Oracle® Financial Services Data Foundation Cloud Service for Banking

Data Services



Release 26C
G30800-28
August 2026

ORACLE

Oracle Financial Services Data Foundation Cloud Service for Banking Data Services, Release 26C

G30800-28

Copyright © 2000, 2026, Oracle and/or its affiliates.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software, software documentation, data (as defined in the Federal Acquisition Regulation), or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle®, Java, MySQL, and NetSuite are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Oracle customer access to and use of Oracle support services will be pursuant to the terms and conditions specified in their Oracle order for the applicable services.

Contents

1 Data Services

2 Managing Source Data

2.1 External Data Descriptor (EDD) 1
2.1.1 Defining an External Data Descriptor (EDD) 10
2.1.2 Search and Filter 10
2.1.3 Modifying and Viewing an External Data Descriptor (EDD) 10
2.1.4 Deleting an External Data Descriptor 11
2.1.5 Parameters in EDD Definition 11

3 Catalog Lookup Object

3.1 Browsing Catalog Lookup Objects 1
3.2 Defining Catalog Lookup Objects 2
3.2.1 Using Catalog Lookup with a Dimension Entity 4
3.2.2 Using Catalog Lookup with a Map 6
3.2.3 Using Catalog Lookup with a Fact Entity 7
3.2.3.1 Using Catalog Lookup for Dimension, Map, and Fact (Without Aggregation) 8
3.2.3.2 Using Catalog Lookup with Aggregation for Fact Entities 8
3.2.4 Using Catalog Lookup with Dimension as Sourced 9

4 Data File Encryption

4.1 Configuring Administrator Key for Data File Encryption 1
4.2 External Data Descriptor - Encrypted Files 2
4.3 Troubleshooting Encryption Error 2

5 Data Trace

5.1 Viewing Data Trace from Application Connectors 1
5.2 Viewing Data Trace from Ingest Connectors 2

Copyright © 2000, 2026, Oracle and/or its affiliates.

6 Ingest Connectors

6.1 Icons and Description 1
6.2 Ingesting Data into Data Foundation Cloud Service for Banking 2
6.3 Using Join (Inner Join) 4
6.4 Using Filter 4
6.5 Using Lookup (Left Outer Join) 5
6.6 Using Reference Identifier Generator 6
6.7 Using Aggregation 6
6.8 Using Transpose (Rows to Columns) 7
6.9 Transpose (Columns to Rows) 7
6.10 Derived Column 8
6.11 Mapping 8
6.12 Flattened Table to Hierarchy 9
6.13 Hierarchy Data Flattening 10
6.14 Specifying Alias in Connector 10
6.15 Modifying and Viewing a Connector 11
6.16 Copying a Connector 11
6.17 Deleting a Connector 11
6.18 Search and Filter 11
6.19 Parameters in Connector 11
6.20 Using Target Rejection Threshold 12
6.20.1 Reading the Error Table 13
6.21 Executing Connectors 13
6.21.1 Executing Connector Using Process Modelling Framework 13
6.21.2 Specifying Runtime Parameters 13
6.22 Quality 13
6.23 Protection 13
6.23.1 Data Redaction 14

7 Application Data Services (DFCS Only)

7.1 Setting up Application Data Services 1
7.1.1 Create A New User in DFCS IDCS Console for ADS Subscription 1
7.1.2 Assign User Access for ADS Subscription 2
7.1.3 Get the OAuth Client ID and Client Secret for the User 3
7.1.3.1 ADS Credential Validation through the IDCS Integrated Application 3
7.2 Managing Application Data Service Definitions 6
7.3 Subscribing Application Data Services 8
7.3.1 Application Data Services Version Compatibility 8
7.3.2 Viewing Data Services Subscription Dashboard 10
7.3.3 Steps to Subscribe to Application Data Services 11

Copyright © 2000, 2026, Oracle and/or its affiliates.

7.3.4 How to Refresh Extensions and Updates from DFCS 16
7.4 Published Application Data Services 17
7.4.1 ADS Run History 17

8 Integration with Oracle Applications

8.1 Integration with Oracle Enterprise Data Management Cloud Service 1
8.1.1 Pre-Configured Standard Dimensions 1
8.1.2 Validations in EDMCS 2
8.1.3 Managing Dimensions and Hierarchies in EDMCS 3
8.1.4 Steps for Integrating EDMCS with DFCS 7

9 Data Ingestion Errors Browser

9.1 Accessing Data Ingestion Errors Dashboard via DFCS 3
9.2 Understanding EDD (External Data Descriptor) within a Application Connector 4
9.3 Common Error Types 6

10 Data Extraction

10.1 Accessing Data Extraction 1
10.2 Adding a Data Extraction 1
10.3 Copying a Data Extraction 5

11 Metadata Report

11.1 Accessing Metadata Export 1
11.2 The Metadata Export Summary Page 1
11.3 Creating a New Metadata Export 2
11.4 Viewing or Editing an Export 2
11.5 Generating a Report 2
11.6 Downloading the Report and Logs 3

12 Dimension Transformation

12.1 Adding a Custom Dimension 2
12.2 Execution through PMF 4
12.3 For Key Dimensions 7
12.4 For Simple Dimensions 9
12.5 Clear and Back up Dimension Data 11
12.6 Data Preparation Guidelines 12

Copyright © 2000, 2026, Oracle and/or its affiliates.

13 Data Pipelines

14 Key Terms and FAQs

Copyright © 2000, 2026, Oracle and/or its affiliates.

