Oracle® Financial Services Data Foundation Cloud Service for Banking

Data Integration Guide

Release 24D
G26914-01
February 2025

ORACLE

Oracle Financial Services Data Foundation Cloud Service for Banking Data Integration Guide, Release 24D

G26914-01

Contents

Index

1

Managing Data

This section provides details about exchange of data between Financial Services Data Foundation Cloud Service for Banking and external systems. This is facilitated through the logical abstraction of the Data Catalog exposed as Application Data Interfaces (ADI). External Data Descriptors (EDD) are defined through the user interface, which also facilitate the mapping of EDDs to ADIs and forming Connectors.

Related Topics:

- External Data Descriptor
- Connectors

2
File Operations

The file operation process enables you to add files that must be processed to the object store. You can upload any CSV, XLSX, and Text file that you wish to be stored in the object store, which can be processed later.

To upload a file, complete the following steps:

1. On the DFCS home page, click the **User Menu** and select **Administration**.
2. Click **File Operations**.
3. Click **Upload File**.

   The **Generate PAR URL to upload file** window is displayed.
4. Enter the **File Name**.
5. Select the file format from the drop-down list.

   The supported file types are: CSV, XLSX, Text, and Zip.
6. Enter the file size (in bytes) as seen on the disk.

   To add more files, click **Add** and repeat the steps.
7. Click **Generate**.

   This generates a PAR (pre-authenticated request) URL which is valid for 24 hours. This PAR URL is used to upload the file into the object store.
8. An entry for the PAR URL you just created appears on the File Operations page.

   The search feature provides you multiple options to search for a file based on name, creation date, and so on. You can also sort the list by name and creation date in ascending or descending order.
   You can download a file using REST API. For more details, refer Generating PAR URL for File Download.

3
Data Ingress

This section describes about the different Data Descriptors.

> **Note:**
>
> Partitioning of physical tables are decided based on the initial load date for any entity. Hence, users are advised to load any entity from the initial load date. Loading data for a prior date after loading the data for later date will receive an error.

3.1 External Data Descriptor

External Data Descriptors (EDD) are definitions of specific data content from External Data Sources (EDS). Each EDS may have a number of EDDs defined against it. EDD definitions can be used for ingesting data into Financial Services Data Foundation Cloud Service for Banking or extracting data from it.

DFCS also supports the usage of control information while ingesting data into the Data Foundation. Such control information may also be defined as EDDs.

1. On the **Financial Services Data Foundation Service for Banking** home page, click **External Data Descriptors**. The **Summary** page is displayed.
2. Use the search functionality if you are looking for an existing EDD.
3. Click **Add** to configure additional EDDs. Provide a name and description, and select the **Datastore**. The **Data**, **Transformation**, and **Control** tabs appear based on your selection.
4. Click **Delete** to delete an existing EDD.

3.2 Data Tab

The following table describes the information you must provide in the Data tab when creating an EDD.

Table 3-1 Data Tab Description

Fields	Description
Is the file archived	By enabling this option, you can upload an archived data file to the object store as input to the DFCS data ingestion process. You can also combine multiple files into an archive and use it to load data into DFCS.

Table 3-1 (Cont.) Data Tab Description

Fields	Description
Archive file name	This field is displayed only when **Is the file archived** option is enabled.Enter the archive file name. Example: td_contracts%#MISDATE%.zip Only archives with.zip extension using standard DEFLATE algorithm is supported.
Format for data files in archive	This field is displayed only when **Is the file archived** option is enabled.Enter the format of the data files. For example: *.csv, td_contracts%#MISDATE%.csv. **NOTE**: The supported wildcard character is *
Data File Name	You can add multiple data files to an EDD. For example, if you need to add the 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, such as td_contracts%#MISDATE%_1.csv, and as the next record, add the Term Deposits Contracts data file for corporate accounts. In case the file sizes are large, it is recommended you break the file into smaller files. Optimize file size is 3 GB for each file. Example: td_contracts%#MISDATE%_1.csv
Record Delimiter	The records are stored differently in different operating systems. The available options are: • MS-DOS • Unix • No Record Delimiter • Other For example, select Unix.
File Format	There are two options: • Fixed Length: The file has records and columns with a fixed length. Each column has a predetermined and unchanging size, set when the record layout is designed, and the sum of the column sizes add up to the record size. • Delimited: There is a separation of the records and columns using a delimiter character like a comma, semicolon, hyphen, and so on. For example, select Delimited.
Text Qualifier	A character that identifies text. Generally, double quotes are prefixed and suffixed to identify text. This is optional.
Skip Number Of Records	Provide the number of records to be skipped. The records are skipped from the top. Generally, this is used to skip headers.
Decimal Separator	The character used to identify the decimal and fractional part. Usually point (.) or comma (,)

Table 3-1 (Cont.) Data Tab Description

Fields	Description
Read from template	Specify if you want to use a template containing all values in the Excel file format. If the template is not available, create it manually by clicking Add, under the Data Elements. If the template is available, you can browse for the template. See the File EDD Template. You can also drop the template in the area “Drop template here or click to select “.
Select Template (*.xls,*.xlsx,*.csv Files Only)	Click Browse and select the required template.
Data Elements	
Name	Name of the field in EDD. Example: Field name in a file or column name in a table. **Note:** The Field name of XML type must not be more than 25 characters and for others must not be more than 30 characters.
Type	This shows the Data type. Example: String, Number, and Date.
Length	This is applicable only for the EBCDIC format. This is the length of the EBCDIC data type. In the case of a file, it is length only.
Precision	This is used to specify the number of digits after the decimal point. Example: 10.3.

