refer to the File SDI Template.

In the Data Elements tab:

Fields	Description
Fields marked in red asterisk(*) are mandatory	
Name	Name of the field in EDD. Example: Field name in a file or column name in a table. Note: Field name of XML type should not be more than 25 characters and for others should not be more than 30 characters.

Fields	Description
Fields marked in red asterisk(*) are mandatory	
Type	This shows the Data type of the field. Example: String, Number, EBCDIC and so on.
Physical Length	This is applicable only for EBCDIC format. This is the length of EBCDIC data type. In case of file, it is length only.
Logical Length	This is applicable only for EBCDIC format. This is the logical length of EBCDIC data type. It gets automatically calculated in template based EDD creation. In case of file, it is available.
Precision	It is to specify the decimal point. Example: 10.3.
Format	It is the format to specify the date format. Example: YYYY-MM-DD. The format should be in capitals.
Record Type Code	This identifies the Record type in a file where Header, Trailer and Data are of different record length and type. The values can be any string available in the text file. This value is only possible for the first field in a file. Example: The values can be DATA; CTRL to specify it is a control record.

In the Control tab:

Fields	Description
Fields marked in red asterisk(*) are mandatory	
File Name	Specify the name of the file.

Fields	Description
Fields marked in red asterisk(*) are mandatory	
File Format	There are 2 options: ▪ Fixed Length: The file will have records and columns with a fixed length. Each column will have a predetermined and unchanging size, set when the record layout is designed, and the sum of the column sizes will add up to the record size. ▪ Delimited: There will be a separation of the records and columns using a delimiter character like comma, semicolon, hyphen and so on. With respect to the above example, select *Delimited.*
Column Delimiter	If the File Format is selected as Fixed Length, the Column Delimiter would by default be Other. If the File format is selected as Delimited, the following options are available in the drop down list. ▪ Other ▪ Space ▪ Semicolon ▪ Comma ▪ Tab With respect to the above example, select *Comma.*
Record Type Code	Used to uniquely identify a record within a file. Many times, Bank provides file having data and control record within same file. In that case, in order to distinguish between data record and control record, the first field is Record Type. It has a specific value to identify that. Here, specify the value that identifies the Data. Values can be 'DATA' and so on. For Control record, the value is specified under control tab. Only the first field of a file is used for Record Type.

Fields	Description
Fields marked in red asterisk(*) are mandatory	
Record Delimitter	The records are stored differently in different operating systems. The options available are: ▪ MS-DOS ▪ Unix ▪ No Record Delimiter ▪ Other With respect to the example, select Unix.
Text Qualifier	It's a character that identifies a text. This is used when some characters exists within a text. Generally, double quotes are used, prefixed and suffixed with a text. This is optional.
Skip number of records	The number of records to be skipped is to be mentioned. The records are skipped from the top. Generally, this is to skip the Headers.
Decimal separator	This mentions up to which decimal digit you want to view the result in.
Record Type Length	This is applicable only for Control records that are of Fixed length. The length of the record type value to pickup the correct record. For example, if control record is "DATATotal Records400" .and DATA is the Record type, the length is '4'.
Control Name Length	Based on the above example, Control name is "Total Records". Hence, the Control Name Length is '13'.
Control Value Length	Based on the above example, the Control value is 400. Hence, the length of the control Value is '3'

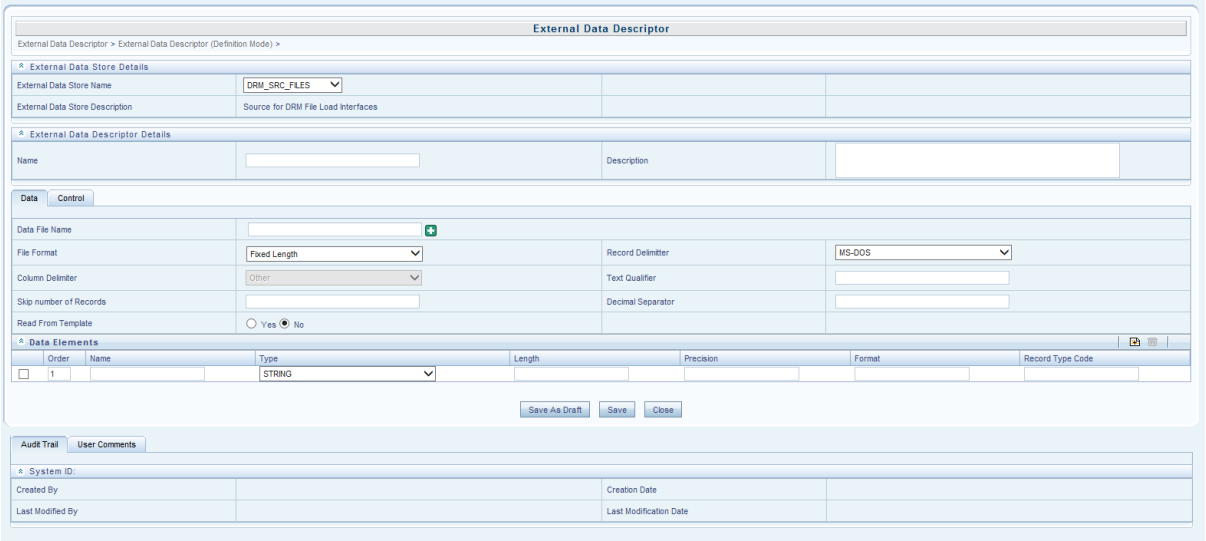
Procedure to create an EDD

To create a new EDD from the *External Data Descriptor* window,

1. Click the Add  button from the tool bar. The *External Data Descriptor Specifications* window appears. This screen is divided into two sections, Source Details and External Data Descriptor Specification Details.
2. In the Source Details section, select the Source Code from the drop down list. The Source Code is the Source you had created. In this example it is,*USG_FILE_SRC*. The values in creating an external data store example is used. The description comes up automatically.

NOTE:

- The fields in the External Data Descriptor Specification Details section changes as per the Source Code selected.
- For HDFS data, the working date format is `yyyy-MM-dd`.
- For Hive table, the working date format is `dd-MON-yyyy`.
- For Sybase source database, date data type is not supported. It has to be timestamp.
- For file extract, SDI date format is not supported. By default it produces timestamp. If a particular format is required then derived column has to be defined with specific format in connector.



3. In the *External Data Descriptor Specification Details* tab, enter the Name and description.

   For example:

   Name: USG_FILE_EDD. This must be in uppercase.

   Description: Term deposit data from USG source on daily basis.

   In the Source Data tab, enter the Data File Name. In order to add more files to the existing EDD, click the + symbol against the Data File Name field.

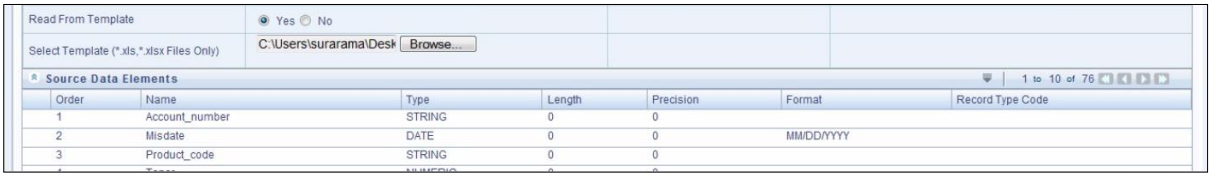
   Example: td_contracts%#MISDATE%_1.csv

   td_contracts%#MISDATE%_2.csv

4. Select the File Format, Record Delimiter, and Column Delimiter from the drop down list.
5. Enter the Text Qualifier, Skip number of Records, and Decimal Separator fields.

6. For the 'Read from Template' option, on selecting **Yes**, you can browse the required template (only in .xls and .xlsx format). If the template is not available, you need to create it manually by clicking the Add button, under *Source Data Elements* tab.

   The fields are explained as tabulated in the Fields and their Description section.



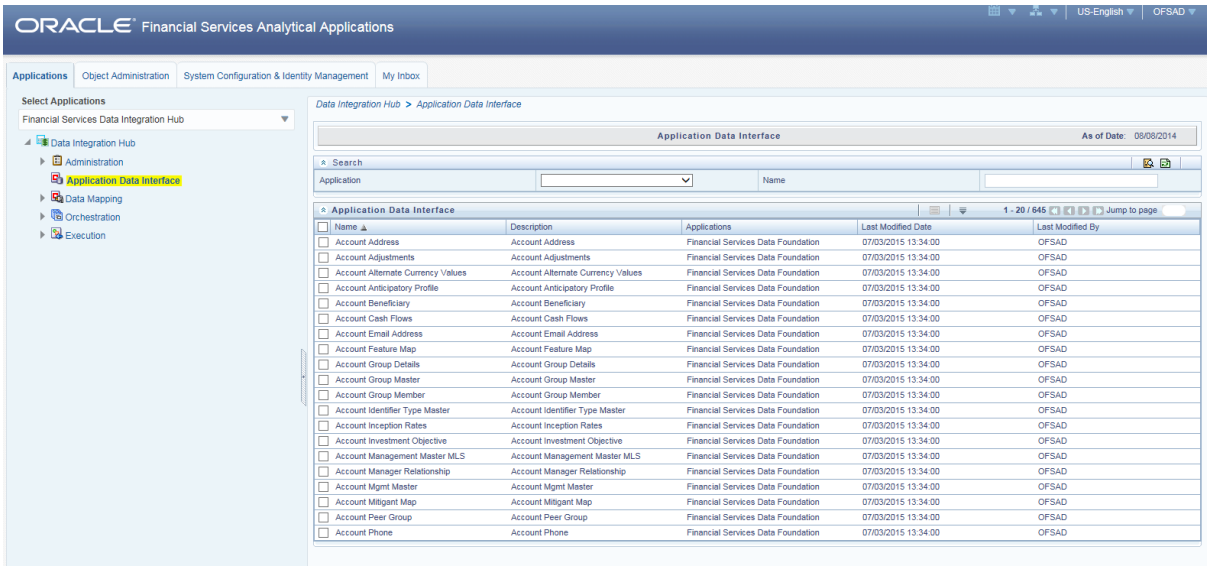
7. If data needs to be reconciled post loading, then control has to be defined for the EDD by clicking on Control Tab. In this version, only Number of records control is possible.

8. Click **Save**.

7 Setting ADI

Application Data Interface data comes pre-seeded based on the application that is installed.

Application Data Interface enables you to view the logical definition of OFSAA physical entities of staging and Result area. This can be viewed by selecting application and its subtype.

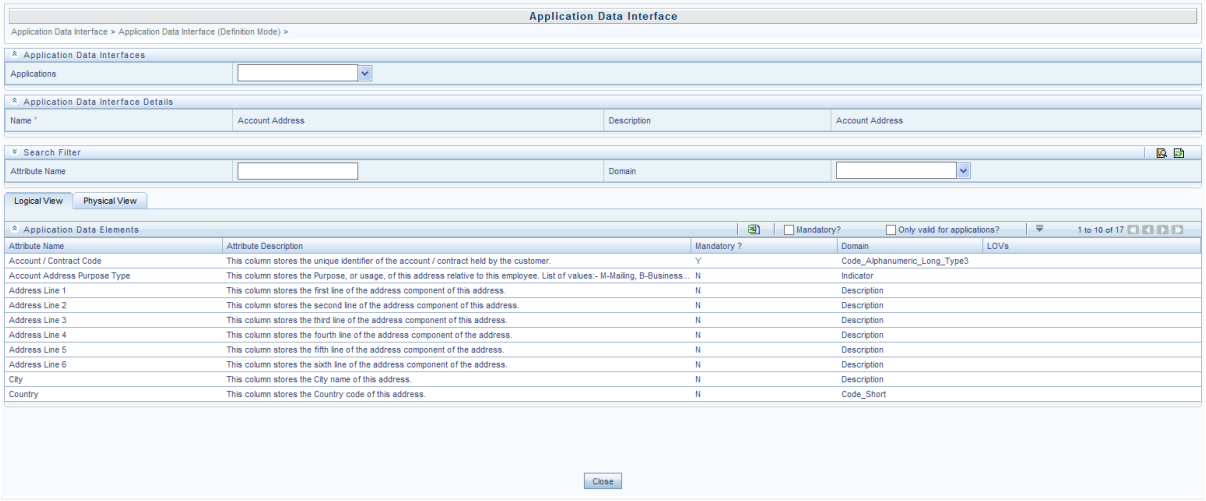


This chapter has a section named viewing an Application Data Interface.

7.1 Viewing an Application Data Interface

You can view an Application Data Interface at any given point. To view an OFSAA Data Interface:

1. Select the **Application** from the drop down list and click on search icon.
2. Select the checkbox adjacent to the Application Data Interface name.
3. Click the View button from the tool bar. The *Application Data Interface* window appears.



4. Select one or more Application(s) for the given ADI. The list of applicable attributes is mentioned below under Application Data Elements tab.
5. Depending on the ADI selected, there may or may not be additional subtype filters. Such as, for Transactions: Customer Account, there is a Product Class list as subtype filters available. You can choose one or more Product class to filter the attributes listed below.
6. Depending on the Application selected, there will be additional filters such as Jurisdiction and Approach for BASEL application.

   The Search panel helps to search the attributes based on Attribute name, domain and Staging and Results area. You can search all the relevant attributes for staging and result area.

   The Application Data Elements section has 2 tabs

   - Logical View
   - Physical View

The Logical view shows all the attributes and its associated description with additional information. For example, if the attribute is mandatory or not for the selected application, its domain and LOV (List of values) that are possible for the particular attribute.

The Physical view tab shows the underlying physical table name of the selected ADI. On selecting the physical table name, it shows the mapping between the logical attribute name and its corresponding physical column name.

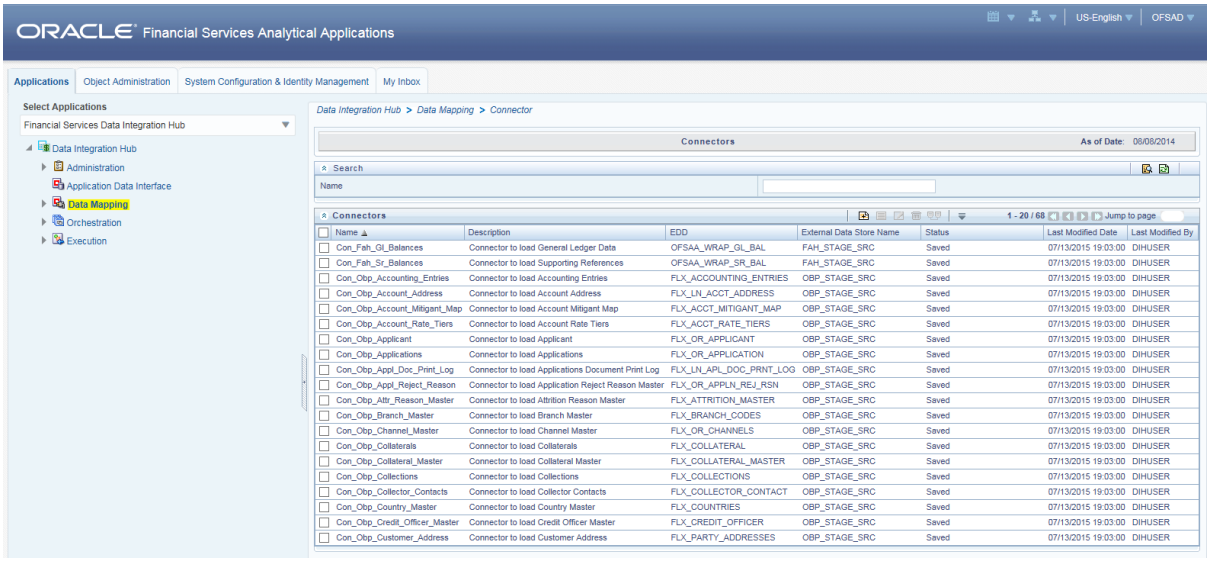
In case of ADI with subtype such as Customer Account, the physical table name is based on the Subtype. Hence, one or more physical table names may appear

8 Mapping an EDD to ADI

Connector allows mapping one or more External Data Descriptor with Application Data Interface. It allows mapping of one or more ADI with EDD as well, in case of extract type connector.

There are pre-built Connectors for Oracle applications like, Flexcube, Oracle Banking Platform, and so on. For other applications you need to define Connectors for your EDDs.

Click the **Data Mapping** option on the LHS and select **Connector.** The *Connector* window appears in the RHS.



This chapter has the following sections:

8.1 Creating a Connector

To create a connector the following information should be decided at the beginning. Based on the options the connector functionality varies.

Fields	Description
Fields marked in red asterisk(*) are mandatory	
Which operation should this connector perform on OFSAA?	This option decides whether the connector will load data into OFSAA or extract data out of OFSAA. This field has two options: • Insert Data – Select this option when connector should load data into OFSAA. This is the default option. • Extract data – Select this option when the connector needs to extract data.
On which OFSAA module should this operation be performed?	This field has two options: • Staging – When data needs to be loaded or extracted into/from the staging area. • Results – When data needs to be loaded or extracted into/from the Result area.
For which application (if any) should this connector be mapped?	This is an optional choice to filter source/targets at the beginning.
For which External Data stores (if any) should the connector be mapped?	This is an optional choice to filter source/targets at the beginning.

While creating a Connector, the fields that are displayed are explained as tabulated.

Fields and their descriptions

In the Connector Details tab:

Fields	Description
Fields marked in red asterisk(*) are mandatory	
Connector Name	The name of the Connector to be created. Example: USG File Connector 1
Connector Description	A description for the Connector is to be mentioned.

In the Target tab:

Fields	Description
Fields marked in red asterisk(*) are mandatory	
Application	Select the application from the drop down list. Example: FSDF
Application Data Interface	Select the Application Data Interface from the drop down list. Example: Customer Account
Sub Type	Select the Product Class from the drop down list. Example: Term Deposit

In the Properties tab:

The properties by default have some values. For every connector, it needs to be reviewed. Generally, no change is required. If the change is required for every connector then, the default value can be changed in the FSI_DIH_OPTIONS_B table.

The columns name is DEF_VAL_ID and DEF_TEXT_VAL.

Fields	Description
Fields marked in red asterisk(*) are mandatory	
Loading mechanism	This option is only applicable for ASCII file source type EDD. This has two options: • External Table • SQLLDR Note: If the loading mechanism is selected as External Table, the file should be located in the same place as the database server. Please provide CREATE DIRECTORY role to the atomic schema. And the path/folder used in the directory should be having read, write permission.
DIRECT	Direct path load of SQLLDR. Values can be True and False. By default, it is set as TRUE. Only applicable for ASCII/Text File source type EDD.
Parallel	Parallel option. True means the loading happens with parallel option and False means it happens in sequential way.
Degree of Parallel	Decides the degree of parallelism.