Table 3-1 (Cont.) Data Tab Description

Fields	Description
Format	Specify the format for columns of type date here. If left blank, a default format of DD/MM/YYYY is assumed and used. **Note:** • The default format is fixed and does not change with database or system language settings. • For data ingestion, enter the format in which date fields are provided. • For data extraction, enter the format in which extracted date fields must be recorded. For example, in the file extract, to represent the date, 31st January 2020, as “31/01/2020”, specify the format as DD/MM/YYYY for the date columns.
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.

3.3 Control Tab

Using the Control tab, you can:

1. Reconcile between source data received and data loaded into Financial Services Data Foundation Cloud Service for Banking, using the control information available from the source system. This feature is used to implement record count validation or amount reconciliation from a source file. It also allows specifying a threshold for validation. The specified threshold is compared with the difference (record count or amount) to perform the validation. The execution fails only if the difference is more than the threshold value.
2. The threshold can be an absolute or a percentage value. The connector execution process considers the threshold type while performing reconciliation.
3. Generate control information for file extracts from DFCS. Specifying control for extract EDDs generates control details, based on the configuration in the EDD. Both record count

and amount value (sum, average, max, and so on) for specific columns are recorded into the control file.

> **Note:**
>
> This option is applicable only for file type EDDs (ASCII and EBCDIC).

Table 3-2 Control Tab Description

Field	Description
NOTE: The Reconciliation details present in separate file option cannot be modified if the option **Is the file archived** in **Data Tab** is enabled. When Archive option is enabled:	
If you want to perform reconciliation, control file must always be present in the same archive as input data files and should be a separate file.	
The control file should have only two columns that are Control Name and Expected Value.	
When **Separate File** is selected as **Yes**.	
File Name	Specify the name of the file.
File Format	There are two options: • **Fixed Length**: The file has records and columns with a fixed length. Each column has a predetermined and unchanging size, set when the record layout is designed, and the sum of the column sizes add up to the record size. • **Delimited**: There is a separation of the records and columns using a delimiter character like a comma, semicolon, hyphen, or so on. In the previous example, select Delimited.
Column Delimiter	If the File Format is selected as Fixed Length, the Column Delimiter would be Other by default. If the File format is selected as Delimited, the following options are available in the drop-down list. • Other • Space • Semicolon • Comma • Tab In the previous example, select Comma.
Record Type Code	Used to uniquely identify a record within a file. A financial institution sometimes provides files that have data and control records within the same file. In such a case, 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 the Control record, the value is specified under the Control tab. Only the first field of a file is used for Record Type.

Table 3-2 (Cont.) Control Tab Description

Field	Description
Record Delimiter	The records are stored differently in different operating systems. The following options are available: • MS-DOS • Unix • No Record Delimiter • Other For example, select Unix.
Skip number of records	Provide the number of records to be skipped. The records are skipped from the top. Generally, this is used to skip Headers. Example: If you specify this as 1, the first row in the file will be ignored.
Text Qualifier	A character that identifies text. Generally, double quotes are prefixed and suffixed to identify text. This is optional.
Decimal separator	Specify up to which decimal digit you want to view the result.
Record Type Length	The length of the record type value to pick up the correct record. For example, if the control record is “DATATotal Records400” and DATA is the Record type, the length is ‘4’. This is applicable only for Control records that are of fixed length.
Control Name Length	Based on the previous example, the Control name is “Total Records”. Therefore, the Control Name Length is ‘13’.
Control Value Length	Based on the previous example, the Control value is 400. Hence, the length of the control value is ‘3’.
When **Separate File** is selected as **No**.	
Record Type Code	Used to uniquely identify a record within a file. A financial institution sometimes provides files that have data and control records within the same file. In such a case, 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 the Control record, the value is specified under the Control tab. Only the first field of a file is used for Record Type.
Control Value Length	Based on the previous example, the Control value is 400. Hence, the length of the control value is ‘3’
Control Name Length	Based on the previous example, the Control name is “Total Records”. Hence, the Control Name Length is ‘13’.
Controls	
Control Name	Specify the name of the control.

Table 3-2 (Cont.) Control Tab Description

Field	Description
Aggregation Method	Select either Aggregation Method or Count. The supported aggregation methods are as follows: • Min • Max • Average • Sum
Aggregation Column Name	Select the column on which the aggregation method is applied. NOTE: For count, no column needs to be selected.
Threshold Type	This field is optional. There are two available options for threshold: percentage or absolute. If you select percentage, the reconciliation difference in percent is matched against this threshold value. If you select absolute, the absolute percent difference is matched against this threshold value.
Threshold Value	Specify the difference value in percent or absolute.

3.4 Transformation Tab

Table 3-3 Transformation Tab Description

Fields	Description
Transformation Type	A drop-down listing the different types of transformation supported. Currently, only Aggregation is supported.
If the Transformation Type is selected as **None**:	
Derived Data Elements	
Name	Name of the derived field in the EDD. NOTE: Field names must not be more than 30 characters.
Type	Shows the Data type of the field. Example: Varchar2, Number, Date, and so on.
Expression	When you select the **Add** option, the Specify Expression window is displayed. Here, you can select the required entities, functions, and operators. That is, you can write your expression. Enter the field name and click **OK**. Then the newly created field name is listed.
If the Transformation Type is selected as **Aggregation**:	
Derived Data Elements	
Name	Name of the derived field in the EDD. Note: Field names must not be more than 30 characters.

Table 3-3 (Cont.) Transformation Tab Description

Fields	Description
Type	This shows the Data type of the field. Example: Varchar2, Number, Date, and so on.
Expression	When you select the **Add** option, the Specify Expression window is displayed. Here, you can select the required entities, functions, and operators. That is, you can write your expression. Enter the field name and click **OK**. Now the newly created field name is listed.
Aggregation Properties	
Group By	This is available when Aggregation is selected.
Having	This is available when Aggregation is selected.

3.5 Defining an External Data Descriptor

To define a new **External Data Descriptor (EDD)** from the DIH Designer window, follow these steps:

1. Navigate to the **Data Foundation Cloud Services** home page, and then click **EDD**.

   The **Data Ingestion Summary** page is displayed.

2. On the **Data Ingestion Summary** page, click **Add**.

   This will open the **New** tab.

3. Enter the **name** and **description** for the data descriptor, and then select **Datastore** from the drop-down list. Select the data source (example: **FOUNDATION_DATA_STORE**). The values in Defining an External Data Store example are used. The description will be automatically populated.

4. Click **Save as Draft** or click **Save**.

5. Enter the required values in the fields.

6. If data needs to be reconciled post-loading, click the **Control** tab. Currently, only the Number of Records controls is available in the version.

7. To transform the EDD, click the **Transformation** tab.

8. You can add derivation to data elements of the EDD.

   - Click **Add** to create derived data elements.
   - To edit the derived data elements, click **Edit**. This will open the **Expression** window, where you can specify the expression using data elements defined in the **Data** tab and functions.
   - To delete the derived data element, click **Delete**.

9. Click the **Transformation** tab and select the **Transformation Type**.

   - Select **Aggregation** and click **Edit** to view **Expression** window.
   - In the expression window, specify the **Group by** clause and **Having** expression, if applicable.
   - Define Derived Data Elements for the field to be aggregated under the previous tab.

10. Click **Save**.

3.6 Modifying and Viewing an External Data Descriptor

You can edit or view existing EDDs.

> **Note:**
>
> You cannot edit EDDs in **Published** status.

To edit or view an EDD, complete the following steps:

1. From the EDD Summary page, select the required EDD.
2. The details of the selected EDD are displayed. You can modify or view the details.
3. Update the required details.
4. Click **Save** to save the changes made.
5. Click **Save as Draft** to save and update later.

   The status shows as **Draft**.

3.7 Deleting an External Data Descriptor

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

For example, assume an EDD is used in a Connector. Then, unless the Connector is deleted, the used EDD cannot be deleted.

To delete an existing EDD, complete the following steps:

1. On the EDD Summary, click **Delete** corresponding to the EDD you want to delete.
2. Acknowledge the confirmation message.

   The EDD details are deleted.

3.8 Search and Filter

Use the search and filter options to find the required information. You can enter keywords nearest to what you are looking for in the search box. You can search for an EDD using either the name, description, status, or type.

For example, enter the EDD keyword as Loan Data in the search box. The search results show the list of all EDDs containing the text **Loan Data**.

3.9 Parameters in EDD Definition

While defining an EDD, the parameter can be used as a placeholder in a data filename.

For example: Consider a table with two columns: Account number and Balance.

Table 3-4 EDD Parameters Example

Account Number	Balance
A1	1000
A2	1000
A3	1000
A1	1000
A2	1500
A3	1500

In this example, a customer has three 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 placeholder 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 Runtime.

Parameter 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
Data Egress

This section describes the Data Connectors used in Financial Services Data Foundation Cloud Service for Banking.

4.1 Connectors

Connectors allow mapping one or more External Data Descriptors with an Application Data Interface. You can configure the connectors for data exchange (inbound and outbound).