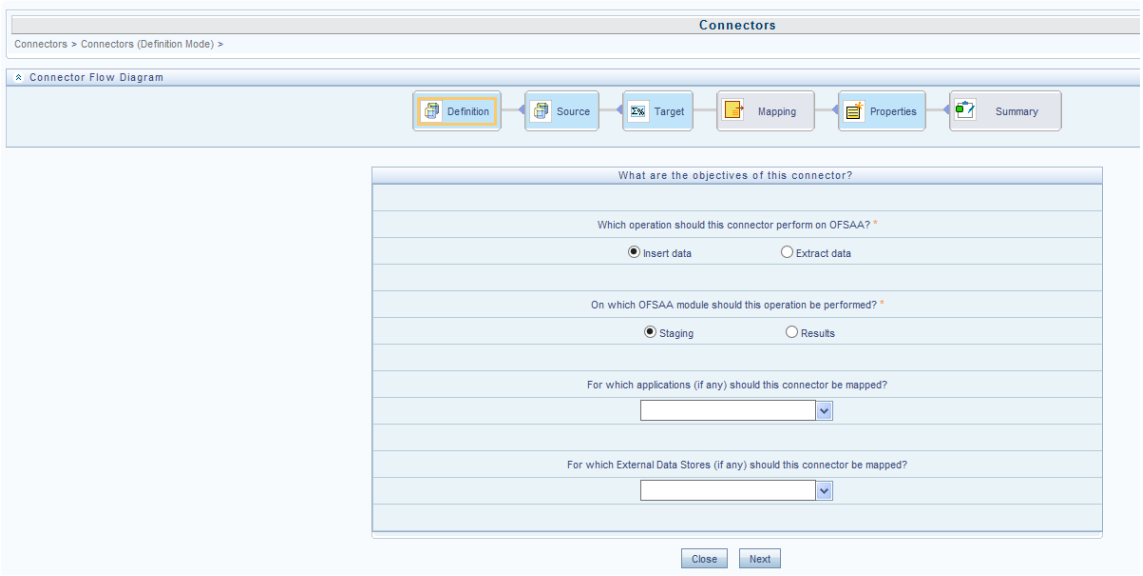
Fields	Description
Fields marked in red asterisk(*) are mandatory	
No: of Errors	Shows the number of errors allowed for the SQLLDR and External Table to proceed. By default it is 0. That means single record fails the job fails.
Maximum Discard	Discarded records allowed for SQLLDR. This needs to be set to a very high number when using multiple subtypes under an ADI. That means it is multiple targets.
ODI FOLDER	By default the value is same as whatever is set under Settings page. The value decides under which folder in ODI repository, the connector will be published. If user needs to have a different folder name than whatever is already set in the settings page, then this value has to be edited before publish.
XML date Format	In this field, you can define the format of the XML Date. Example: MMDDYYYY.
Avoid Partition Exchange	It has 2 values 'Yes' and 'No'. By default it is set as 'No'. The parameter decides if the user wants to avoid Partition Exchange option or not. The partition exchange is used if the target table has partition and the connector execution goes via temp table. Sometimes because of some unknown reason, if the connector execution fails due to partition exchange option, then user has an opportunity to avoid partition exchange by selecting "Yes" in the drop down list.
Do you want to use Data Pump?	There are two values 'Yes' and 'No'. By default it is set as 'Yes'. If the value is 'Yes', it indicates that the Oracle Database source is loaded into OFSAA using the Data Pump method. Alternatively, the standard way of using DBLink method is followed. **Note**: Provide the following access is required for data pump option. - Grant create any directory to Source schema - Grant create any directory to target schema - Grant execute on DBMS_FILE_TRANSFER to target schema - Grant execute on utl_file to source schema

Fields	Description
Fields marked in red asterisk(*) are mandatory	
Source and Target in Same Environment?	This parameter is used only if Data Pump is used. If the value is 'Yes' then file transfer step is not performed during loading. Alternatively, it will transfer file from source to target folder using DBLink.
Source Dump Location	Specify a folder/path that is accessible by Source Oracle Database to create the dump file.
Target Dump Location	Specify a folder/path that is accessible by Target Oracle Database to read the dump file.
Number of Splits for Dump File	Specify a number to transfer the files in parallel chunks. For example, three, indicates that dump file will be split in three and transferred separately. This is to improve performance of file transfer.
Effective Dated Key for Result Area?	This parameter is used for loading data into the result area. It decides whether it will perform lookup into dimension for latest record, or effective dated record. To get the latest record, it appends f_latest_record_indicator ='Y' and for effective dated it appends mis_date between d_record_start_date and d_record_end_date. The default value is **No**. That indicates that it will append f_latest_record_indicator='Y'. Note: For extraction data or any date attribute, effective dating does not work. If this parameter is selected as "Yes" and any date field needs to be loaded into Extraction date or any other date field, then do not rely on surrogate key generation. Use derived column and enter the value in the format "YYYYMMDD" as a number.
Do you want to use DBLink?	This parameter is used to specify the source database connection method. There are two values 'Yes' and 'No'. By default it is set as 'No' If the value is 'Yes', it indicates connection source database will be created using DBLink method If the value is 'No' , it indicates connection to source database will be created using JDBC url
Hive Date Format	This parameter is used to specify the date format for date columns in Hive source. the default value will be 'yyyy-MM-dd'

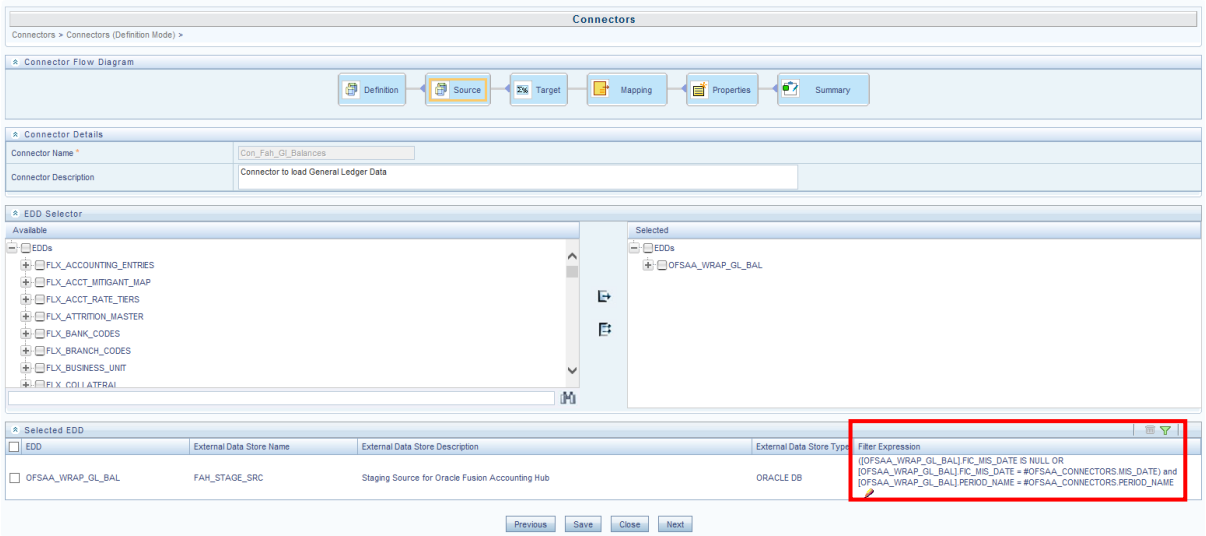
8.1.1 Creating a Connector for Loading Data into Staging

To create a Connector for loading data into staging follow the below steps:

1. Navigate to **Data Mapping** > **Connector.**
2. Click the Add button from the tool bar. The *Connectors Definition* window appears.
3. The Definition tab provides the option to load and extract data. The options **Insert data** and **Staging** are selected by default.
4. Click **Next**.



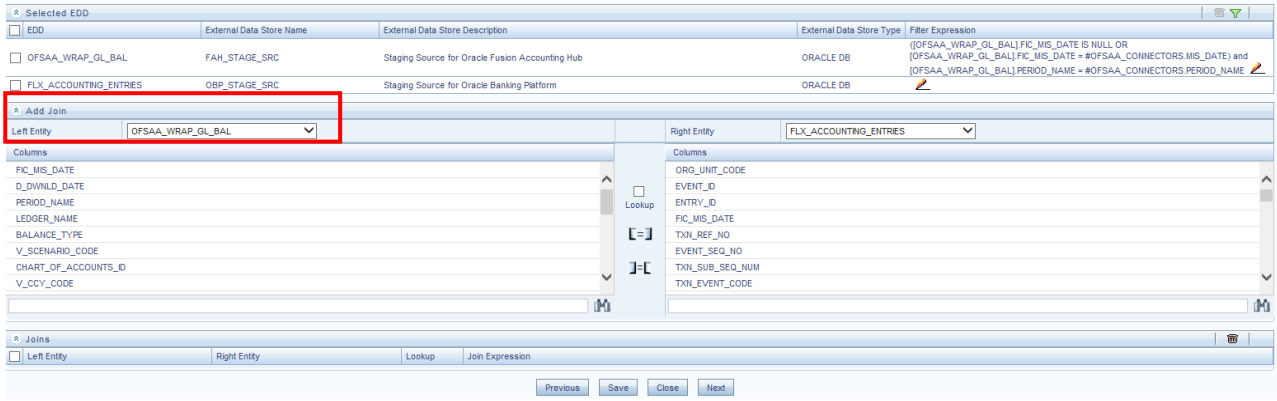
5. The 'Source' block of the flow chart is selected and the respective fields are displayed by default.



Note: When the Insert data option is selected, the EDD becomes source and ADI becomes target. Similarly, if Extract Data option is selected, then ADI becomes source and EDD becomes target.

6. Enter the Connector Name and description. The Fields and details are explained as tabulated in "Fields and their descriptions" section.
7. Under the **EDD Selector** section, the available EDD's are listed. Select the required EDD in order to map it. The selected EDD's are displayed in the 'Source' section as displayed in the above figure.
8. In case multiple EDDs are selected, **Add Join** section appears below with selected EDDs appearing on the left and right side. Select the fields from the Left and right side, and join the fields. If LOOKUP is required, then check the lookup checkbox has during the mapping.

NOTE: If Lookup option is checked, then the join would be left outer join. Else it would be inner join.



9. To edit the filter expression, click  icon. The *Specify Expression* window appears. Select/key in the required expression and click **OK**.

NOTE:

- You do not need to add 'WHERE' clause for the filter.
- For File data loading, use filter expression of Number type along with single quotes. For example: N_DRAWN_AMOUNT ='40000'.
- For Date field refer To_CHAR function for comparison.
- Parameters can also be used in filter expression. Date format must be a valid SQL date format.

For Example:

```
[EDD_GL_DATA].[EXTRACTION_DATE]                                    =
TO_DATE(#DIHDEV.MIS_DATE,'dd-MM-yyyy')
```

10. If Source type is Hive, the filter expressions must conform to the following restrictions:

    - Expression must be valid HiveQL
    - Does not include Oracle built in or user defined functions
    - Does not include Sub queries
    - Includes Hive built in functions only
    - Parameters can also be used in filter expression. MISDATE can also be passed dynamically so that it is loaded from Batch Execution Screen. The date format specified must be valid Hive Date format i.e. yyyy-MM-DD

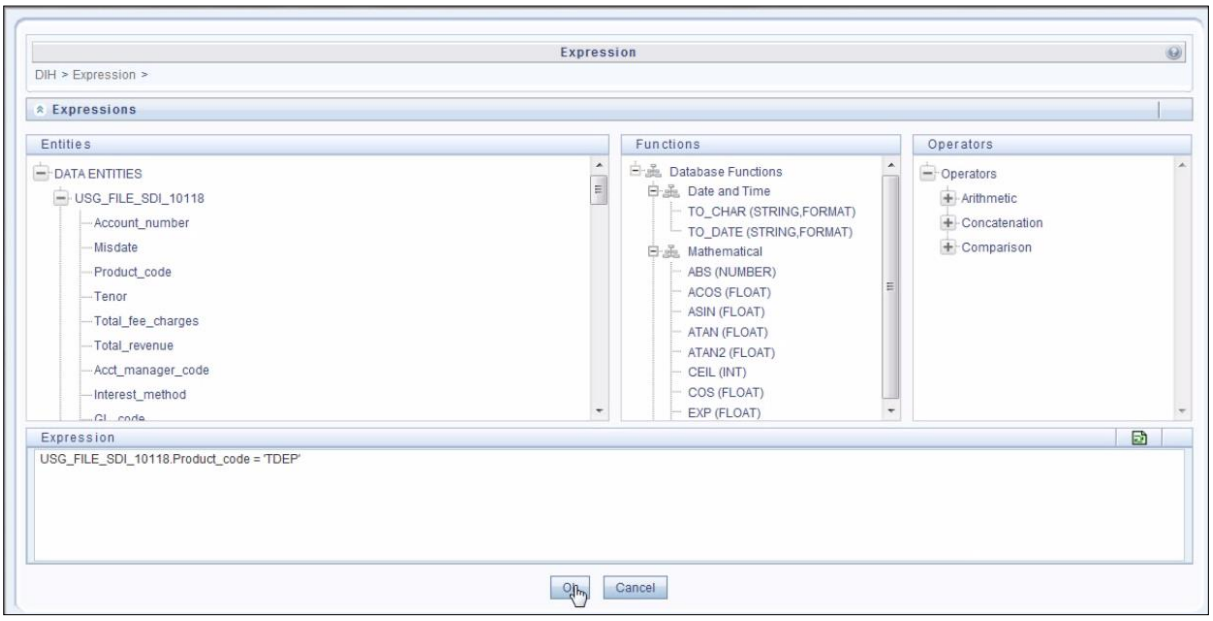
    For Example:

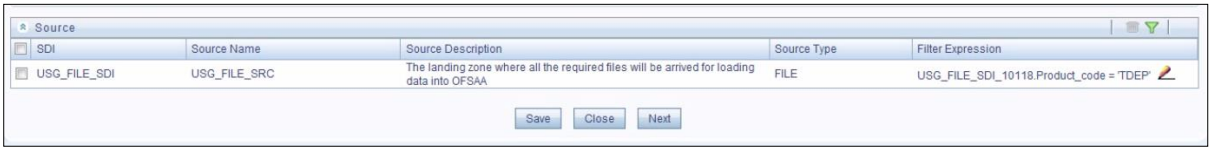
    Filter Expression in Connector:-`[EDD_GL_DATA].[EXTRACTION_DATE] = #DIHDEV.MIS_DATE`

    MISDATE Parameter in Batch:

    `MISDATE=$MISDATE:yyyy-MM-dd`

NOTE: This is only applicable if the patch 8.0.2.1.4 (Bug - 24487929) is applied.



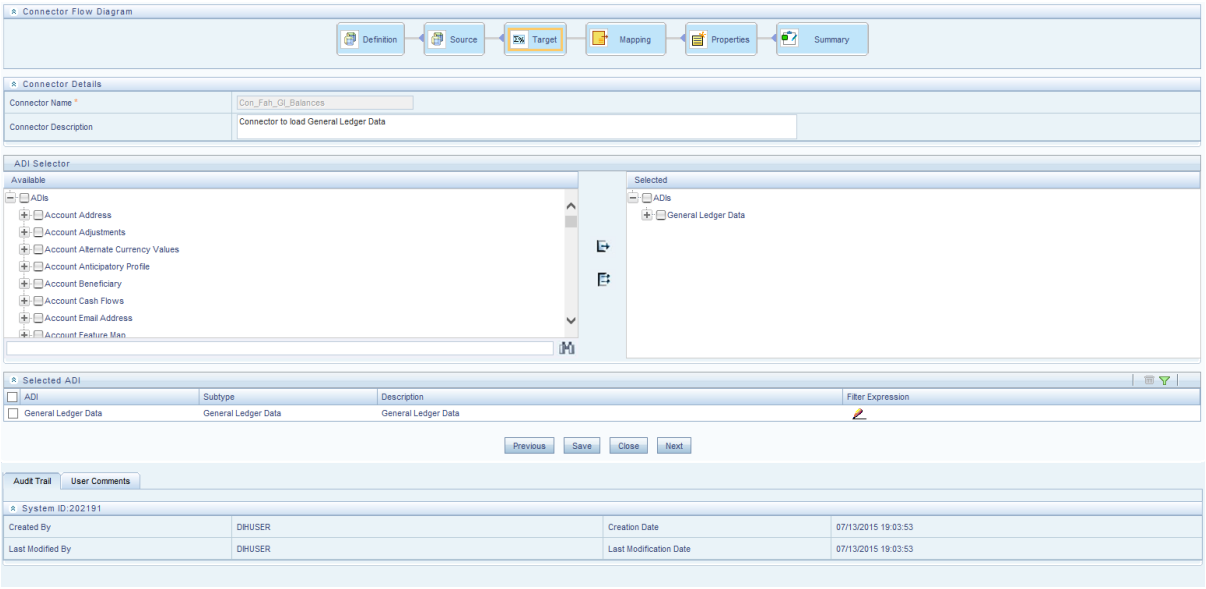


11. Click **Next.** The 'Target' block appears.
12. Select Application Data Interface from the available list in the left panel. Import that to right side of the panel. The **Selected ADI** grid shows the selected ADIs along with

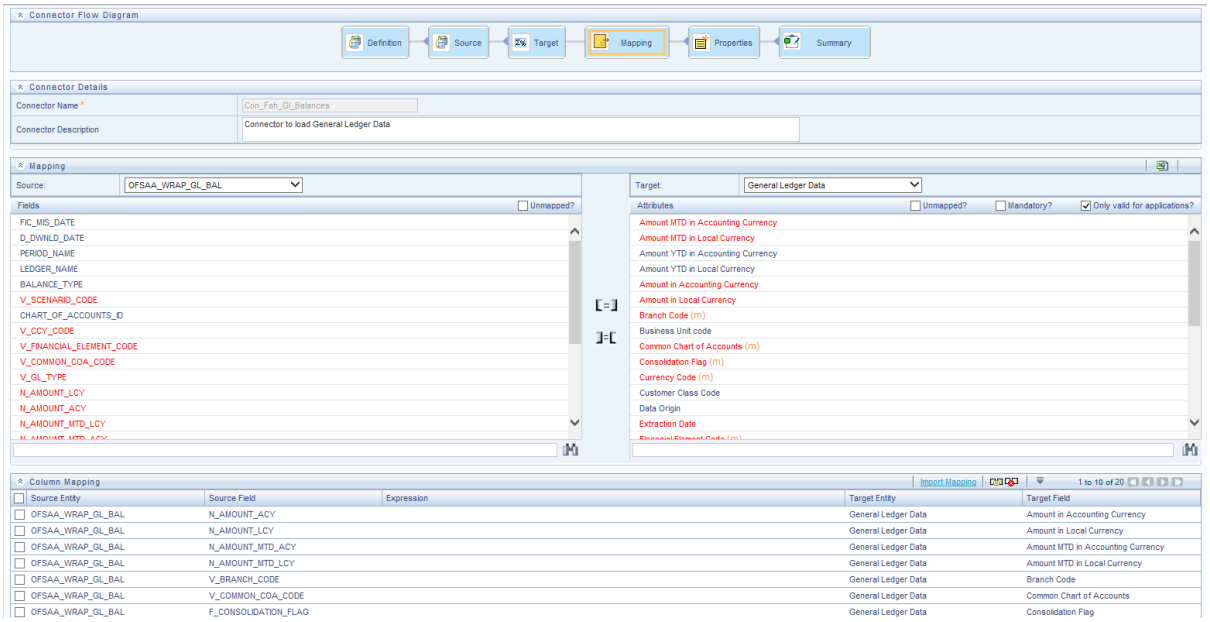
filter expression. The filter on target can be applied by clicking the filter expression. The Fields and details are explained as tabulated in "Fields and their descriptions" section.

Note: If multiple sub types/ADI is selected, then the filter expression is necessary to direct required data into the correct entity/table.

This is required to identify which record goes into which target. It will be an expression such as <EDDNAME>.<Field Name> = 'TD' under Term deposit and <EDDNAME>.<Field Name> = 'CARDS' under Cards.



13. Click the **Mapping** block in the flow chart, to map the EDD's. For details on Auto-mapping refer to section Auto-mapping.



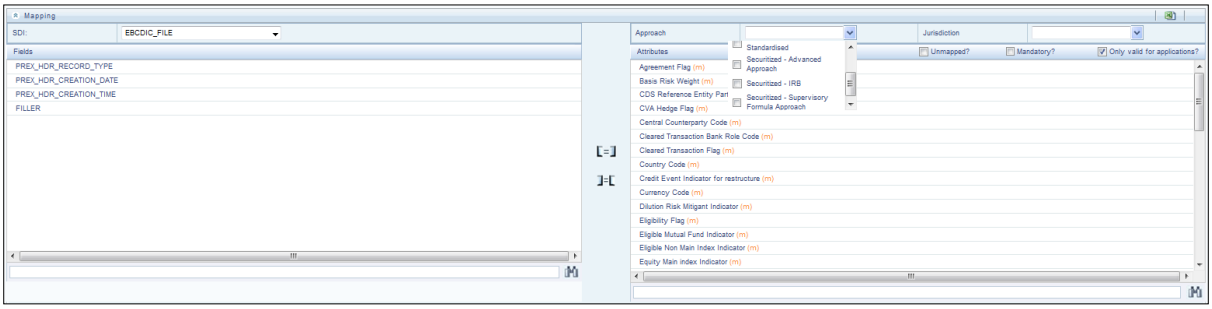
14. Select the EDD from the drop down list. For details on drop-down options for EDD refer to section Options in Mapping EDD.