1. Navigate to home page, and click **Ingest Connectors**.
2. Click **Add**. The **New Connector** window appears.

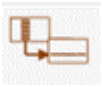
4.2 Icons and Description

This section describes the tools Financial Services Data Foundation Cloud Service for Banking provides to create connectors.

Table 4-1 Icons and Description

Icon	Description
	Click the Source button to view the list of all External Data Descriptors created in the setup. Use the **External Data Store** field to filter and the search icon to quickly locate the EDD you are looking for. After locating the EDD, you can drag the desired EDD on to the canvas.
	Click the Target button to view the list of all Application Data Interfaces (ADIs). Use the **Applications** field to filter and the search icon to quickly locate the ADI you are looking for.
	The Mapping component is used to map the source columns to target columns.
	The Join component is used to define a join between two entities. Double click this icon and provide the join condition information.

Table 4-1 (Cont.) Icons and Description

Icon	Description
	The Filter component is used for define the filter of a given entity. Double click this icon and provide the filter expression information.
	The Lookup component is used to define the lookup condition. Double click this icon and provide the lookup information.
	The Expression component is used to define defining the derived columns. Double click this icon to define an expression, which can be mapped to the target column.
	The Flattened Table To Hierarchy component is used to transform flattened hierarchy entities into parent-child hierarchy entities.
	The Transpose (Row to Column) component is used to transpose rows to columns for a given entity. Double click this icon to define the pivot data element and the new columns, which are transposed from multiple rows of the source entity.
	The Transpose (Column to row) component is used to transpose columns to rows for a given entity. Double click this icon to define the unpivot data element and new rows which are transposed from the columns of the source entity.
	The Aggregation component is used to define a group by and having a clause for aggregation. Double click this icon to define a group by and having a clause for aggregation.
	Click this button to remove all the nodes added to the canvas.
	The Reference Identifier Generator component is used to generate unique identifiers for DFCS dimensions even though source systems do not provide it.

4.3 Creating Connectors

Connectors allow you to map one or more External Data Descriptors with Application Data Interfaces. You can also map one or more ADIs with EDD using the extract type connector.

4.3.1 Ingesting Data into Data Foundation Cloud Service for Banking

To create a connector for ingesting data into Data Foundation Cloud Service for Banking, complete the following steps:

1. Navigate to the home page, click **Data Integration**, and then click **Ingest Connector**.

   The connectors which are defined are listed here.
2. From the LHS menu select **Connectors** if it is not already selected.
3. Click **Add**.

   The **New Connector** window is displayed.
4. To define a connector, you must have a source with EDD and a target, which is ADI.
5. Click **Source** to view the list of EDDs defined in the default EDS of the ERP File Extracts.
6. Select the default EDS-ERP File Extracts file type EDD and drag it on to the canvas.
7. Click **Target**.
8. Search for the ADI you are looking for.
9. Select the required ADI and drag it on to the canvas and then link the input and output nodes.
10. To link the nodes, right click on a component and click **Link to** and select the component you want to link it with. You can also use your mouse to link the nodes. Point your mouse on the white circle of the input node and with your left-mouse button pressed, move the cursor to the white circle on the output node and release the mouse button.
11. At any given time, you can right-click a node to either link, delink, remove inlinks/outlinks, or delete a node.
12. To edit or view the properties, on the Connector window, click the Next button on the top of the page.
13. Under **Connector Details**, enter the name and description for the connector.
14. Under **Pre-Load Options**, select the truncate option to be defined in the target.
    - Select one of the available truncate options if you want to remove data from the table. Select No, if you do not wish to truncate the table before loading.

    The Partial Truncate is not enabled and should not be chosen.

    > **Note:**
    >
    > For multi-target loads, the truncate type must be the same for all targets. However, truncate expression may vary.

    - Select **Full Truncate** to fully truncate. Example: To truncate the entire table. Here no expression is required.

- Select **Selected Rows** to truncate based on a specified filter expression. Click the edit button next to **Specify filter expression** to define the expression. Example; If you want to remove some rows, use the **Selected Rows** option. Specify the filter condition for the rows to be deleted. Those rows are removed from the table before load.
  - Click **Edit** to filter the selected rows.
  - Select the required entity and click **Validate**. This validates the expression.
  - Click **Ok** after the expressions are selected.

15. Under **Properties**, enter the value in **the Target Rejection Threshold** field. You can define the value in number or percentage of error records. For more details, see the Using Target Rejection Threshold. section.
16. Click **Save** or **Save As Draft**.

> **Note:**
>
> If one or more connectors are referring to the same File or EDD, then create a process pipeline to execute the connectors sequentially and they must not be executed in parallel.

5
Filter

This section provides information about the **Filter** component.