15. Select the 'Mandatory' option to populate the mandatory Attributes to be mapped.

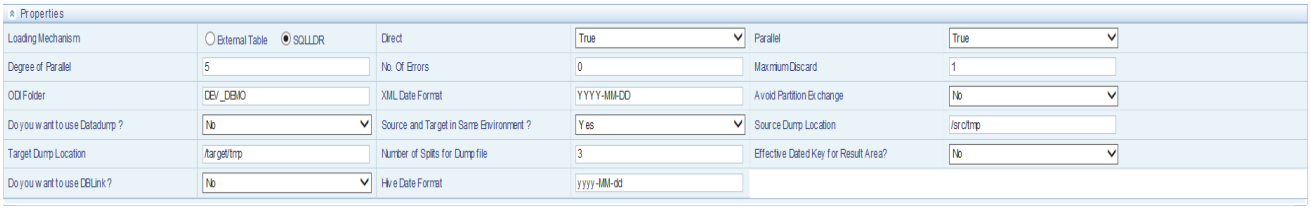
    Note: On selecting 'Derived Column' option as EDD from the drop down list, you can add an expression.

    All the mapped attributes are listed under the 'Column Mapping' sector. For details on the options in the Mapping grid such as Auto-mapping, Export to Excel and so on; refer to section Fields in Mapping.

Note: If the application is chosen as BASEL, while mapping, you can select the type of approach (such as, IRB or Advanced Approach) and the Jurisdiction (BIS, BRAZIL, China, India and so on) for filtering.

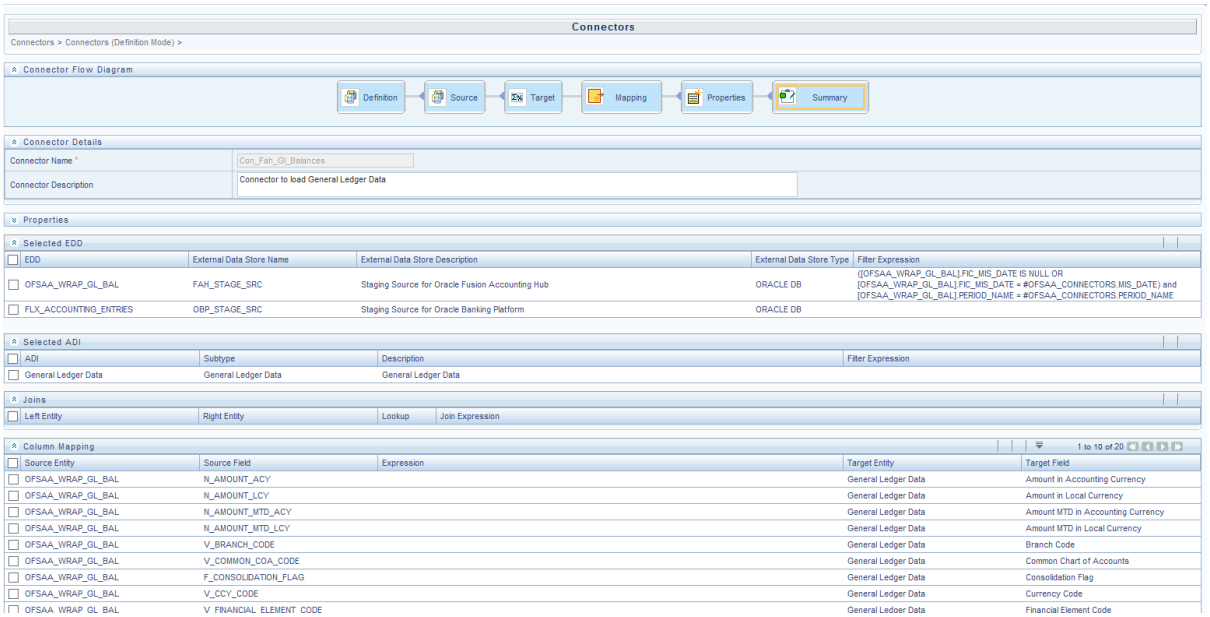


16. Click the **Properties** block in the flow chart. The Properties related fields and the Connector details appear automatically.



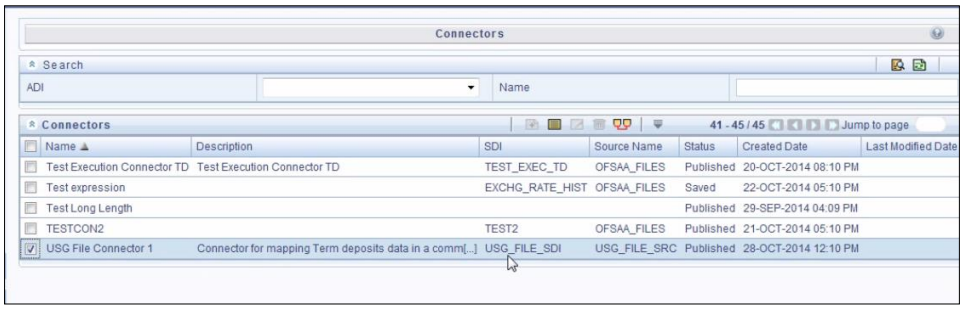
17. Select the Loading Mechanism. Select the Direct and Parallel option from the drop down list. Key in the value for Degree of Parallel, No: of Errors, Maximum Discard, ODI Folder, XML Date Format, Do you want to use Data Pump?, Source and Target in Same Environment?, Avoid Partition Exchange, Source Dump Location, Target Dump Location, Number of Splits for Dump File, Hive Date Format, Effective Dated Key for Result Area? and Do you want to use DBLink?. The Fields and details are explained as tabulated in "Fields and their descriptions" section.

18. Click the **Summary** block in the flow chart to view the summary of all sections.



19. Click **Publish**. This converts DIH metadata into ODI objects.

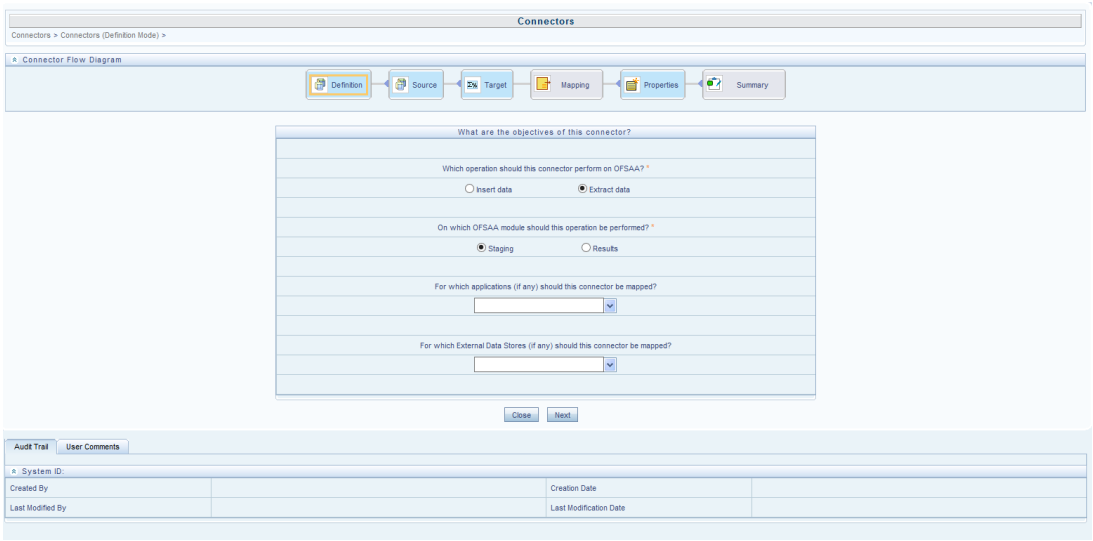
The Audit Trail section at the bottom of the window displays the information of the source created. The User comments section facilitates you to add or update additional information as comments.



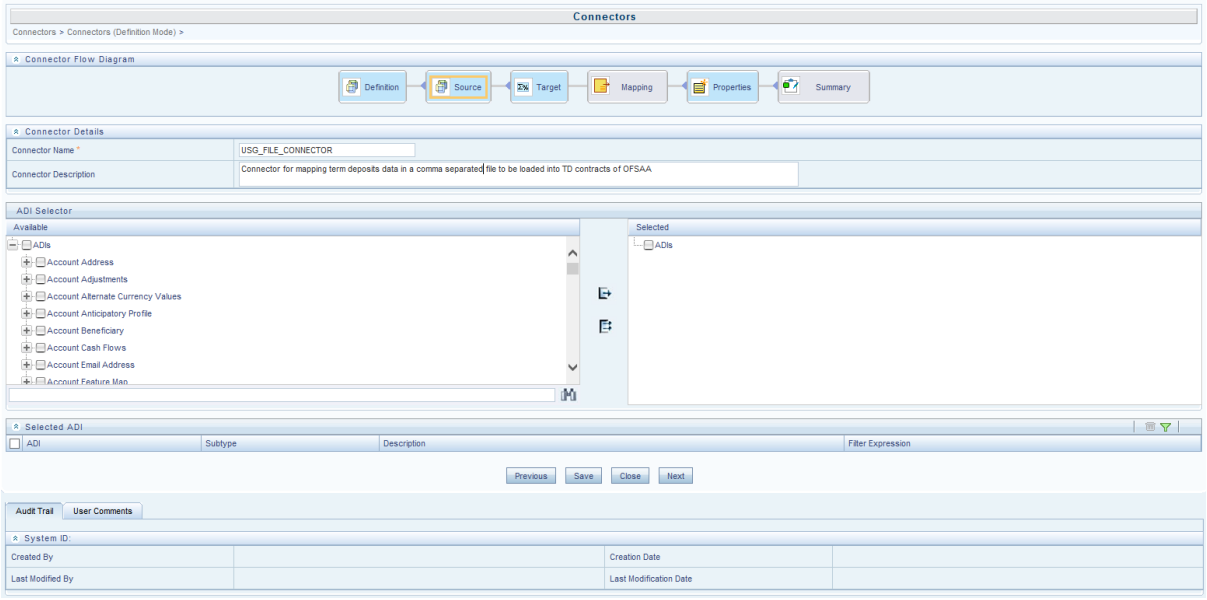
8.1.2 Creating a Connector for Extracting Data from Staging

To create a Connector for extracting data from staging follow the below steps:

1. Navigate to **Data Mapping** > **Connector.**
2. Click the Add button from the tool bar. The *Connectors Definition* window appears.
3. The Definition tab provides the option to load and extract data. The **Extract data** option enables you to extract the data from the OFSAA system. Additionally, you have the option to select whether the loading/extracting should be performed in the **Staging** or **Results** area. Click **Next**.
4. Select Extract data and Staging.



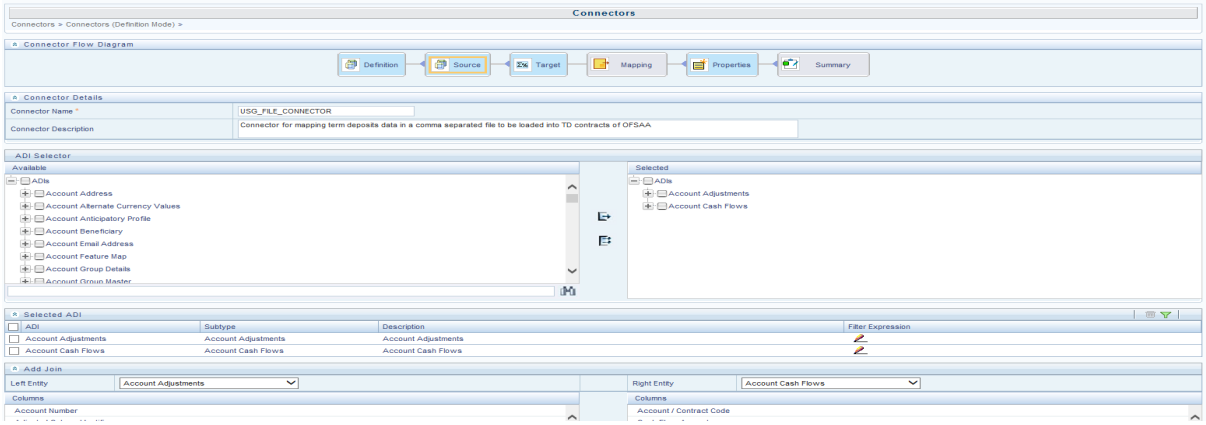
5. Click **Next.** The **Source** block appears.



Note: When the Extract data option is selected, the ADI becomes the source and EDD becomes target. Only ADIs relevant to staging area are displayed.

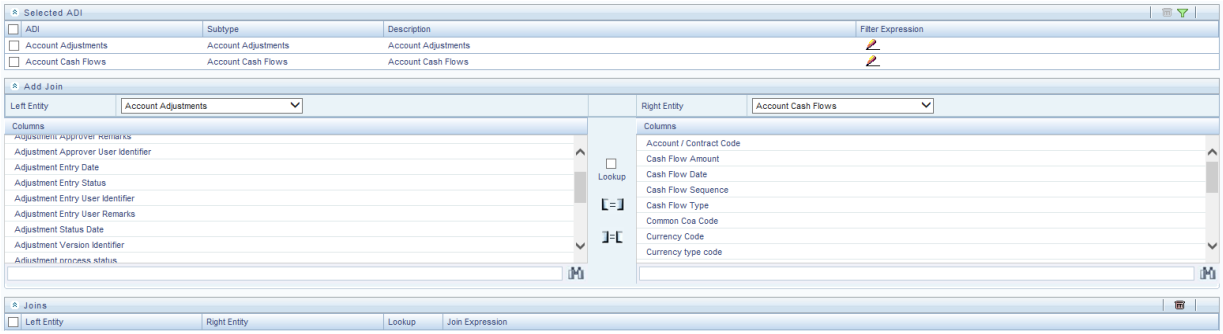
6. Enter the **Connector Name** and **Connector Description**. The Fields and details are explained as tabulated in "Fields and their descriptions" section.

7. Under the **ADI Selector** section, the available ADI's are listed. Select the required ADI in order to map it. The selected ADI's are displayed in the **Source** grid.



8. If multiple ADIs are selected, **Add Join** section appears below with selected EDDs appearing on the left and right side. Select the fields from the Left and right side, and join the fields. If LOOKUP is required, then check the lookup checkbox has during the mapping.

NOTE: If Lookup option is checked, then the join would be left outer join. Else it would be inner join.



9. To edit the filter expression, click  icon. The *Specify Expression* window appears. Select/key in the required expression and click **OK**.

NOTE:

- You do not need to add 'WHERE' clause for the filter.
- For File data loading, use filter expression of Number type along with single quotes. For example: N_DRAWN_AMOUNT ='40000'.
- For Date field refer To_CHAR function for comparison.
- Parameters can also be used in filter expression. Date format must be a valid SQL date format.

For Example:

```
[EDD_GL_DATA].[EXTRACTION_DATE]                                    =
TO_DATE(#DIHDEV.MIS_DATE,'dd-MM-yyyy')
```

10. If Source type is Hive, the filter expressions must conform to the following restrictions:

    - Expression must be valid HiveQL
    - Does not include Oracle built in or user defined functions
    - Does not include Sub queries
    - Includes Hive built in functions only
    - Parameters can also be used in filter expression. MISDATE can also be passed dynamically so that it is loaded from Batch Execution Screen. The date format specified must be valid Hive Date format i.e. yyyy-MM-DD

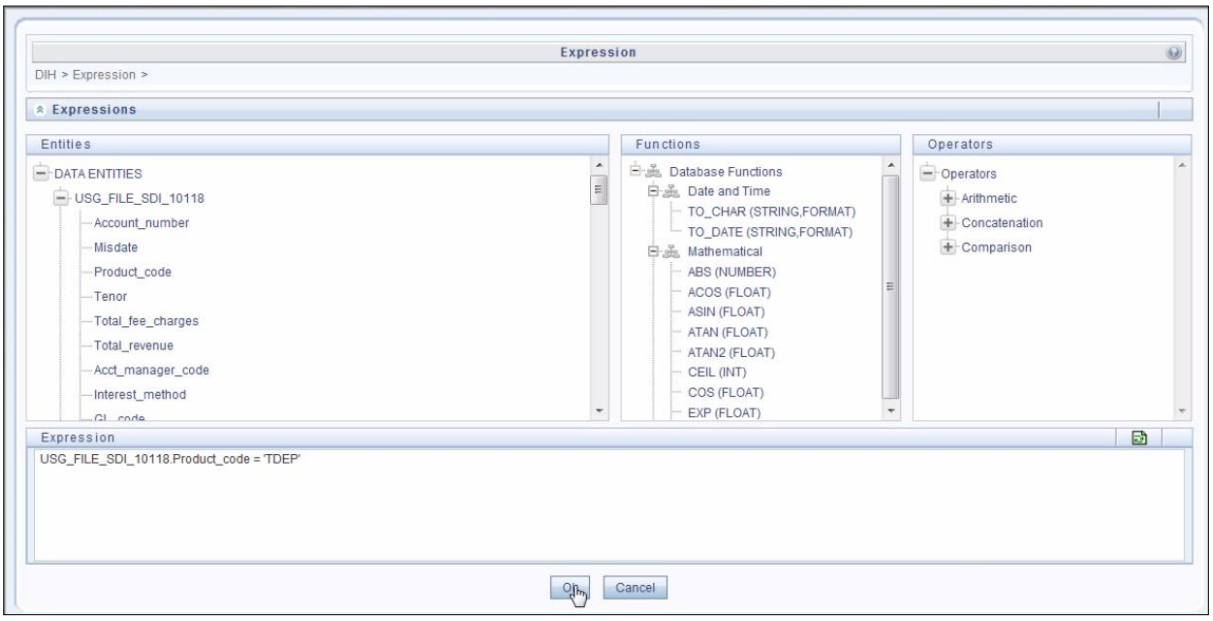
    For Example:

    Filter Expression in Connector:-`[EDD_GL_DATA].[EXTRACTION_DATE] = #DIHDEV.MIS_DATE`

    MISDATE Parameter in Batch:

    `MISDATE=$MISDATE:yyyy-MM-dd`

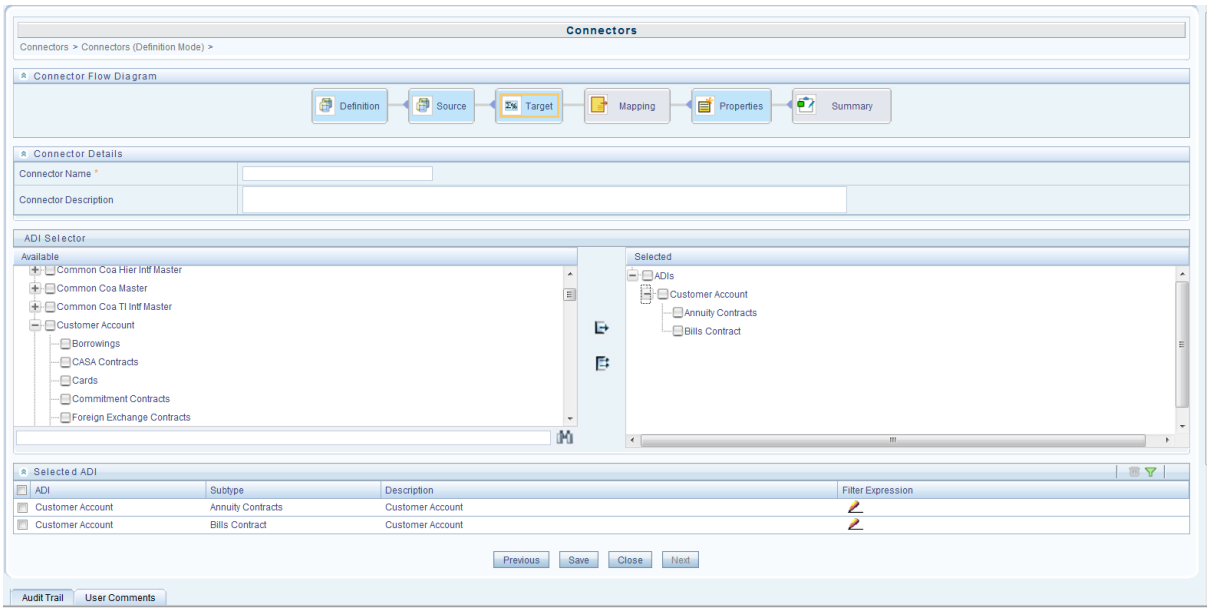
NOTE: This is only applicable if the patch 8.0.2.1.4 (Bug - 24487929) is applied



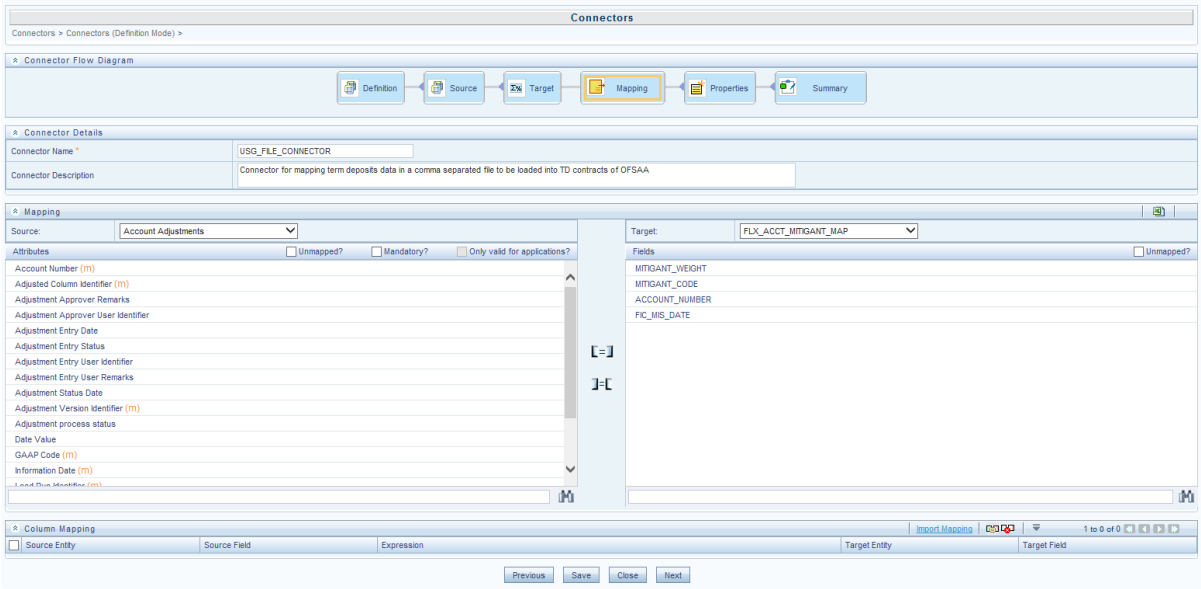
11. Click **Next.** The 'Target' block (in Flow chart) appears.

12. Select External Data Descriptor from available list shows left panel. Move that to right side on selected panel. The Selected EDD grid shows the selected EDDs along with the filter expression. The filter on target can be applied by clicking the filter expression. The Fields and details are explained as tabulated in "Fields and their descriptions" section.

    Note: For extract type of connector, only one EDD should be selected. It does not support multiple EDD.

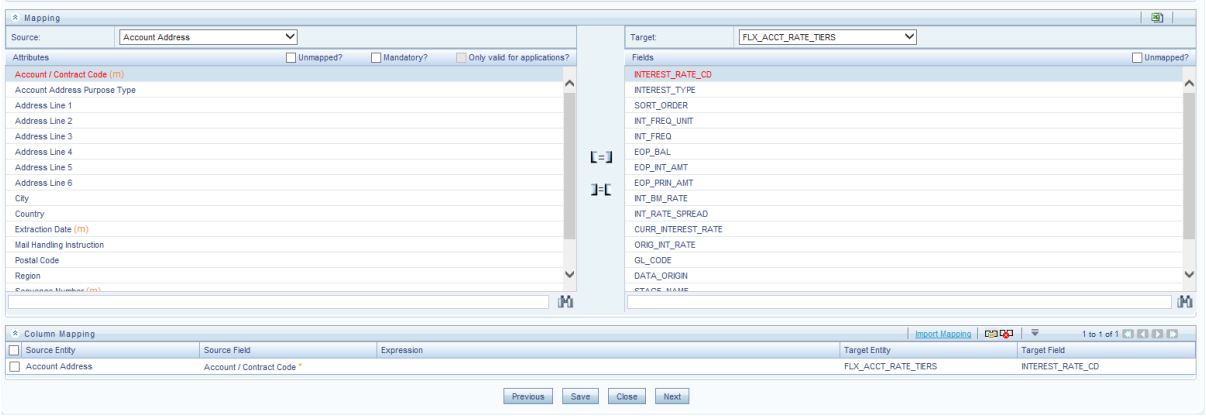


13. Click the **Mapping** block in the flow chart, to map the EDD's. For details on Auto-mapping refer to section Auto-mapping.



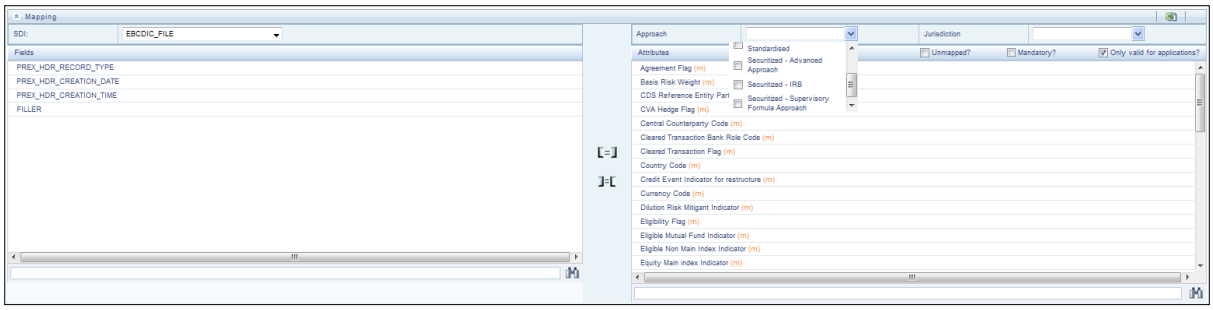
14. Select the ADI from the drop down list. There are 3 options in the drop down list. For details on options for ADI refer to section Options in Mapping ADI.

Note: For extract connector, mandatory attributes are not applicable. There is no validation of mandatory attributes during publish. Mandatory indicator (*) against an attribute for ADI is for reference only. All the mapped attributes is listed under the 'Column Mapping' sector.



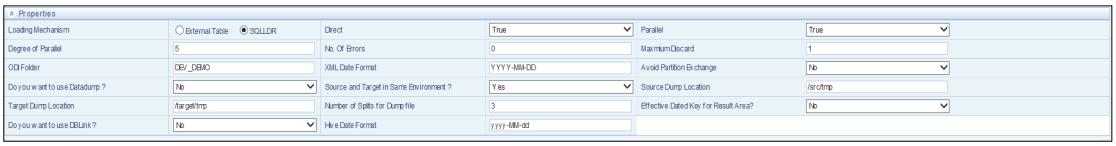
15. Select the ‘Mandatory’ option to populate the mandatory Attributes to be mapped. All the mapped attributes are listed under the ‘Column Mapping’ sector. For details on the options in the Mapping grid such as Auto-mapping, Export to Excel and so on refer to section Fields in Mapping.

Note: If the application is chosen as BASEL, while mapping, you can select the type of approach (such as, IRB or Advanced Approach) and the Jurisdiction (BIS, BRAZIL, China, India and so on) for filtering.



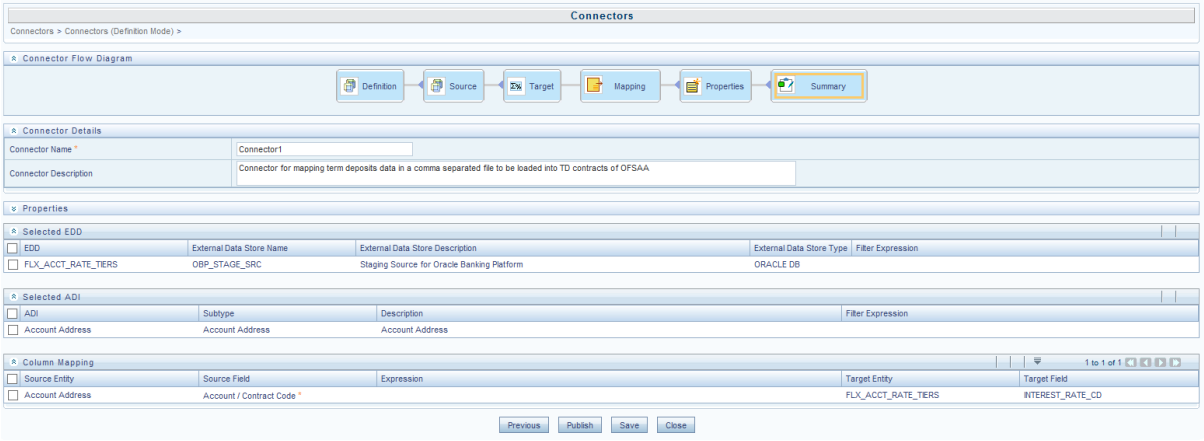
16. Click the **Properties** block in the flow chart.

The Properties related fields appear. The Connector details appear automatically.



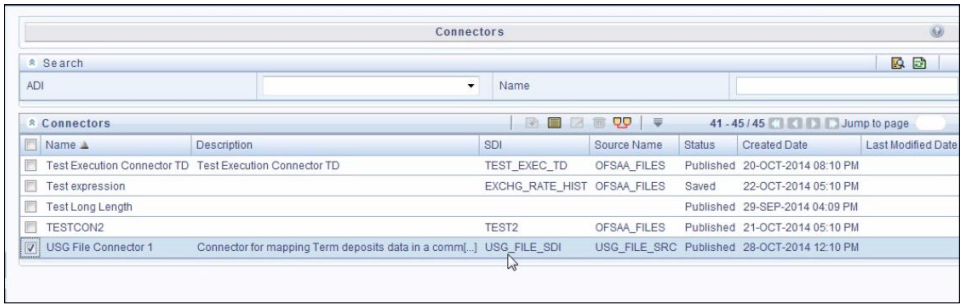
17. No action is required in properties section. Proceed to the summary tab.

18. Click the **Summary** block in the flow chart to view the summary of all sections.



19. Click **Publish**. This converts DIH metadata into ODI objects.

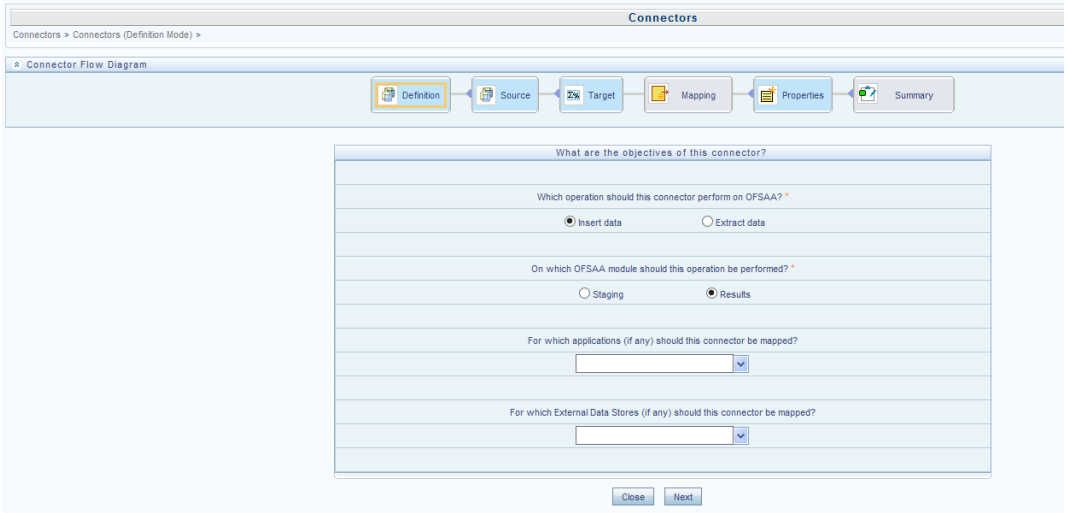
The Audit Trail section at the bottom of the window displays the information of the source created. The User comments section facilitates you to add or update additional information as comments.



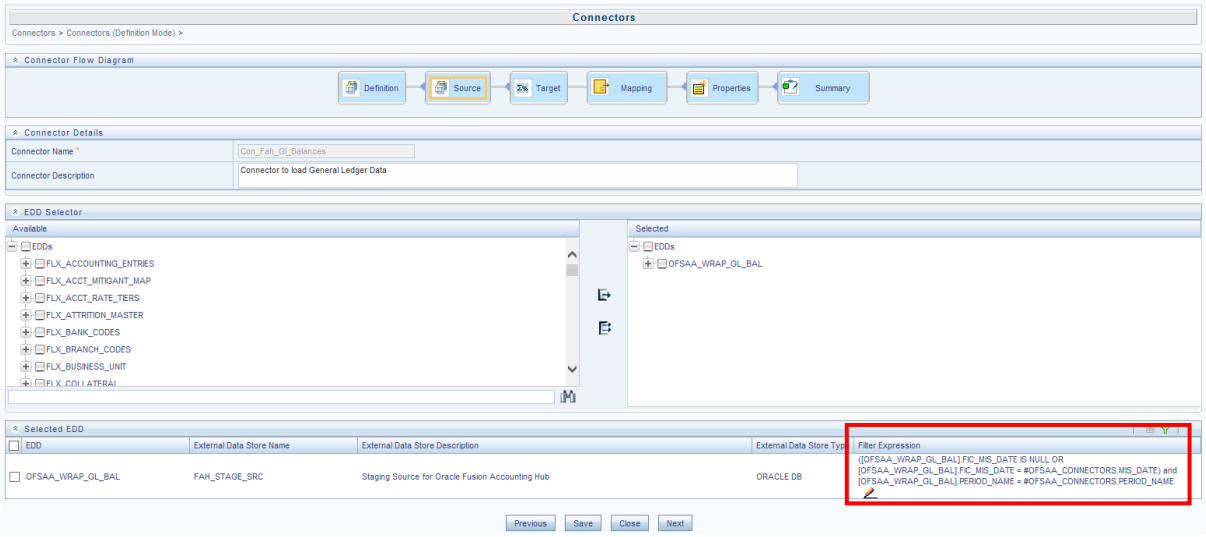
8.1.3 Creating a Connector for Loading Data into Results

To create a Connector for extracting data from staging follow the below steps:

1. Navigate to **Data Mapping** > **Connector**.
2. Click the Add button from the tool bar. The *Connectors Definition* window appears.
3. The Definition tab provides the option to load and extract data. Select **Insert Data** and **Results**. Click **Next**.



4. The **Source** block appears by default, and the respective fields are displayed.



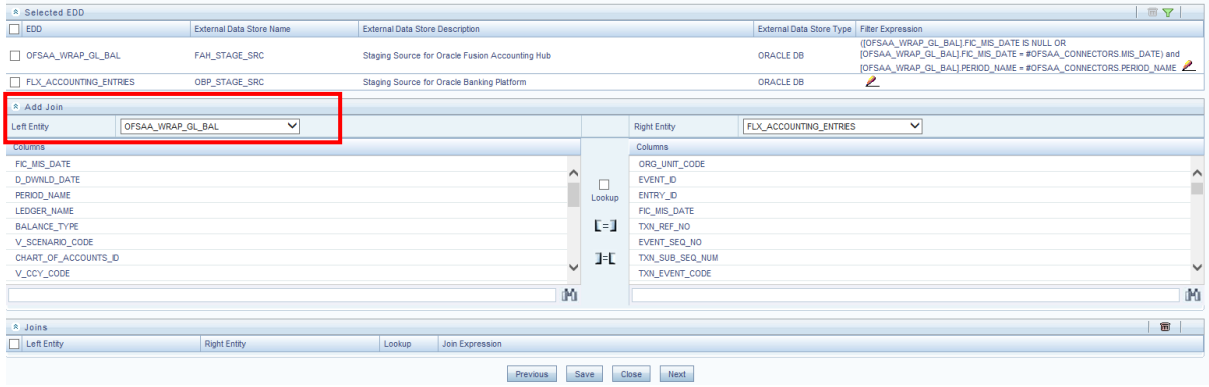
Note: When the Insert data option is selected, the EDD becomes source and ADI becomes target. Similarly, if Extract Data option is selected, then ADI becomes source and EDD becomes target.

5. Enter the Connector Name and description. The Fields and details are explained as tabulated in "Fields and their descriptions" section.

6. Under the 'EDD Selector' section, the available EDD's are listed. Select the required EDD in order to map it. The selected EDD's are displayed in the 'Source' section.

7. In case of multiple EDDs selected. **Add Join** section appears below with selected EDDs appearing on the left and right side. Select the fields from the Left and right side, and join the fields. If LOOKUP is required, then check the lookup checkbox has during the mapping.

NOTE: If Lookup option is checked, then the join would be left outer join. Else it would be inner join.



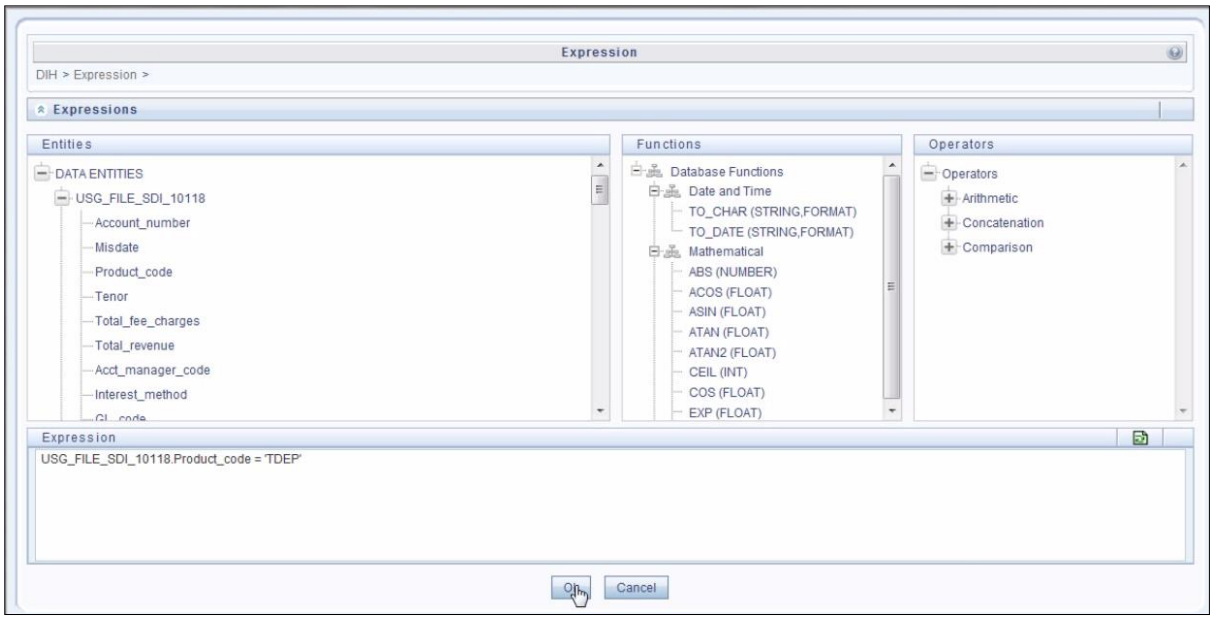
8. To edit the filter expression, click  icon. The *Specify Expression* window appears. Select/key in the required expression and click **OK**.

NOTE:

- You do not need to add 'WHERE' clause for the filter.
- For File data loading, use filter expression of Number type along with single quotes. For example: N_DRAWN_AMOUNT ='40000'.
- For Date field refer To_CHAR function for comparison.
- Parameters can also be used in filter expression. Date format must be a valid SQL date format.