1. Drag and drop the **Filter** component onto the canvas to define a filter for an entity such as **EDD** (Insert Connector) or **ADI** (Extract Connector).
2. To position the component on the canvas, either drag and move it, or click the component and use the arrow keys on the keyboard to adjust its position.
3. If filter component accepts input only from an entity and can have only one output.
4. To apply filters to multiple entities, drag and drop the corresponding number of filters. Connect each filter to its respective entity and define their expressions. For example, to add a filter to three entities, place three filters on to the canvas and connect them accordingly.
5. At any time, right-click the filter component and select the relevant options to either delink, remove inlinks or outlinks, or delete the component.
6. To define the filter expression, double-click filter component. The **Filter Expression** window is displayed, showing selected entities and parameters. Specify the required filter expression using columns and parameters.
7. Click **Validate** to verify the correctness of the SQL Expression.
8. Click **OK** to confirm and apply the filter expression.

> **Note:**
>
> - You do not need to add a **WHERE** clause for the filter.
> - For **file data loading**, use the filter expression of the type **Number** with single quotes. For example: `N_DRAWN_AMOUNT = '40000'`.
> - For the **Date** field, see `To_CHAR` function for comparison.
> - Parameters can also be used in the filter expression. The date format must be a valid SQL date format. For Example: `[EDD_GL_DATA].[EXTRACTION_DATE] = TO_DATE(#DIHDEV.MIS_DATE,'dd-MM-yyyy')`

5.1 Join

This section provides information about the Join component.

1. Drag and drop the Join component on the canvas to link multiple entities such as EDDs (Insert Connector)/ADIs (Extract Connector).
2. To position the component on the canvas, drag and move the component, or click the component and use arrow keys on the keyboard to move it around.
3. Join accepts input from two entities.

4. To join more than two entities, drag another Join component. Link the output of the first join to the input of the second join and then connect the other entities. You can repeat this for multiple entities. Select the source entity and click **Ok**.
5. At any given time, right-click the component and select the relevant options to either delink, remove inlinks or outlinks, or delete the component.
6. Double-click the Join component to define a join condition. The selected entities are displayed on the left and right tabs.
7. You can drag and reorder the left and right tab to choose the right/left entity in a join condition.
8. To join entities, the select a column from the left, select a column from the right tab and click = (Add Join). This displays the joined entities. You can join multiple entities.
9. To remove two joined conditions, select the two columns from the left and right tabs, and click Remove Join. The joined condition is removed from the list.
10. Click **Reset** to reset all the joined conditions.
11. Click **Ok**.

> **Note:**
>
> This creates an inner join between the connected EDDs.

5.2 Lookup

This section provides information about the Lookup component.

1. Drag and drop the Lookup component on the canvas to lookup values from an entity.
2. To position the component on the canvas, drag and move the component, or click the component and use arrow keys on the keyboard to move it around.
3. The lookup component accepts input from two entities. One from the Value Entity and the other from the Lookup Entity.
4. At any given time, right-click the component and select the relevant options to either delink, remove inlinks or outlinks, or delete the component.
5. Double-click the lookup component to define a lookup condition. In the Lookup window, you will see the connected entities on the left and right tabs.
6. You can drag the lookup entities on the right and left to reorder them.
7. To specify the lookup condition, the select a data element from the left, select a data element from the right and click = (Add Join). The lookup condition is displayed at the bottom of the window. Do this for an many entities on the left and right tabs you want to define.
8. To remove a lookup condition, select data elements from left and right entities and click **Remove Join**.
9. Click **Reset** to reset the lookup condition.
10. Click **Ok**.

> **Note:**
>
> This creates a left outer join between the connected entities.

5.3 Aggregation

This section provides information about the Aggregation component.

1. Drag and drop the Aggregation component on the canvas to define an aggregation on an EDD.
2. To position the component on the canvas, drag and move the component, or click the component and use arrow keys on the keyboard to move it around.
3. Aggregation component accepts input only from an EDD and it can have only one output.
4. If you have multiple EDDs to be aggregated then you must select as many number of aggregation components, connect each to the respective EDD, and then define their group by having clauses. Example: To add aggregation to three EDDs, drag three aggregation components on to the canvas and link them to their respective EDDs.
5. At any given time, right-click the component and select the relevant options to either delink, remove inlinks or outlinks, or delete the component.
6. Double-click the Aggregation component to define an aggregation condition. In the Aggregation Window, you will see the selected EDD under the Entities tab. Double click the entities, functions, and operators to build your expression in the Group By field. Repeat this step to build your clause in the Having field.
7. Click **Validate** to verify the correctness of the SQL Expression.
8. Click **Ok**.
9. Click **Reset** to reset all the aggregation conditions and begin afresh.

5.4 Reference Identifier Generator

This section provides information about the Reference Identifier Generator component. Use this component to generate a unique sequence identifier for the selected attribute.

1. Drag and drop the **Reference Identifier Generator** component on the canvas.
2. To position the component on the canvas, drag and move the component, or click the component and use arrow keys on the keyboard to move it around.
3. At any given time, right-click the component and select the relevant options to either delink, remove inlinks or outlinks, or delete the component.
4. Double-click the component to define a reference identifier generator condition.
5. Provide a **Name**.
6. Specify the source attributes with unique values by selecting the relevant value from the drop-down list.
7. Specify **Identifier Type** as Numeric or Varchar.
8. Select the **Target Attribute**.
9. Click **OK**.

5.5 Transpose (Rows to Columns)

This section provides information about the Transpose (Rows to Columns) component.

1. Drag and drop the Transpose (Rows to Columns) component on to the canvas.
2. To position the component on the canvas, drag and move the component, or click the component and use arrow keys on the keyboard to move it around.
3. The Transpose (Rows to Columns) component accepts input only from an EDD and can have only one output.
4. If you have multiple EDDs selected, and you want to have a Transpose (Rows to Columns) component for more than one EDD, then you must select as many number of Transpose (Rows to Columns) components, connect each to its respective EDD, and then define their expressions.
5. At any given time, right-click the component and select the relevant options to either delink, remove inlinks or outlinks, or delete the component.
6. Double-click the component to transpose the entity rows into columns. Specify the pivot data element to transpose rows into columns.
7. Specify the Column name - matching row value – Expression combination. You must have a minimum of two combinations.
8. Click **Review** to review the transformation. The sample of the transformed data is displayed.
9. Click **Ok**.

5.6 Transpose (Columns to Rows)

This section provides information about the Transpose (Columns to Rows) component.

1. Drag and drop the Transpose (Columns to Rows) component on the canvas to define a Transpose (Columns to Rows) Component on an EDD.
2. To position the component on the canvas, drag and move the component, or click the component and use arrow keys on the keyboard to move it around.
3. The Transpose (Columns to Rows) component accepts input only from an EDD and can have only one output.
4. If you have multiple EDDs selected, and you want to have a Transpose (Columns to Rows) component for more than one EDD, then you must select as many number of Transpose (Columns to Rows) components, connect each to its respective EDD, and then define their expressions.
5. At any given time, right-click the component and select the relevant options to either delink, remove inlinks or outlinks, or delete the component.
6. Double-click the component to transpose the entity columns into rows. Specify the **Unpivot Data Elements** to transpose columns into rows.
7. Specify the **Header Column Name** and **Value Column Name**.
8. Specify the column value (**Header column**) and expression pair (**Value column**) for each transposed row. You must have a minimum of two pairs.
9. After specifying the **Unpivot Data Elements**, click **Auto Transpose**. This will transpose columns into rows based on the unpivot data elements selected.

10. Click **Review** to view the transformed data.
11. Click **Ok**.

5.7 Derived Column

This section provides information about the Derived Column component.

1. Drag and drop **Derived Column** component on the canvas.
2. To position the component on the canvas, drag and move the component, or click the component and use arrow keys on the keyboard to move it around.
3. Connect the output of the **Derived Column** component to the **Mapping** component.
4. At any given time, right-click the component and select the relevant options to either delink, remove inlinks or outlinks, or delete the component.
5. Double-click the **Derived Column** component and click **Add** to define a new derived column.
6. On the right pane, under **Entities**, double-click to select the entities which you want to use to build the expression. The selected entities appear in the Expression field at the bottom. Modify the expression based on your needs. Provide the Name for the derived column.
7. Click **Validate** to verify the correctness of the SQL Expression.
8. Click **Apply**. The saved details appear as a list on the left pane.
9. Click the **Edit** button to modify the name and expression of the required Derived Column.
10. Repeat the steps to create as many Derived Columns as you require.
11. Click **Ok**.

5.8 Mapping

This section provides information about the Mapping component.

1. Drag and drop the **Mapping** component on the canvas to define a mapping. Connect the inputs and outputs for the **Mapping** component before specifying the mapping details.
2. To position the component on the canvas, drag and move the component, or click the component and use arrow keys on the keyboard to move it around.
3. Double-click the **Mapping** component on the canvas. The EDDs, ADIs, and their respective data/derived data elements are displayed in the Mapping window.
4. Filter the source attributes by selecting an option from the drop-down list. Click a data element under **Source**.
5. Filter the target attributes by selecting an option from the drop-down list. Click a data element under **Target**. and then click the **Map** button. The column mappings are displayed.
6. Click the **Map** button. The mapped items are displayed on the right. column mappings are displayed.
7. Repeat the steps to map as many source-target pairs as required. Note that you can use the filter icons for the **Source** and **Target** lists to filter unmapped, mandatory, or those items that are valid for the application.
8. The following validations are performed for the mapping:
    a. Data type validation
    b. Data length validation

c. Data precision validation

9. The result of the validation for each mapping is indicated as **Remarks** next to each mapping.
10. At any given time, you can select the **Unmap** button to unmap the source and target.
11. Click **Auto Map** to auto map a source and target.

> **Note:**
>
> Auto-mapping is done by matching the logical/physical column name of both the source and target.

12. Hover your mouse over a data element under **Target** column to see detailed information which includes the description, length, and scale.
13. Use the search icons to search for source or target data elements.
14. Click the **Delete All** icon to delete all the mappings. You can also delete individual mappings by selecting the cross symbol next to the column mapping.
15. Click the **Import Mapping** icon to import a mapping Excel sheet.
16. Click the **Export Mapping** icon to export the mapping information in an Excel format.
17. Use the search field to search for mappings. You can search for an item based on the **Source Column Name**, **Target Column Name**, **Source Entity**, **Target Entity**, or **Remarks**.