For Example:

```
[EDD_GL_DATA].[EXTRACTION_DATE]                                    =
TO_DATE(#DIHDEV.MIS_DATE,'dd-MM-yyyy')
```



9. If Source type is Hive, the filter expressions must conform to the following restrictions:

   - Expression must be valid HiveQL
   - Does not include Oracle built in or user defined functions
   - Does not include Sub queries
   - Includes Hive built in functions only
   - Parameters can also be used in filter expression. MISDATE can also be passed dynamically so that it is loaded from Batch Execution Screen. The date format specified must be valid Hive Date format i.e. yyyy-MM-DD

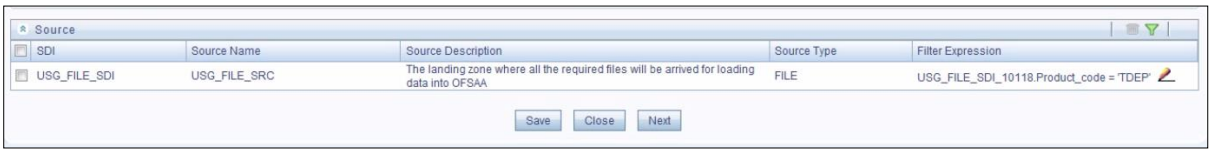
   For Example:

   Filter Expression in Connector:-`[EDD_GL_DATA].[EXTRACTION_DATE] = #DIHDEV.MIS_DATE`

   MISDATE Parameter in Batch:

   `MISDATE=$MISDATE:yyyy-MM-dd`

NOTE: This is only applicable if the patch 8.0.2.1.4 (Bug - 24487929) is applied

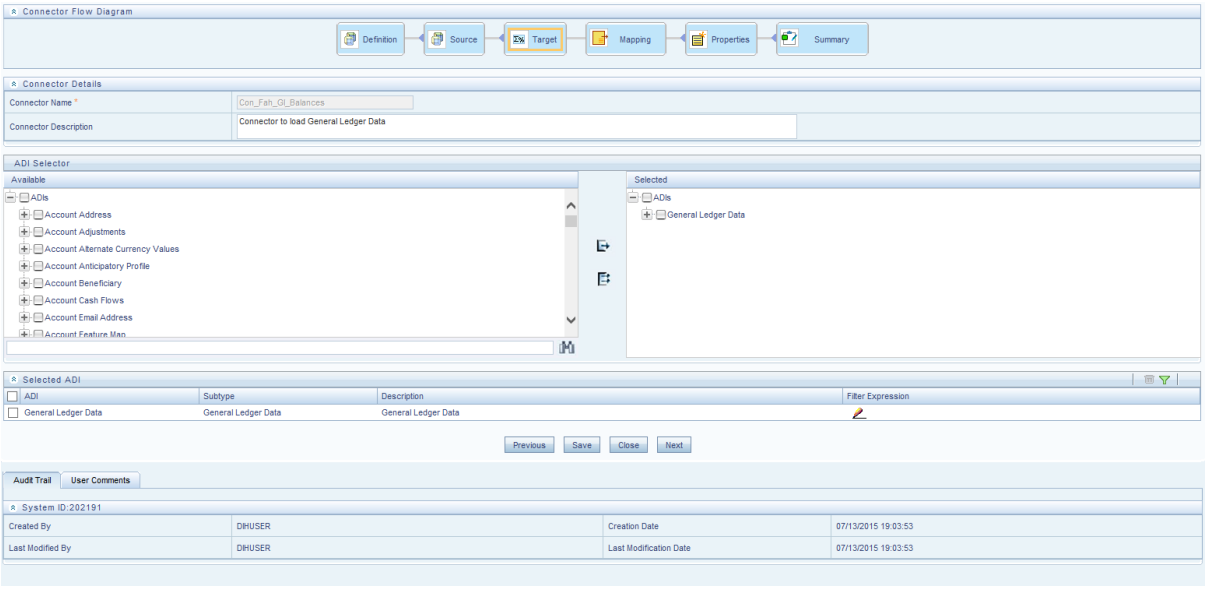


10. Click **Next.** The 'Target' block appears.

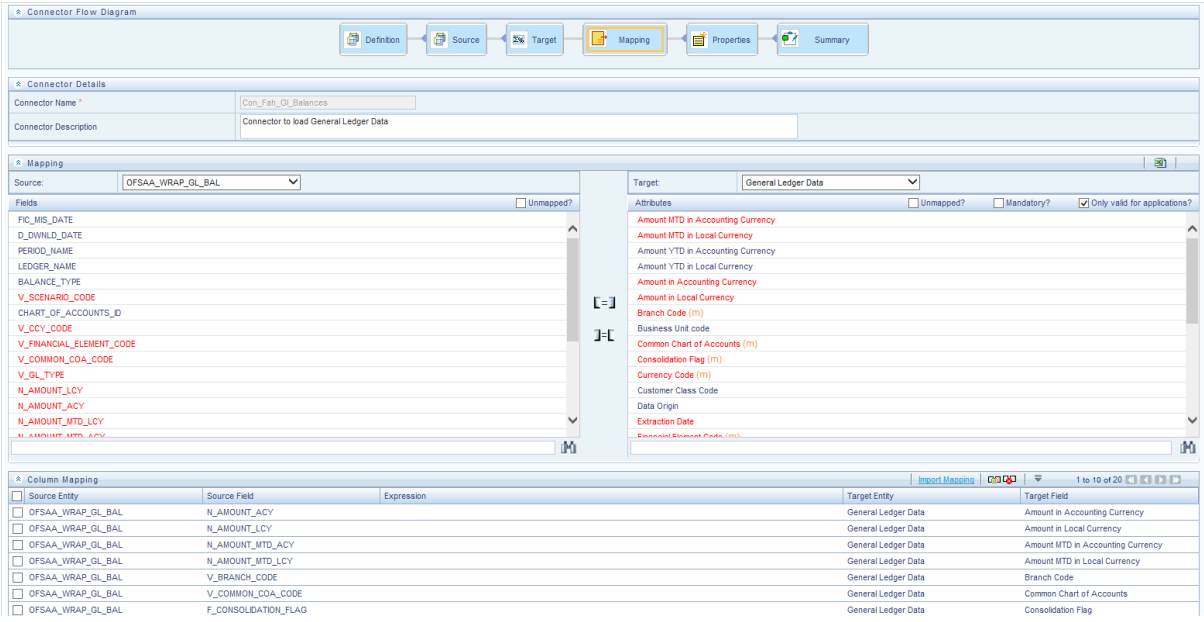
11. Select Application Data Interface from available list shows left panel. Move that to right side on selected panel. Below grid shows the selected ADIs along with filter expression. The filter on target can be applied by clicking the filter expression. The Fields and details are explained as tabulated in "Fields and their descriptions" section.

Note: In case multiple sub types/ADI are selected, the filter expression is necessary to direct required data into right entity/table. If no filter expression chosen then all the records are loaded into all the ADI/subtypes entities/tables.

This is required to identify which record goes into which target. It will be an expression such as <EDDNAME>.<Field Name> = 'TD' under Term deposit and <EDDNAME>.<Field Name> = 'CARDS' under Cards.



12. Click the **Mapping** block in the flow chart, in order to map the EDD's. For details on Auto-mapping refer to section Auto-mapping.

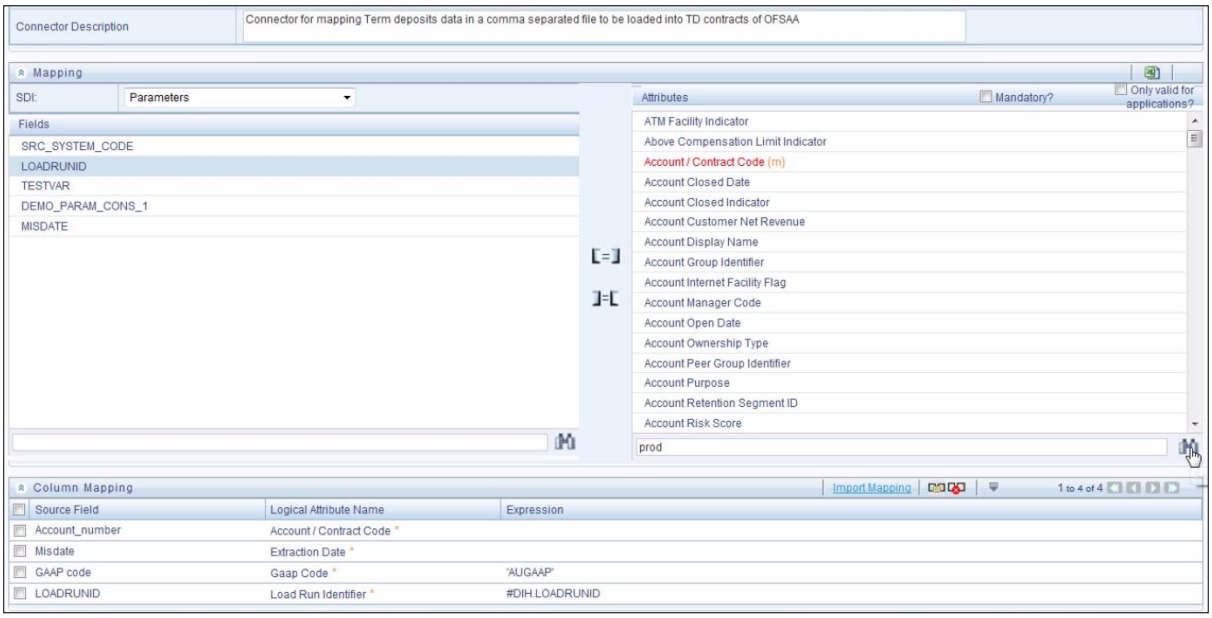


13. Select the EDD from the drop down list. For details on drop-down options for EDD refer to section Options in Mapping EDD.

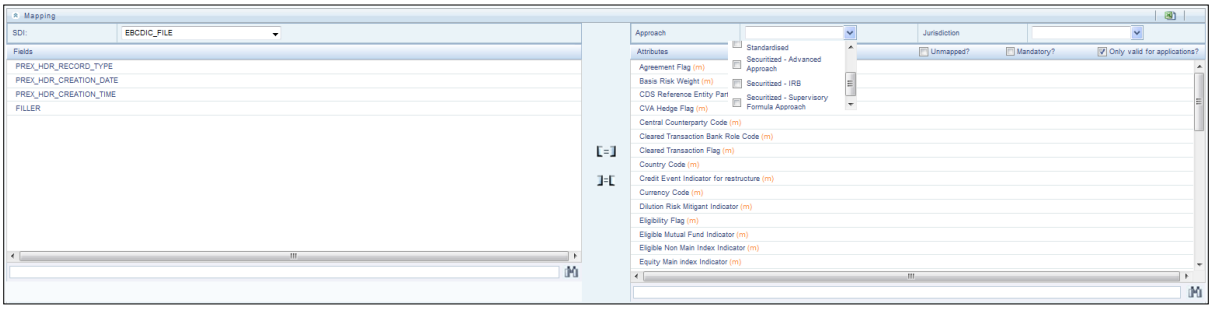
14. Select the 'Mandatory' option to populate the mandatory Attributes to be mapped.

    Note: On selecting 'Derived Column' option as EDD from the drop down list, you can add an expression.

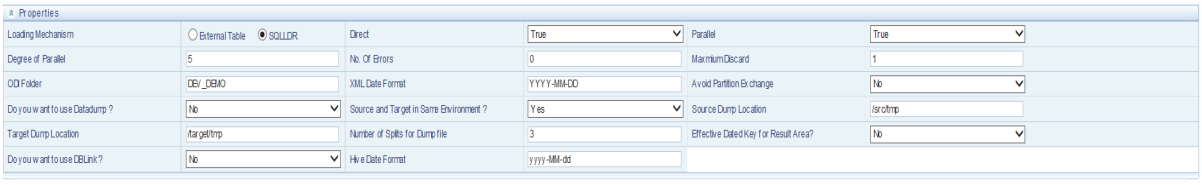
    All the mapped attributes are listed under the 'Column Mapping' sector. For details on the options in the Mapping grid such as Auto-mapping, Export to Excel and so on, refer to section Fields in Mapping.



Note: If the application is chosen as BASEL, while mapping, you can select the type of approach (such as, IRB or Advanced Approach) and the Jurisdiction (BIS, BRAZIL, China, India and so on) for filtering.

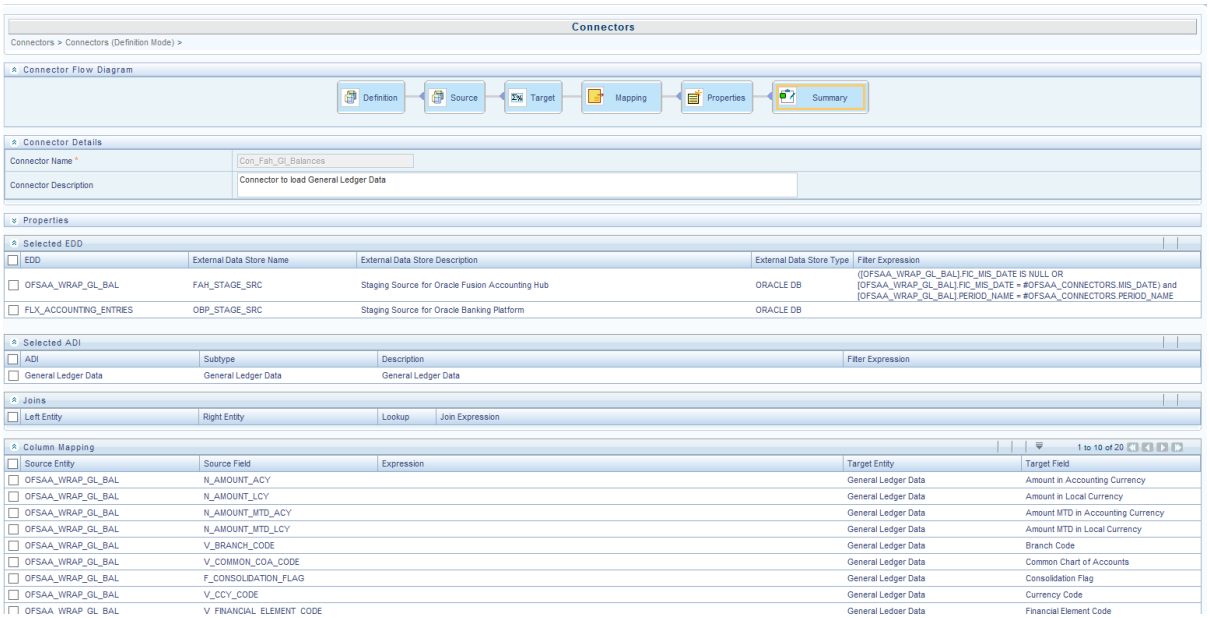


15. Click the **Properties** block in the flow chart. The Properties related fields come up. The Connector details come up automatically.



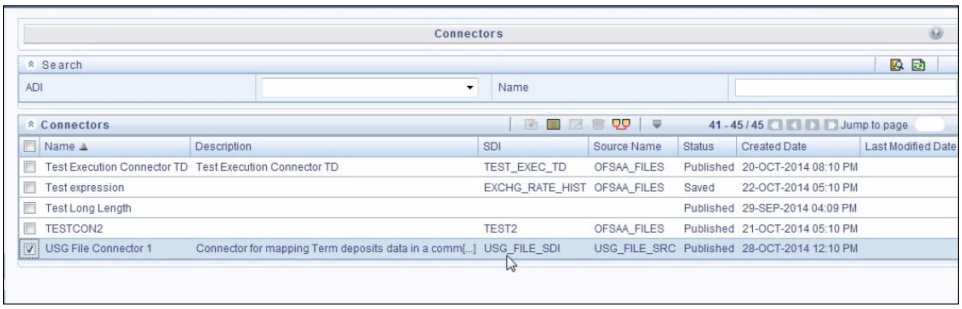
16. Select the Loading Mechanism. Select the Direct and Parallel option from the drop down list. Key in the value for Degree of Parallel, No: of Errors, Maximum Discard, ODI Folder, XML Date Format, Do you want to use Data Pump?, Source and Target in Same Environment?, Avoid Partition Exchange, Source Dump Location, Target Dump Location, and Number of Splits for Dump File, Effective Dated Key for Result Area?. The Fields and details are explained as tabulated in "Fields and their descriptions" section.

17. Click the **Summary** block in the flow chart to view the summary of all sections.



18. Click **Publish**. This converts DIH metadata into ODI objects.

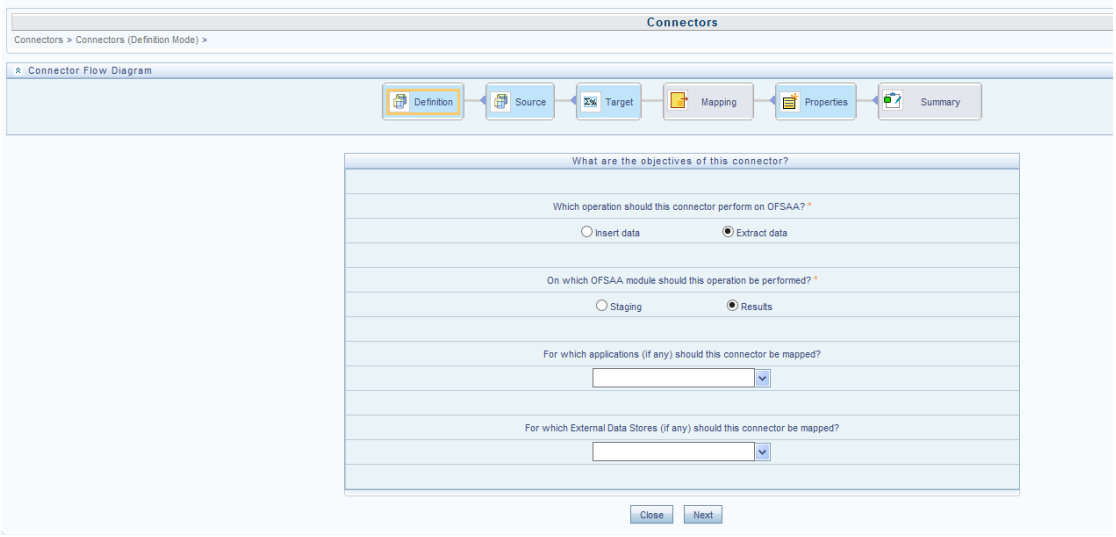
The Audit Trail section at the bottom of the window displays the information of the source created. The User comments section facilitates you to add or update additional information as comments.