5.9 Flattened Table to Hierarchy

This section provides information about the Flattened Table to Hierarchy component.

1. Drag and drop the **PC Hierarchy** component on the canvas.
2. To position the component on the canvas, drag and move the component, or click the component and use arrow keys on the keyboard to move it around.
3. Connect an EDD / Source Filter to the input of the **Flattened Table to Hierarchy** component.
4. Connect the output of the **Flattened table to Hierarchy** component to the **Join / Lookup / Mapping** component.
5. At any given time, right-click the component and select the relevant options to either delink, remove inlinks or outlinks, or delete the component.
6. To specify the details, double-click the **Flattened table to Hierarchy** component.

   The **Flattened Table to Hierarchy** window is displayed.
7. Choose the **Hierarchy Type** as **Balanced**, **Ragged**, or **Skipped**. Click **Help** icon to view the details and understand how the hierarchies are defined.
8. Specify the **Number of levels** in the hierarchy.
9. Specify the **Parent Node Column Name** and **Child Node Column Name**.
10. Select the **Key Elements** from the drop-down list. You can select multiple Key Elements for a hierarchy.
11. Select all the node level details from the respective drop-down lists.
12. Click **Review** to view the transformation changes.

13. Click **Ok**.

5.10 Hierarchy Data Flattening

The hierarchy flattening component in the Data ingestion definition allows you to use input files in a parent-child hierarchy structure and convert them into a flattened dataset during the ingestion process without any additional configuration outside the system. The application supports balanced, unbalanced, and skipped hierarchies for flattening.

This section provides information about the Hierarchy to Flattened Table component.

1. Drag and drop the **Hierarchy to Flattened Table** component on the canvas.
2. To position the component on the canvas, drag and move the component, or click the component and use arrow keys on the keyboard to move it around.
3. Connect an EDD / Source Filter to the input of the **Hierarchy to Flattened Table** component.
4. Connect the output of the **Hierarchy to Flattened Table** component to the **Join / Lookup / Mapping** component.
5. At any given time, right-click the component and select the relevant options to either delink, remove inlinks or outlinks, or delete the component.
6. To specify the details, double-click the **Hierarchy to Flattened Table** component.

   The **Hierarchy to Flattened Table** window is displayed.
7. Select **Key Elements** from the drop-down list. You can select multiple Key Elements for a hierarchy. Parent or Child attributes cannot be selected as Key Elements.
8. Specify the maximum **Number of levels** in the hierarchy. By default, the number of levels is set to 3.
9. Select the **Parent Identifier**, **Parent Code**, and **Parent Name**.
10. Select the **Child Identifier**, **Child Code**, and **Child Name**.
11. Click **Review** to view the transformation changes.
12. Click **Ok**.

5.11 Specifying Alias in Connector

Alias refers to an assumed name or pseudonym assigned to an EDD as you define connectors, much like table aliases used in SQL statements. Aliases allow an EDD to be referred in multiple joins, lookups, or both, within the same connector definition, each in a distinct context.

Aliases are automatically initiated by Data Integration Hub (DIH) as and when required while defining connectors.

The following scenario explains the mechanism:

1. Drag and drop an EDD more than once, for use in separate joins or looks-up, on the **New Connector** canvas. DIH initiates the specification of an alias by displaying the **Specify Alias** dialog box.
2. Enter a name of your choice under **Alias Name** and click **Ok**.

   Note that the Alias Names must be unique within a connector.

5.12 Modifying and Viewing a Connector

To edit or view a connector, complete the following steps:

1. To edit or view a connector, you can select the required connector from the Connector Summary.
2. The details of the selected connector are displayed. You can modify or view the details.
3. Modify the connector's details as required. Connector Name cannot be edited.
4. Click **Save**.
5. To make changes to a published connector, click **Unpublish**. This clears the ODI metadata that has been created during publishing. Update the required changes and then click **Publish**. The updated changes are synced in ODI.

5.13 Copying a Connector

To copy an existing connector, complete the following steps:

1. Click the Copy button of the required connector. Depending on the view in which the original connector was created, the copied connector too will have the same view.
2. Enter the **Name** and **Description**.
3. Click **Save**. The details are saved with a new connector name. The existing connector remains unmodified.

5.14 Deleting a Connector

To delete an existing connector, complete the following steps:

1. Click the **Delete** button of the required connector.
2. Acknowledge the confirmation message.

5.15 Search and Filter

Use the search and filter options in the Connector Summary page to find the required connector. Enter the nearest matching keyword to search, and filter the results. You can search for a connector with either the name, description, or status of the connector.

You can sort the list by connector name or modified date in ascending or descending order.