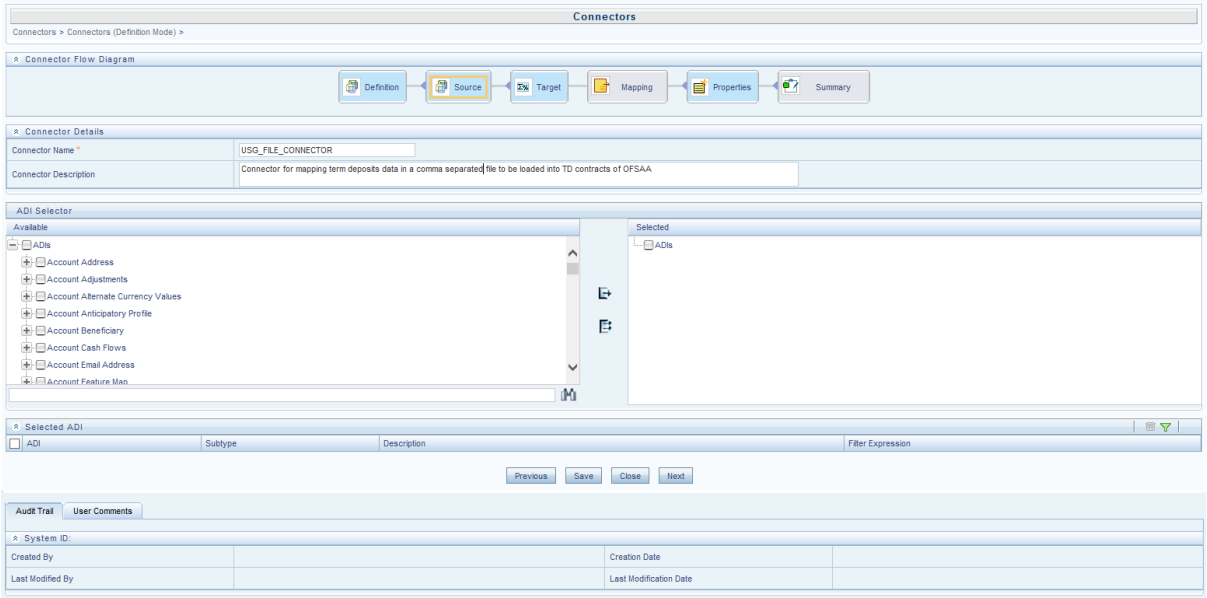
8.1.4 Creating a Connector for Extracting Data from Results

To create a Connector for extracting data from results follow the below steps:

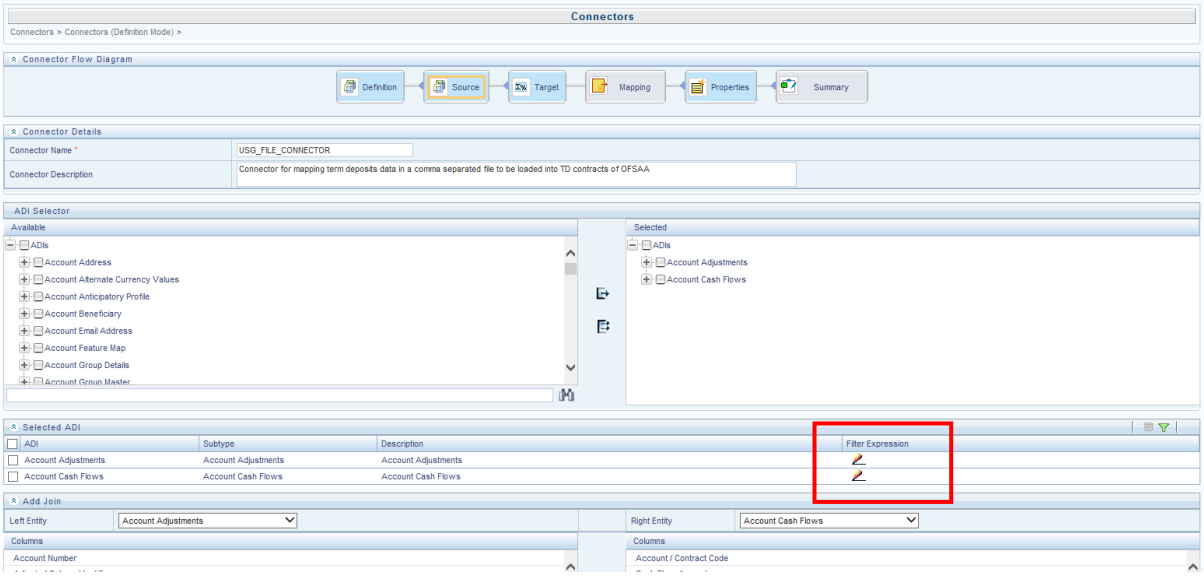
1. Navigate to **Data Mapping** > **Connector.**
2. Click the Add button from the tool bar. The *Connectors Definition* window appears.
3. The Definition tab provides the option to load and extract data. Select Extract data and Results. Click **Next**.



4. Click **Next.** The ‘Source’ block appears.



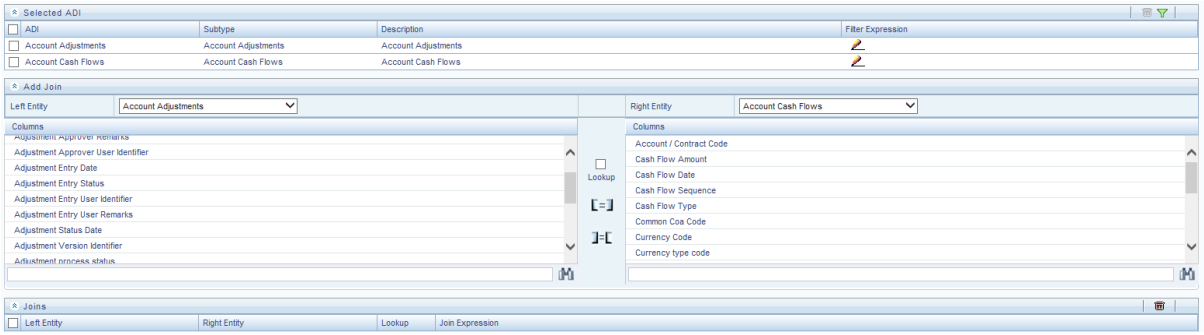
5. If Extract data option is selected, the ADI becomes source and EDD becomes target. Only staging relevant ADIs appear there.
6. Enter the Connector Name and description. The Fields and details are explained as tabulated in “Fields and their descriptions” section.
7. Under the ‘ADI Selector’ section, the available ADI’s are listed. Select the required ADI in order to map it. The selected ADI’s are displayed in the ‘Source’ section.



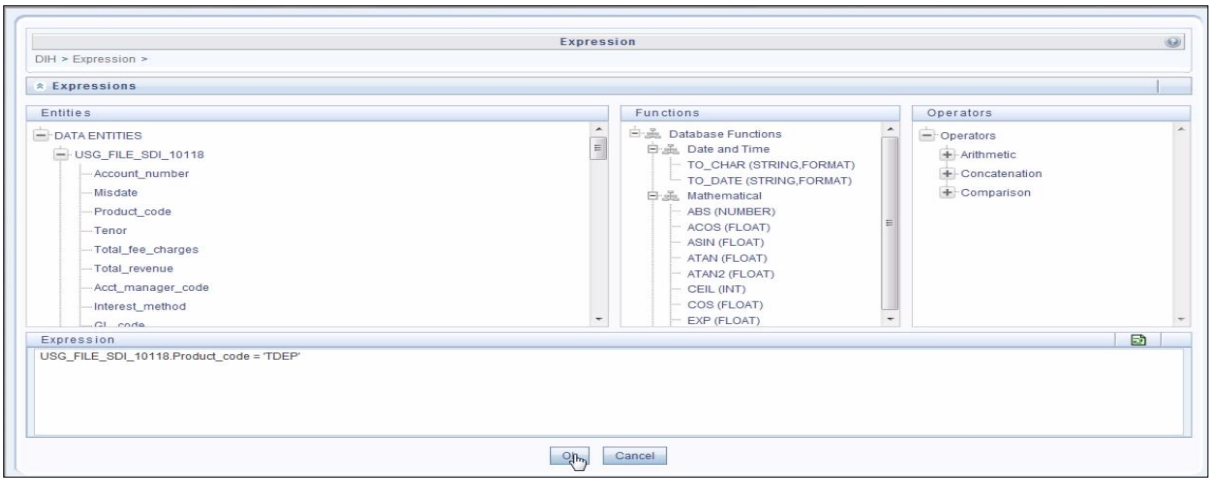
8. In case of multiple ADIs selected, **Add Join** section appears below with selected EDDs appearing on the left and right side. Select the fields from the Left and right side, and join the fields. If LOOKUP is required, then check the lookup checkbox has during the mapping.

NOTE: If Lookup option is checked, then the join would be left outer join. Else it would be inner join.

Note: For few ADIs/Subtypes, there are known relationship so joins in case of multiple ADIs occurs automatically.



9. To edit the filter expression, click  icon. The *Specify Expression* window appears. Select/key in the required expression and click **OK**.



NOTE:

- You do not need to add 'WHERE' clause for the filter.
- For File data loading, use filter expression of Number type along with single quotes. Example: N_DRAWN_AMOUNT ='40000'.
- For Date field refer To_CHAR function for comparison. Parameters can also be used in filter expression. Date format must be a valid SQL date format.

For Example:

```
[EDD_GL_DATA].[EXTRACTION_DATE] =
TO_DATE(#DIHDEV.MIS_DATE,'dd-MM-yyyy')
```

10. If Source type is Hive, the filter expressions must conform to the following restrictions:
    - Expression must be valid HiveQL
    - Does not include Oracle built in or user defined functions
    - Does not include Sub queries
    - Includes Hive built in functions only
    - Parameters can also be used in filter expression. MISDATE can also be passed dynamically so that it is loaded from Batch Execution Screen. The date format specified must be valid Hive Date format i.e. yyyy-MM-DD

    For Example:

    Filter Expression in Connector:-`[EDD_GL_DATA].[EXTRACTION_DATE] = #DIHDEV.MIS_DATE`

    MISDATE Parameter in Batch:

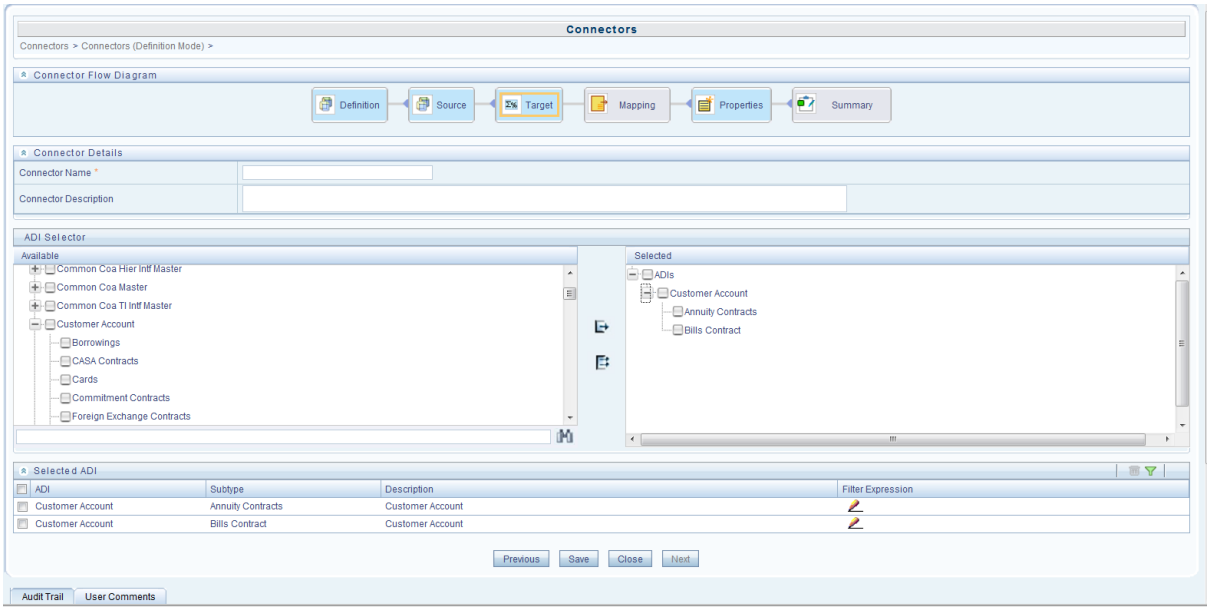
    ```
    MISDATE=$MISDATE:yyyy-MM-dd
    ```

NOTE: This is only applicable if the patch 8.0.2.1.4 (Bug - 24487929) is applied

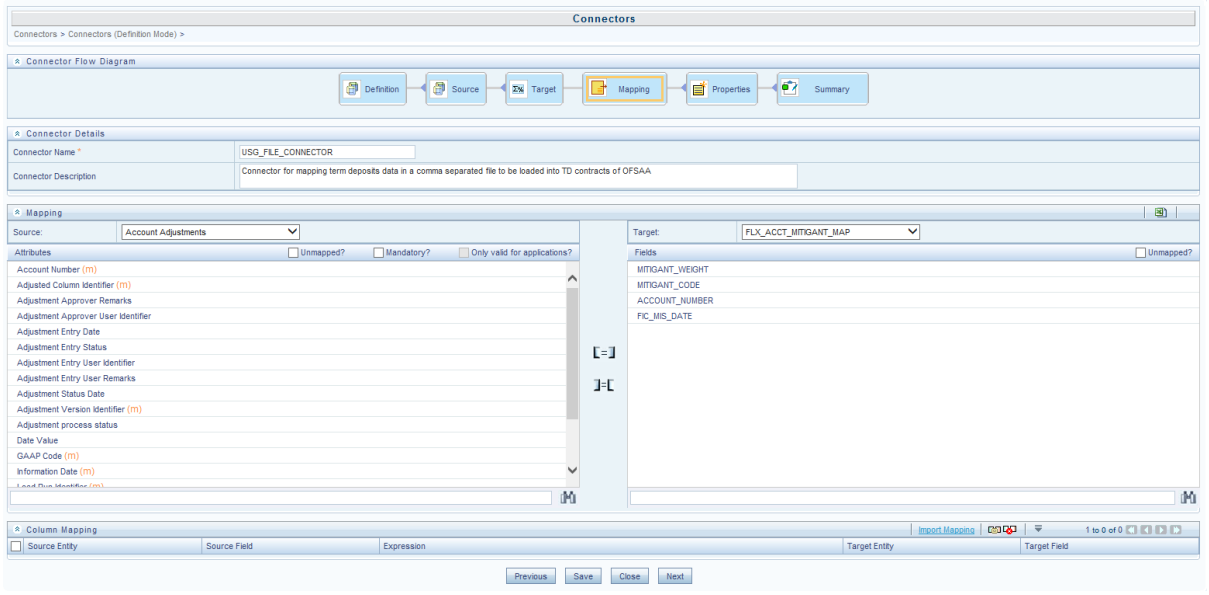
11. Click **Next. T**he 'Target' block appears.

12. Select External Data Descriptor from available list shows left panel. Move that to right side on selected panel. Below grid shows the selected EDDs along with filter expression. The filter on target can be applied by clicking the filter expression. The Fields and details are explained as tabulated in "Fields and their descriptions" section.

Note: For extract type connector, only one EDD should be selected. Multiple EDD is not supported.

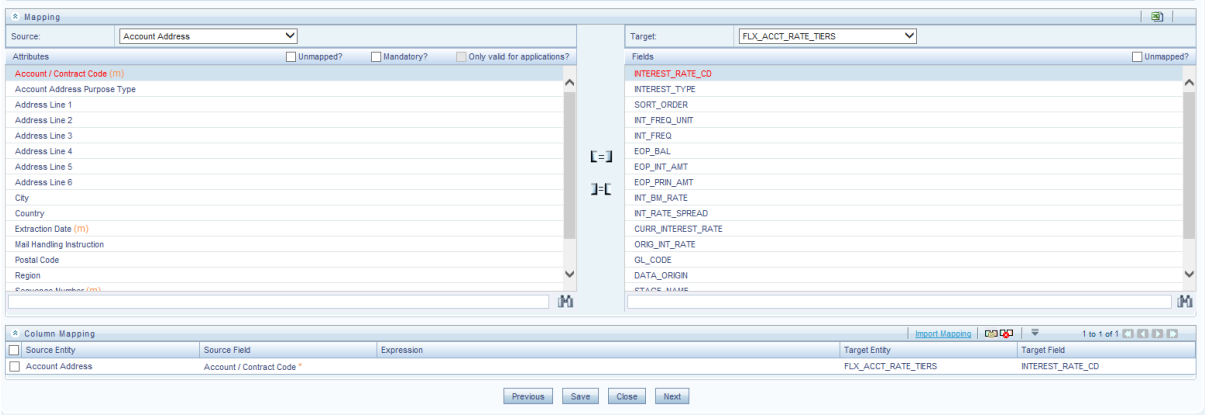


13. Click the **Mapping** block in the flow chart, in order to map the EDD's. For details on Auto-mapping refer to section Auto-mapping.



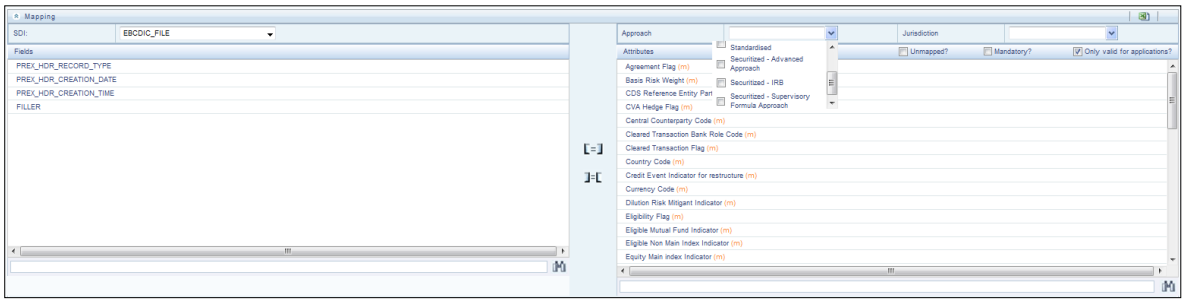
14. Select the ADI from the drop down list. For details on options for ADI refer to section Options in Mapping ADI.

    Note: For extract connector, mandatory attributes are not applicable. There is no validation of mandatory attributes during publish. Mandatory indicator (*) against an attribute for ADI is for reference only.

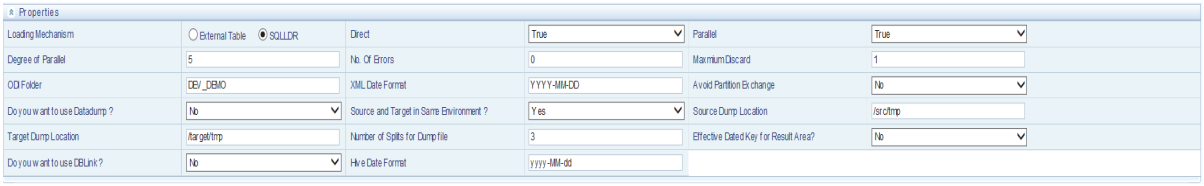


15. Select the ‘Mandatory’ option to populate the mandatory Attributes to be mapped. All the mapped attributes are listed under the ‘Column Mapping’ sector. For details on the options in the Mapping grid such as Auto-mapping, Export to Excel and so on refer to section Field in Mapping.

Note: If the application is chosen as BASEL, while mapping, you can select the type of approach (such as, IRB or Advanced Approach) and the Jurisdiction (BIS, BRAZIL, China, India and so on) for filtering.

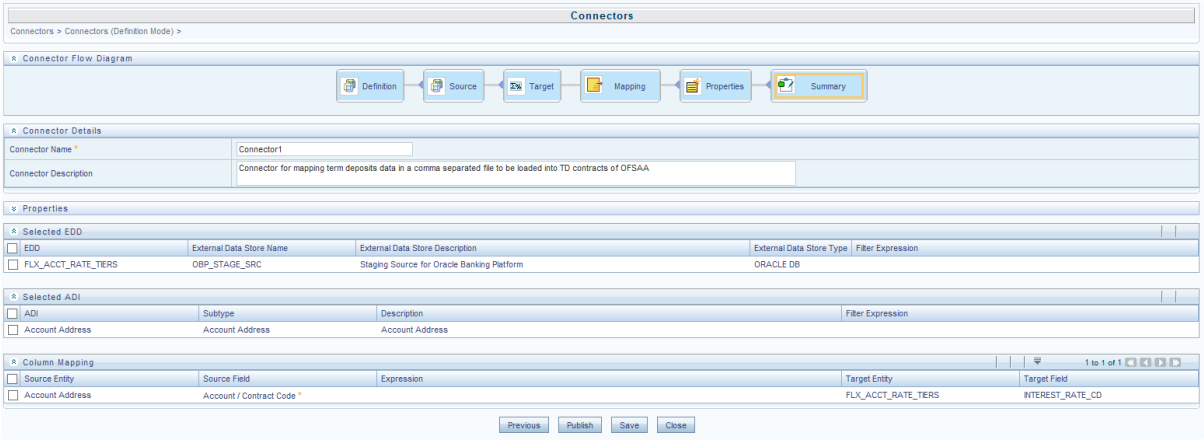


16. Click the **Properties** block in the flow chart. The Properties related fields, and the Connector details appear automatically.



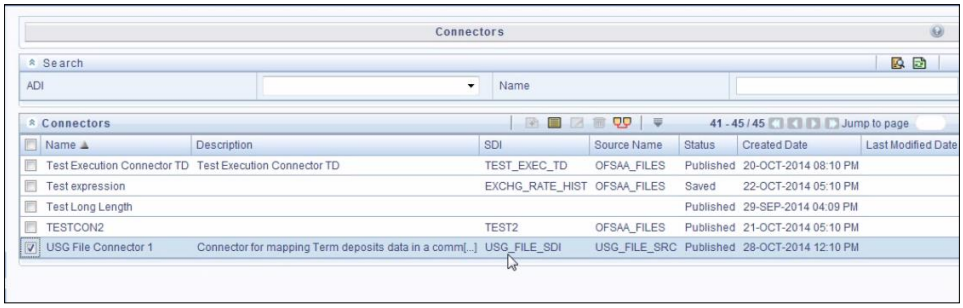
17. No action is required in properties section. Directly proceed to summary tab.