5.16 Parameters in Connector

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

For example, if you want to use the Runtime MIS Date as the parameter, you must first convert it to date. The following is the expression:

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

> **Note:**
>
> Runtime batch MIS date is in String format. You must convert it to a valid SQL date format.

5.17 Using Target Rejection Threshold

Target Rejection Threshold is a value defined for the number or percentage of records with errors that are allowed before the execution is configured to fail when you move records from the source database to the target database. It can be entered as an absolute value or as a percentage in the **Connector Properties** window.

DFCS creates an error table by duplicating the target table name and appending it with **_ERR** during the execution process. The erroneous records are logged into the error table and valid records are moved to the relevant table in the target. The errors can be of the types notional, public key, or data. For example, if the target table name is STG_CASA, then the error table name is created as STG_CASA_ERR. The objective is to log only a required number of errors and then correct them in the source table. After correcting the errors, rerun the execution. In the process, you can iteratively collect the errors and correct them, making the task of correcting the errors in the source table much simpler.

You can also move records from file table to the target database and the structure of the table remains same as file table added with five additional columns. For example, if the file table name is EDD12_THRESHOLD_VOLUMETEST, then the error table name is created as EDD12_THRESHOLD_VOLUMETEST_ERR.

Let us understand this process with the help of the following example.

1. Enter 10 in the Target Rejection Threshold field.

   > **Note:**
   >
   > a. To specify the value as 10 percent of the records in the execution, enter 10%.
   >
   > b. If you leave the field blank or enter 0, DFCS reads the threshold as 0 and the execution fails when an error is encountered.
   > On execution of the process, DFCS permits up to 10 records with errors to process to the target table and any number of valid records. If the number exceeds 10, the execution process fails and stops. The erroneous records are available in the target table name appended with _ERR.

2. Click **Data Ingestion** and select **View Data**. This will display **Data Visualization** window where you can search for the target table name appended with _ERR.
3. Check the details in the error table. For more information, see Reading the Error Table.
4. After checking the error table details, navigate to the source table, and troubleshoot the errors.
5. After correcting the errors, rerun the execution.
6. Repeat the process iteratively and correct the errors.

5.17.1 Reading the Error Table

DSFCS creates the error table in the data service execution layer/schema to store records with errors. The structure of the error table is similar to the target table but with five additional columns. You can refer to the following column for executions:

- ORA_ERR_NUMBER$ - The error number.
- ORA_ERR_MESG$ - The ORA error message with the description of the error.
- ORA_ERR_ROWID$ - The row ID of the error.
- ORA_ERR_OPTYP$ - The operation type.
- ORA_ERR_TAG$ - This column stores the Execution ID. You can use this column to filter and view error records for a specific execution.

5.18 Executing Connectors

Use the Process Modelling Framework to execute a DIH connector.

5.18.1 Executing Connector Using Process Modelling Framework

Process Modelling Framework (PMF) is a design and execution framework that enables process pipeline developers to implement the various pipelines modelled by business analysts. Process pipeline developers use the framework to orchestrate business, run pipelines within DFCS, and to design the artifacts that participate in the pipelines, to complete their implementation. For more details, see the Managing Process section.

To add a connector task in the PMF pipeline, complete the following steps:

1. On the home page, select **Data Pipelines under Data Integration** link on the left.
2. Search for and open the process pipeline to which you want to add a connector task.
3. Click to expand the left menu. Drag and drop **Connector** under **Widget** to the canvas.
4. Double-click the **Connector** and specify the **Activity Name** and **Activity Description**.
5. Under the Dynamic Parameters for Connectors:
    - Select the required connector.
    - Enter the runtime **Variables**. If the connector contains any runtime parameter, see the section: Specifying Runtime Parameters, for more details.
6. Enter all details and click **Save**.

5.18.2 Specifying Runtime Parameters

Note the following points regarding runtime parameters:

- If the connector contains runtime parameters, they can be set in the Variables input field of the connector's definition window. Example: FILE_DATE=31-Jan-2022.
- In this example, the date format appended to MISDATE has to conform to the Simple Date Format. If no date format is specified, the default date format used is yyyyMMdd.
- If variables are used as a part of connector mappings or filter expressions, specify them as given in this example. Example: FILE_DATE=$MISDATE:dd-MMM-yyyy.

5.19 Quality

For information about the data quality checks (DQ Checks) and the out-of-the-box pipelines, see the Data Quality Checks section in the *Data Foundation Cloud Service for Banking User Guide*.

5.20 Protection

This section explains the General Data Protection Regulation (GDPR) related data protection methods implemented in Data Foundation Cloud Service (DFCS).

5.20.1 Data Redaction

Data Redaction is one of the data security features that protects sensitive data against unauthorized access and data theft.

Data Redaction is an in-built process in DFCS. It is applied automatically on all the business terms containing Personally Identifiable Information (PII).

To implement data redaction on a business term, on the **New Business Term** page, enable the Personally Identifiable Information button for that business term. For information about business term creation, see the Manage Business Terms section in the *OFS DFCS Data Catalog User Guide*.

Index