18. Click the **Summary** block in the flow chart to view the summary of all sections.



19. Click **Publish**. This converts DIH metadata into ODI objects.

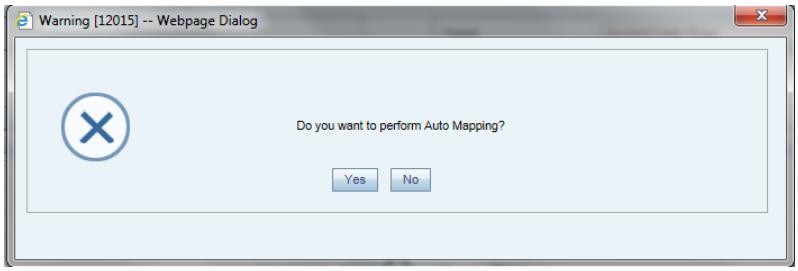
The Audit Trail section at the bottom of the window displays the information of the source created. The User comments section facilitates you to add or update additional information as comments.



8.1.4.1 Auto-mapping

The Mapping section appears, with an dialog box providing an option for Auto Mapping. Click **Yes** to proceed with Auto Mapping. Alternatively, click **No**, to perform Auto Mapping later or perform mapping manually.

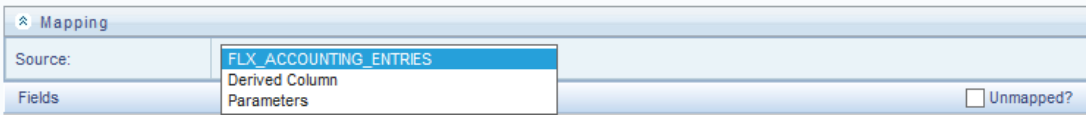
The list of Fields and Attributes are listed.



8.1.4.2 Options in Mapping EDD

In the Mapping grid of the Mapping block, when you click on the EDD drop-down, there are three options:

- **Defined EDD**: In this option, the defined EDD is listed in the Fields column which can be mapped to the required ADI.
- **Derived Column**: On selecting **Derived Column**, under the Fields column, you have 'Add Expression' option. On selecting 'Add option', the *Specify Expression* window appears. Here, you can select the required **Entities**, **Functions** and **Operators.** That is, you can write your own expression. Enter the field name and click **OK.** Now the newly created field name will be listed along with the 'Add Option'. Select the created filed and map it with the respective attributes.
- **Parameter**: On selecting Parameter, you can substitute it with the EDD. The respective parameters are listed in the fields' column. Map the fields and attributes.

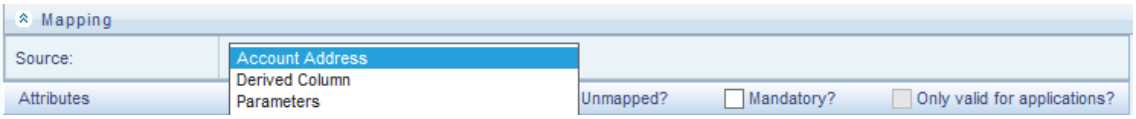


Example: USG_FILE_EDD. The Fields of the selected EDD is listed.

8.1.4.3 Options in Mapping ADI

In the Mapping grid of the Mapping block, when you click on the ADI drop-down, there are three options:

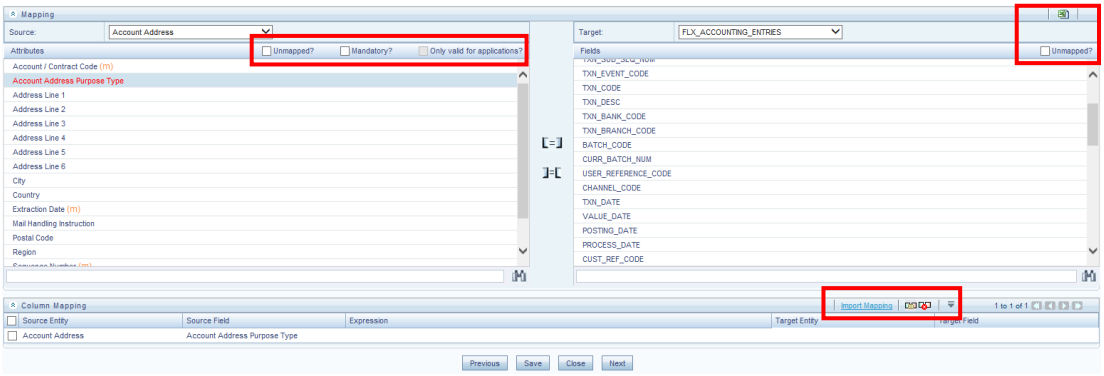
- Defined ADI: In this option, the attributes of the selected ADI which can be mapped to the required EDD.
- Derived Column: On selecting **Derived Column**, under the Fields column, you have 'Add Expression' option. On selecting 'Add option', the *Specify Expression* window appears. Here, you can select the required **Entities**, **Functions** and **Operators.** That is, you can write your own expression. Enter the field name and click **OK.** Now the newly created field name will be listed along with the 'Add Option'. Select the created filed and map it with the respective attributes.
- Parameter: On selecting Parameter, you can substitute it with the ADI. The respective parameters are defined can be mapped to a field on target.



Example: Customer Account. The Fields of the selected ADI is listed.

8.1.4.4 Fields in Mapping

The Mapping and Column Mapping grid in the Mapping block provides options, such as Auto-mapping, Export to Excel and so on, as explained below.



- You can view the unmapped fields and attributes by selecting the “Unmapped” checkbox. The mapped fields are displayed in red. You can also search for an attribute or field by keying the required word in the search option and clicking  icon.
- You can view the mandatory attributes by selecting the “Mandatory” checkbox.
- Only valid for applications, and is not applicable for extract.
- For data loading into Result area, DIH internally converts code/string values into surrogate keys/numbers by looking up into corresponding dimension table. This happens during publish. The lookup happens either on latest record or on the effective dated. It is decided by the parameter “Effective Dated Key for Result Area?”. Therefore, in the result area loading, a data type mismatch warning might appear. You can ignore the warning.
- There is automap button on the “Column Mapping” section next to “Import Mapping”. On clicking it, it maps Fields of the selected EDD on the left to attributes on the target based on name. And it puts underscore “_” in the space between words and tries to match. It tries to match with physical column name of the attribute as well. The user should review the mapping and delete if not correct. The ‘Delete mapping” button is also next to automap button.
- The Export to Excel  icon, helps in moving the mappings to an excel sheet. The excel sheet has the following columns:
    - EDD
    - Source Column
    - Target Attribute
    - Mandatory
    - Applicable for Applications

    The export process dumps all the source columns and target attributes for the current selection along with whatever is already mapped. You can

alter or do the mapping by changing the source/EDD and Target columns in the Excel.

Click “Import Mapping” button on the “column Mapping” section, and select excel mapping. The import process does the mapping if the mapping is valid or the same target attribute is not already mapped.

If EDD is not mentioned in the excel (manually created), then import mapping option enables you to map left and right attributes based on their name irrespective of EDD name. This enables the user to perform reference mapping. This indicates that is once mapping is established between the source fields and target attributes, on any connector with any EDD the same mapping can be used.

8.2 Viewing a Connector

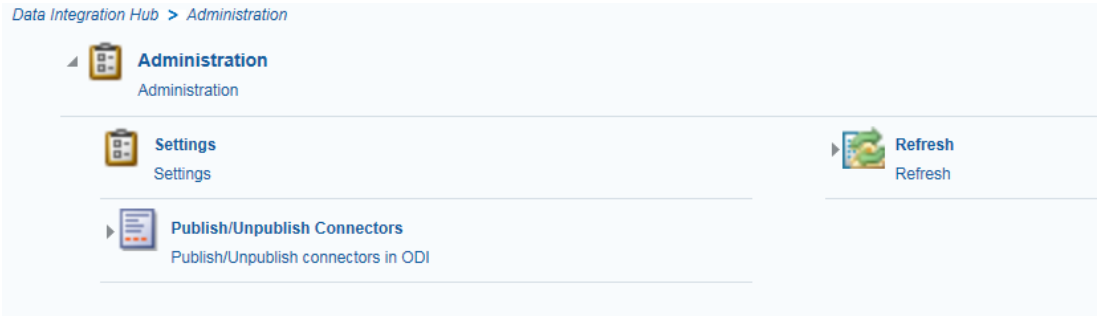
You can view a Connector at any given point. To view an existing Connector:

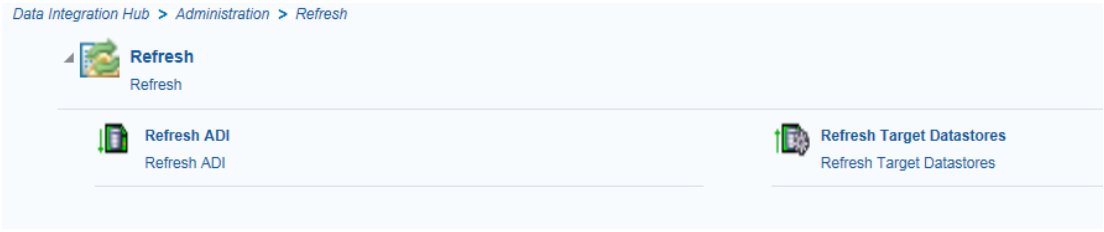
1. Select the checkbox adjacent to the name.
2. Click the View  button from the tool bar.
3. The *View* window displays the details of the selected Connector. In continuation with the example above, the ‘USG File Connector 1‘connector details appears on the view mode window.
4. Verify the details and click the **Publish** button. This creates ODI metadata from DIH metadata.

   If you want to make any changes, click ‘Unpublish’. The ‘Unpublish’ option clears ODI metadata that has been created in publish. Update the required changes and then click Publish, for the updated data to be loaded to ODI.

9 Refresh

Refreshing is the process of creating ADI and syncing the changes that happens in data model. Click the **Administration** option from the LHS menu and select **Refresh** option.





The Refresh option has the following options:

- Refresh ADI
- Refresh Target Datastores

9.1 Refresh ADI

The Refresh ADI creates ADIs by reading the uploaded data model in the setup. It works for incremental changes also. If there is any change in data model from the previous one, then ADI refresh compares the existing ADI with updated model and accommodates the changes either by adding new ADI or adding new attribute and so on.

If any changes are made to the model, you need to click **Refresh ADI** for the changes to appear.

Post the completion of Refresh ADI, please have a look at the log file named DIH.log (found under the Weblayer logs folder). Please check for any 'ALERT' messages in the log that indicate the possibility of inconsistencies in data caused by delete/update operations that were not permitted to be executed for some reason. Please follow any steps that may been suggested in these 'ALERT' messages.

Possible 'ALERT' messages and actions that need to be taken in response to these messages have been listed in the table below:

Message	Action
ALERT: Cannot delete this Adi as it is being used by one or more connectors....	Please unpublish and delete any connector/s which may be using this ADI.
ALERT: Cannot delete this AdiAttrMap as it is being used by one or more connectors….	Please delete any mappings to the concerned attribute from connectors which may be using this ADI.
ALERT: Cannot delete this attribute as it is being used by one or more connectors….	Please delete any mappings to the concerned attribute from connectors which have such mappings.
ALERT: Cannot delete this StVal as it is used by one or more connectors….	Please delete any mappings to this sub-type from connectors that may be using it.
ALERT: Cannot modify this StValAttrMap as it is being used by one or more connectors. This may have caused some inconsistencies in data….	Please delete any mappings that correspond to the given sub-type and attribute combination from connectors which may have such mappings.
ALERT: Cannot delete this Data Entity as it is used by a connector….	Please delete any mappings to the sub-type that corresponds to this data entity from connectors that may be using this sub-type.
ALERT: Cannot delete this column as it is being used by one or more connectors….	Please delete any mappings that correspond to the given sub-type and attribute combination from connectors that may have such mappings.

There are two audit tables which captures all the changes done by ADI refresh program. The structures are as follows:

- **fsi_adi_data_changes:** It stores the type of change for the object type and its name along with, the time when it is performed.

Column Name	Description
RUN_ID	This number identifies one execution of ADI refresh. Maximum number represents the latest ADI refresh execution.
ACT_ID	Activity ID. It is a number against each activity.
CHANGE_TIME_STAMP	Displays the activity time.
OBJ_TYPE	Displays the object type whether it is column or table.

OBJ_ID	It is an internal number.
OBJ_NAME	Displays the object name such as table or column name. If the obj_type is Col then it is column name and Table then it is table name.
CHANGE_TYPE	Represents what type of change whether it is added or removed etc. A means added.
ACTION_TAKEN	Whether action is taken up or not taken up.
COMMENTS	Displays the description of the activity. Such as “Column has been added”.

- **fsi_adi_data_changes_details:** It Stores the old and new values in case of rename of name and description. This is the detailed table that has reference to fsi_adi_data_changes.

Column Name	Description
RUN_ID	This number identifies one execution of ADI refresh. Maximum number represents the latest ADI refresh execution.
ACT_ID	Activity ID. It is a number against each activity. This activity ID is same as whatever is there in fsi_adi_data_changes table.
CHNG_DESC	Displays type of change.
OLD_VAL	Displays the old value. In case of rename of column name, it shows name of the column before change.
NEW_VAL	Displays the new value. In case of rename of column name, it shows name of the column after change.

9.2 Refresh Target Datastores

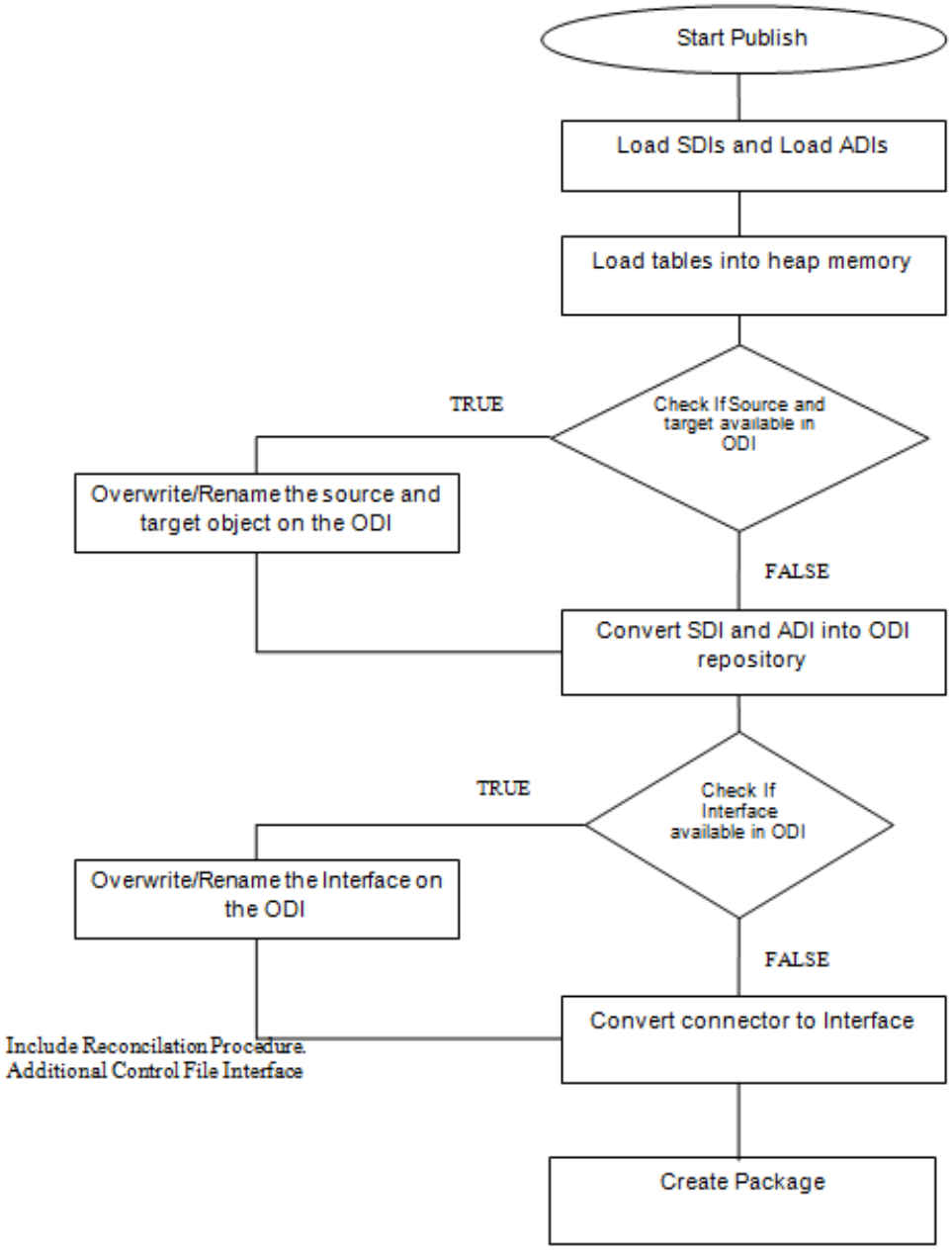
It represents underlying physical tables of every available ADI as data store in ODI. This process also works on incremental way

Post data model upload, if the changes are to be made available in DIH, then first click **Refresh ADI**. Once Refresh ADI is done, click **Refresh Target Datastore**.

NOTE: This is not a day to day activity.

10 Publishing DIH Metadata

Publishing is a process of transferring the DIH metadata to Oracle Data Integrator (ODI). It converts the DIH metadata into ODI metadata by creating required ODI objects such as DataServer, Logical/Physical Schema, Datastores, Interface, Package and Scenario. You cannot edit any DIH object after publishing. In order to edit, you need to unpublish, make the necessary changes and then re- Publish.



10.1 Batch Publishing and Unpublishing

You can batch publish/unpublish all the connectors/EDDs/EDSs to ODI at any given point.

Click the **Administration** option from the LHS menu and select **Publish/Unpublish Connectors** option. Click “Publish All” to publish all the connectors. Click “Unpublish All” to unpublish all the connectors.

11 Object Migration

This chapter has the following sections:

11.1 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, refer 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 Source and Target environment should be on data model version.
- Ensure that the source and target are at same DIH application level as well as at same model level before migrating objects from source to target.
- ADI refresh to be completed before this process.
- ODI settings are correctly specified in target environment prior to import of objects.
- Target data store refresh is also performed and successful.
- If exported object already exists in the target environment or an object with same name already exists in target environment then it will be overwritten with new definition details.

NOTE: If exported Parameter already exists in target Environment, they will not be overwritten unless the new definition is of different Parameter Type. If exported EDS already exist in target Environment, they will not be overwritten unless the new definition is of different EDS Type.

- Any exported object, if already exists in target or an object with same name already exists in target then that object and all its dependent objects must be in unpublished state for migration to go through successfully.

NOTE: Dependent objects for a Connector are EDS, EDD and Parameter. Dependent objects for EDD are EDS and Parameter. Parameter and EDS do not have dependent objects.

11.2 Exporting Objects from Source Environment

Follow the below procedure to export object from source environment.

1. Replace placeholders of UserId, Infodom with Source UserId & Infodom.

2. For `$Folder` put segment name for the infodom provided above. Mention locale as 'en_US'.
3. `$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 archive folder.

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

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

   In Mode tag: mention EXPORT.

   For **FAILONERROR** and **OVERWRITE**, it's recommended to mention 'Y'.
4. In OBJECT tag, mention "*" for Code property, to export all definitions. Else, for each definitions put equal number of OBJECT tag with map ID and version number in comma separated format.

   Type: Use following for definitions:

   - 70 for Parameter Definitions
   - 71 for EDS Definition
   - 72 for EDD Definition
   - 74 for Connector Definition

   Object codes can be derived from the following tables:

   - For object code of Connector: CONT_ID against the CONT_CODE in FSI_CONNETCOR_B
   - For object code of EDD: de_id against de_code in fsi_data_entity_b
   - For object code of EDS: ds_id against ds_code in fsi_ds_b
   - For object code of Parameter: var_id against var_code in fsi_var
5. The format for All OBJECTS tag is:

   ```
   <OBJECTS TargetFolder="DIHUSERS"><OBJECT Code="*" Type="73" /></OBJECTS>
   ```
6. For three definitions, OBJECTS tag is:

   ```
   <OBJECTS TargetFolder="DIHUSERS">
   <OBJECT Code="123221" Type="73" />
   <OBJECT Code="321331" Type="73" />
   <OBJECT Code="131213" Type="73" />
   </OBJECTS>
   ```
7. Execute `$MIGRATION_HOME/bin/ObjectMigration.sh` after providing executable permissions.
8. A file `$FILE_NAME.dmp` (rules.dmp) is created in `$MIGRATION_HOME/metadata/archive`

Move this file to `$MIGRATION_HOME/metadata/restore` folder. You can copy the file in the Target environment by creating "**restore**" folder under "**metadata**" directory (if not available).

9. Exporting definitions from Source environment is done successfully.

11.3 Importing Objects from Target Environment

Follow the below procedure to import object to target environment.

1. Repeat 1-3 steps as followed in export mode.

   In Mode tag: mention IMPORT.

2. In OBJECT CODE property, mention “312321”.

   Note: Everything that is exported is imported. You cannot choose only certain definitions to move.

3. Format for OBJECTS Tag is:

   ```
   <OBJECTS TargetFolder="DIHUSERS">
   <OBJECT Code="312321" Type="73" /></OBJECTS>
   ```

4. Execute `$MIGRATION_HOME/bin/ObjectMigration.sh` after providing executable permissions.

5. Check **DIH.log** for logs. It provides details such as, number of definitions that have successfully moved and other errors. Importing objects to target environment is done successfully.

12 Executing DIH Metadata

A published DIH Connector can be executed in the following two methods:

- **External schedule:** Execution is performed with the same connector name and same as ODI package/Scenario name.
- **From OFSAA Batch**: Creates a typical OFSAA Batch and executes it. For more information, refer to the AAI User Guide in OTN documentation library. The component is DIH Connector. You can select all the published connectors during task creation. DIH Connectors can be configured in Run Rules Framework (RRF) also with the same component.

NOTE:

- This can be done only on a published connector and not a saved connector.
- Currently there is a limitation of passing parameter in Run Rule Framework for DIH connectors. Batch framework can be used in case runtime parameter needs to be passed.

- If the connector contains any Runtime parameters, they can be set in the Variables input field of the Task Definition Screen.

  Example: `MISDATE='10-Jan-2015'`

- If there are multiple parameters, they can be passed by separating them with a comma.

  Example: `MISDATE='10-Jan-2015', BATCHID=22015`

- MISDATE and BATCH ID can also be passed dynamically so that it is loaded from Batch Execution Screen as follows:

  Example: `MISDATE=$MISDATE:dd-MM-yyyy, BATCHID=$BATCHID`

  In the above example, the date format appended to MISDATE has to conform to Simple Date Format. If no date format is specified, the default date format used is yyyymmdd.

- If variables are being used as part of connector mappings or filter expressions, they should be passed within single quotes as follows:

  Example: `MISDATE='$MISDATE:dd-MM-yyyy', BATCHID='$BATCHID'`

- If the date format is expected in dd-MON-yyyy format, then in Batch Task it has to be specified in the following format. Note the difference in month format in the following example:

  Example: `MISDATE='$MISDATE:dd-MMM-yyyy'`

- If parameter is used in connector filter expression for an EDD of source type Hive, date format is expected in yyyy-MM-dd format.

  Example: `MISDATE='$MISDATE:yyyy-MM-dd'`

NOTE: This is only applicable if the patch 8.0.2.1.4 (Bug - 24487929) is applied.

13 Execution History

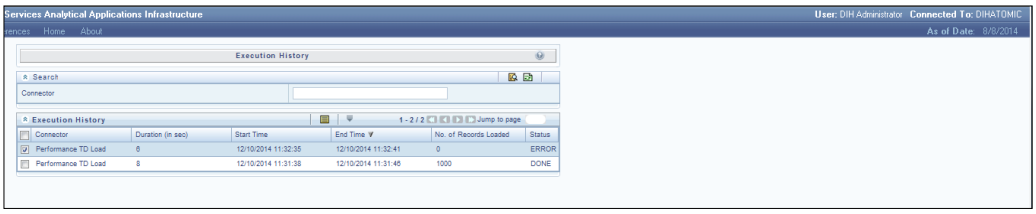
Execution History option, provides the status of the DIH Connector executions, number of records loaded, and error messages if any.

13.1 Viewing the Execution History of a Connector

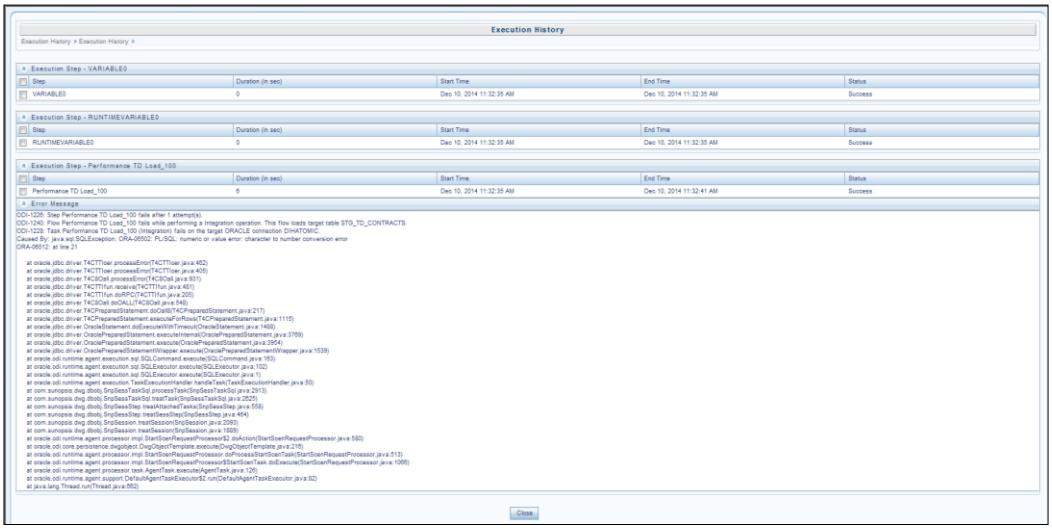
Select the 'Execution History option on the LHS. The Execution History page appears.

You can view the Execution History of a Connector at any given point. To view execution history:

1. Select the **Execution History** option from the LHS.
2. Select the checkbox adjacent to the Connector.



3. Click the View button from the tool bar. The *Execution History* window appears.



As the name suggests, this page provides the status of the DIH Connector executions, number of records loaded, and error messages if any.

14 Metadata Browser

The DIH metadata/objects available in the Meta Data Browser (MDB) are:

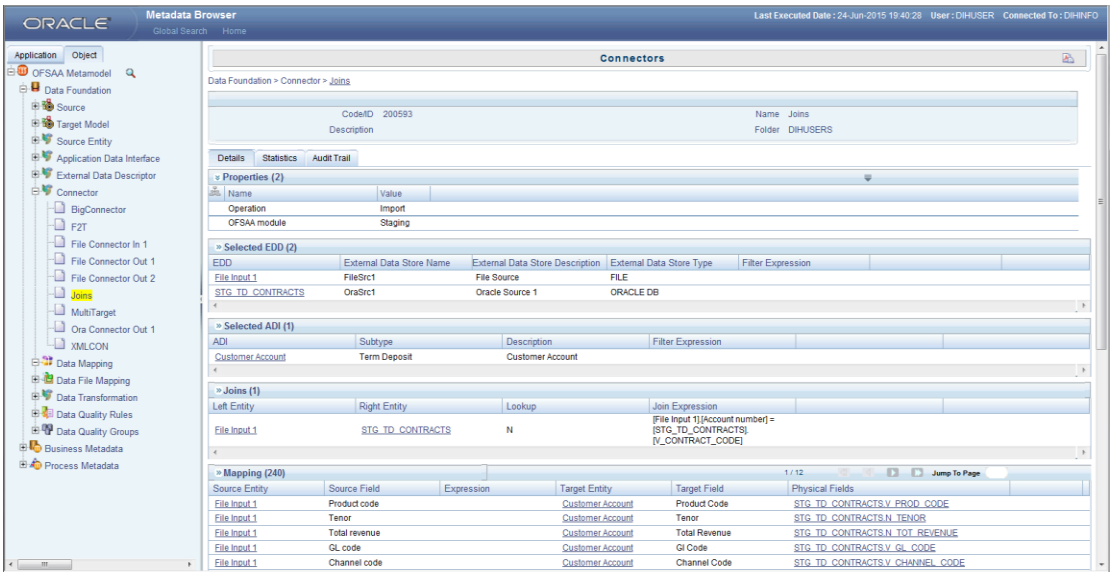
- Connector
- Application Data Interface
- External Data Descriptor

14.1 Connector

The connector link lists down the individual connectors.

When you click a connector name, it displays the definition of that connector.

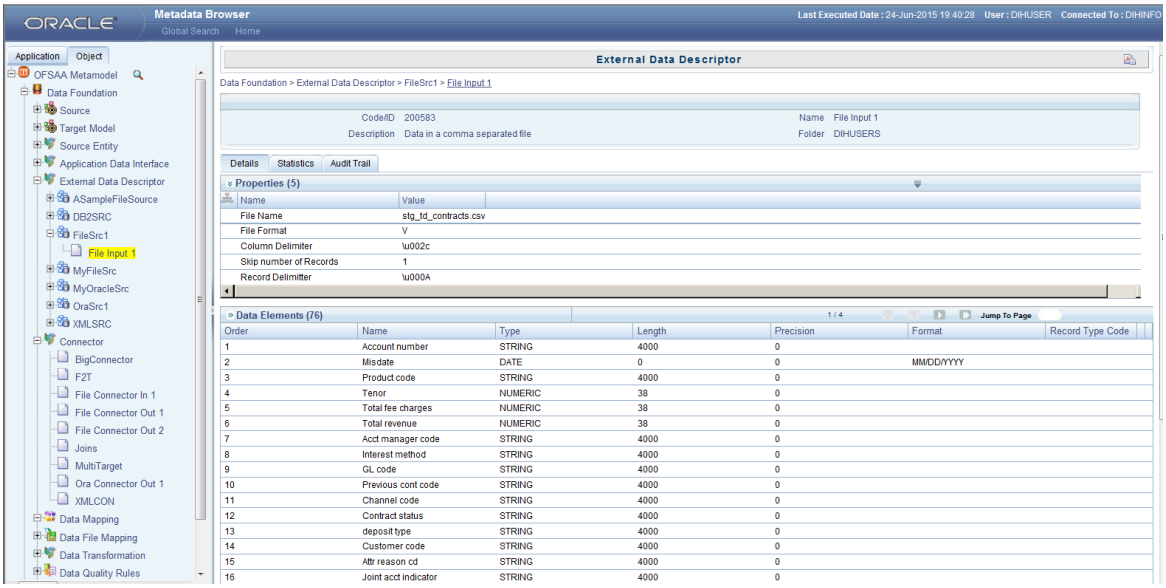
- The name and description of the connector is displayed at the top.
- The “Selected EDD” grid which shows the EDD associated with the connector.
- The “Selected ADI” grid shows ADI for the connector.
- The “Joins” grid shows the join expression between EDDs if applicable.
- The “Mapping” grid shows mapping of the Left field with the Right attributes. Additionally, it displays the underlying physical columns.
- The “Depends On” grid shows EDD and ADI used in the connector.



14.2 External Data Descriptor

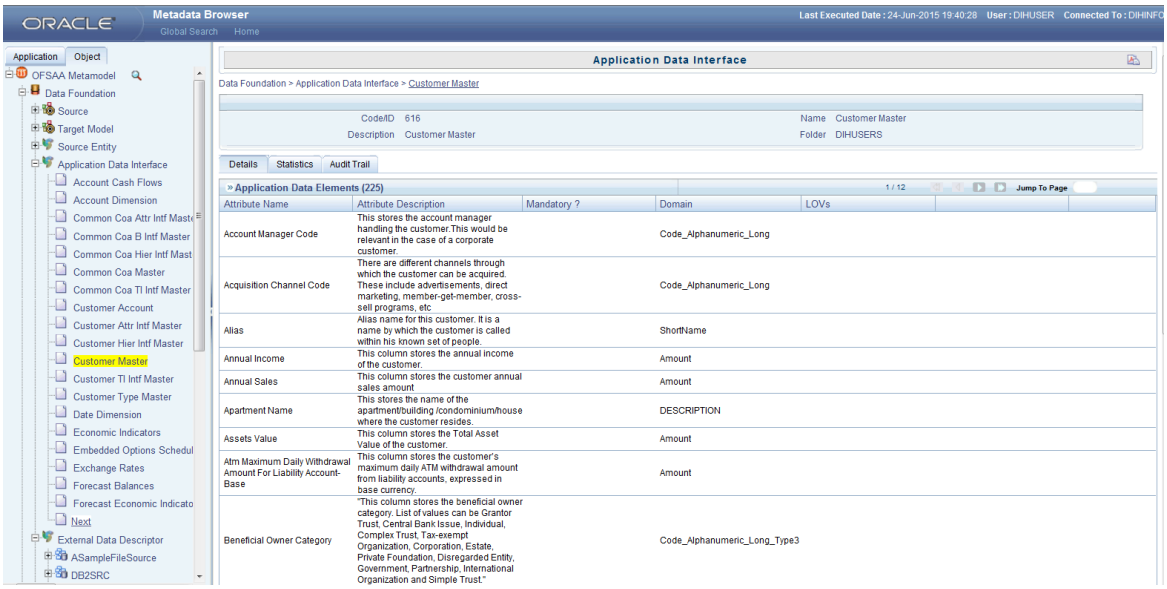
The “External Data Descriptor” window lists all the available External Data Store. Under each EDS it has all EDDs. When you click a particular EDD, it shows the definition of the EDD, with the name and Description displayed at the top.

- The “Properties” sections shows properties such as file name, file format and so on.
- The “Data Element” sections show all the fields that are part of this EDD.



14.3 Application Data Interface

The Application Data Interface window shows all the attributes along with descriptions of the List of Values (LOVs).



Appendix A: Use Cases

Loading Data from One File into Staging

To load data from one file to Staging, follow the below steps:

1. Create an EDS of the type **File**.
2. Create EDD by selecting the pre-defined EDS. Provide all required information while creating the EDD. If post loading reconciliation is required then go to Control tab and provide a control record.
3. Create a Connector for loading data into staging. If SQLLDR option is enabled, then the file with ODI agent running should be available. If External Table option is selected, then file has to be available in target database server.
4. Publish the Connector.
5. Execute the Connector.

Loading Data from Two Files Joined into Staging

To load data from two files which are joined, to Staging, follow the below steps:

1. Create EDS of type **File**.
2. Create two EDD by selecting the pre-defined EDS. Provide all required information while creating EDD. If post loading reconciliation is required then go to Control tab and provide a control record.
3. Create connector for loading data into staging. Select both the EDDs and establish a join in source tab. If SQLLDR option is enabled, then the file with ODI agent running should be available. If External Table option is selected, then file has to be available in target database server.
4. Publish the Connector.
5. Execute the Connector.

Loading Data into Staging from File and Performing Lookup into a Table

To load data from a file, to Staging, and perform lookup into a table, follow the below steps:

1. Create one EDS of type **File** and another EDS of type **Database**.
2. Create two EDD by selecting the predefined EDS. Provide all required information while creating the EDD. If post loading reconciliation is required then go to Control tab and provide a control record. Post loading reconciliation is only applicable for file type Data Loading.
3. Create a Connector for loading data into staging. Select both the EDDs and establish a join and click **Lookup**. If SQLLDR option is enabled, then the file with

ODI agent running should be available. If External Table option is selected, then file has to be available in target database server.

4. Publish the Connector.
5. Execute the Connector.

Loading Data into Staging from an XML File

To load data to Staging from an XML file, follow the below steps:

1. Create an EDS of type **XML**.
2. Create EDD by selecting the predefined EDS. Provide all required information while creating the EDD. To get the XML structure, use XSD file. If post loading reconciliation is required then go to Control tab and provide a control record. The control record should be in a separate file.
3. Create a Connector for loading data into staging.
4. Publish the Connector.
5. Execute the Connector.

Loading Data into Staging from EBCDIC

To load data to Staging from EBCDIC, follow the below steps:

1. Create an EDS of type EBCDIC.
2. Create EDD by selecting the predefined EDS. Provide all required information while creating the EDD. To define the EBCDIC structure, use Cobol Copy book file. If post loading reconciliation is required then go to Control tab and provide a control record. The control record should be in a separate file.
3. Create a Connector for loading data into staging.
4. Publish the Connector.
5. Execute the Connector.

Loading Data from File into Multiple ADI/Subtypes

To load data from File into multiple ADI/Subtypes, follow the below steps:

1. Create an EDS of type **File**.
2. Create EDD by selecting the predefined EDS. Provide all required information while creating the EDD. To define the file structure, you can use Excel template. If post loading reconciliation is required then go to Control tab and provide a control record.
3. Create a Connector for loading data into staging. Select multiple ADIs/Subtypes. Set filter against each ADI/Subtype selected to identify which record goes where.
4. Publish the Connector.
5. Execute the Connector.

Loading Data into Results from File

To load data into Results from File, follow the below steps:

1. Create an EDS of type **File**.
2. Create EDD by selecting the predefined EDS. Provide all required information while creating the EDD. To define the File structure, you can use Excel template for defining the columns. If post loading reconciliation is required then go to Control tab and provide a control record. The control record has to be in a separate file.
3. Create a Connector for loading data into results. Select EDD and ADI (appears only relevant for Result area). During mapping, code values that are available in file should be mapped to the attributes (that are internally number columns). While loading it converts the code value to surrogate key by performing a lookup in the related dimension table. Dimension data should be populated before this loading occurs using SCD process.
4. Publish the Connector.
5. Execute the Connector.

Extracting Data from Staging into File

To extract data from Staging into File, follow the below steps:

1. Create an EDS of type **File**.
2. Create EDD by selecting the predefined EDS. Provide all required information while creating the EDD. To define the File structure, you can use Excel template for defining the columns. If post loading reconciliation is required then go to Control tab and provide a control record. The control record has to be in a separate file.
3. Create Connector for extracting data from staging. ADI becomes source and EDD becomes target. File structure will be according to EDD.
4. Publish the Connector.
5. Execute the Connector.

Extracting Data from Result into File

To extract data from Results into File, follow the below steps:

1. Create an EDS of type **File**.
2. Create EDD by selecting the predefined external data store. Provide all required information while creating the EDD. To define the File structure, you can use Excel template for defining the columns. If post loading reconciliation is required then go to Control tab and provide a control record. The control record should be in a separate file.
3. Create a Connector for extracting data from results. ADI becomes source and EDD becomes target. File structure will be according to EDD. During extract internal surrogate keys get converted into code values by performing looking up into dimension table.

4. Publish the Connector.
5. Execute the Connector.

OFSAA out of the box dimension tables as lookup entity

Out of the box dimension tables will be used in the following scenarios:

- Loading Data into results
- Extracting data from results

While defining mapping, if results ADI surrogate key column is mapped, then corresponding dimention table will be used as lookup.

For Example:

- Loading data into results

Example: If N_Product_Skey of Common Account Summary ADI is mapped to an EDD product code column, then DIM_PRODUCT Table is used as lookup for getting the N_Product_Skey value and V_PROD_CODE will be used in the join expression. The EDD does a left outer join with the DIM_PRODUCT.

- Extracting Data from Results

Example: If N_Product_Skey of Common Account Summary ADI is mapped to an EDD product code column, then DIM_Product Table is used as lookup for getting the V_PROD_CODE value and N_Product_Skey will be used in the join expression. The Common Account Summary lookup does a left outer join with DIM_PRODUCT.

NOTE: The lookup is established identifying the Surrogate key relationship between Results ADI and Dimension table automatically. User need not define the lookup in the connector.

Oracle Financial Services Data Integration Hub User Manual

Release 8.0.2.0.0

Oracle Corporation

World Headquarters

500 Oracle Parkway

Redwood Shores, CA 94065

U.S.A.

Worldwide Inquiries:

Phone: +1.650.506.7000

Fax: +1.650.506.7200

www.oracle.com/ financial_services/