1

Data Services

Data Services facilitates seamless data exchange between ERF Applications and external systems by providing a logical abstraction of the OFSAA Data Foundation—which includes the Financial Services Data Foundation —through Application Data Interfaces (ADIs). Using the Data Services user interface, users can define External Data Descriptors (EDD). These EDDs can then be mapped to ADIs to create Connectors, enabling structured and efficient data flow across systems.

Copyright © 2000, 2026, Oracle and/or its affiliates.

2

Managing Source Data

The **Data Source Templates** section in the **Data Services** module is designed to manage data exchange between the **Financial Services Data Foundation Cloud Service for Banking** and downstream applications. This is accomplished through a logical abstraction layer that exposes the Data Catalog as **Application Data Interfaces (ADIs)**.

Users can define **External Data Descriptors (EDDs)** via the interface, and then map these EDDs to ADIs to create **Connectors** that enable smooth and efficient data services Additionally, the interface provides a summary of all existing templates, displaying key details such as the creator and the date each template was created.

To navigate to the Data Source Template, follow the below steps.

1. From the **Data Foundation for Banking** window, click **Data Services** > **Data Source Templates**.
   The Summary page displays the list of External Data Descriptor (EDD).

> ⓘ **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.

2.1 External Data Descriptor (EDD)

External Data Descriptors (EDD) are definitions of specific data content from External Data Sources (EDS). **EDD** helps define how data from **External Data Sources (EDS)** should be structured and processed within the **DFCS**.

- It enables:
  - **Ingestion** of external data into DFCS.
    Extraction of data from DFCS.
  - **Control information** management during data ingestion.

Each EDS can have multiple **EDDs**, each tailored to a specific data format or control logic.

1. On the **Financial Services Data Foundation Service for Banking** home page, click **Data Source Templates**.

   The **Summary** page is displayed.
2. Click **Search** to search the existing Data Descriptor from the available list. Sort by clicking on the **Name** drop down.
   - Name
   - Last Modified Date
3. Click **Add** to configure additional EDDs.
4. Enter the **Specify Data Descriptor name** and **Describe the Data Descriptor** details.

Copyright © 2000, 2026, Oracle and/or its affiliates.

5. **Configure File Settings** (under the *Data* tab):

Table 2-1 Data Tab Description

Fields	Description
Toggle **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. **Note** The Archive file name field appears 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.
Toggle **Is data files(s) encrypted?**	This option allows you to indicate whether the incoming data file is encrypted. Enable this toggle only if the file you are ingesting has been encrypted using the platform's supported encryption method. When turned **ON**, the system will automatically decrypt the file using the configured encryption key before processing and loading the data. If left **OFF**, the system will treat the file as a plain (unencrypted) file.
Specify data file names	Click on **Add More Files**. Enter the file name and click **OK**. 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

Copyright © 2000, 2026, Oracle and/or its affiliates.

Table 2-1 (Cont.) Data Tab Description

Fields	Description
Select 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.
Select 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.
Specify Text Qualifier (Optional)	A character that identifies text. Generally, double quotes are prefixed and suffixed to identify text. This is optional.
Specify the number of records to skip	Provide the number of records to be skipped. The records are skipped from the top. Generally, this is used to skip headers.
Specify Decimal Separator	The character used to identify the decimal and fractional part. Usually point (.) or comma (,)
Do you want to read from template?	(Optional) Enable **Do you want to read from template?** to auto-populate 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. **Note** CSV files can support UTF-8 encoding, which allows for special characters and non-Latin text. To ensure these characters are preserved correctly, make sure to save the file with UTF-8 encoding.
Data Elements	If not using a template, manually **add data elements** using the **Add** option.

Copyright © 2000, 2026, Oracle and/or its affiliates.

Table 2-1 (Cont.) Data Tab Description

Fields	Description
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.
Scale	This is used to specify the number of digits after the decimal point. Example: 10.3.
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.

6. **Steps to Configure Control Settings in EDD**: 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

Copyright © 2000, 2026, Oracle and/or its affiliates.

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.

7. The Control tab supports descriptive metadata for the source file, in addition to record-count and amount reconciliation. You can capture attributes such as the file name, creation date, and source system by using the same key-value pair format as control information. To enable descriptive metadata, select **Does the reconciliation include descriptive metadata**.
8. The threshold can be an absolute or a percentage value. The connector execution process considers the threshold type while performing reconciliation.
9. 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).

10. Descriptive metadata provides additional information about an input file, such as the source system, file creation date, business date, or file name. Metadata records are extracted during connector execution. They are not used for reconciliation unless the same key is explicitly configured as a control.

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. Records matching the configured Control Names are processed as control records. Other valid key-value records are processed as descriptive metadata. The control file should have only two columns that are Control Name and Expected Value.	-
When **Separate File** is selected as **Yes**.	Select Yes for file separation.
Does the reconciliation include descriptive metadata	Enable this toggle when the control information contains descriptive metadata. When disabled, metadata is not identified or stored.
Metadata Record Type	Identifies metadata records when record types are used. Enter a value distinct from the Data Record Type and Control Record Type values.
Control and Metadata Length Properties	For fixed-length files, configure the length properties required to parse the control/metadata record structure. These properties are not required for delimited files.
File Name	Specify the name of the file.

Copyright © 2000, 2026, Oracle and/or its affiliates.

Field	Description
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 *Other*, enter a single character to be used as the delimiter.
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.
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 Records 400" 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'.

Copyright © 2000, 2026, Oracle and/or its affiliates.

Field	Description
When **Are Reconciliation details present in Separate File** option is disabled, the below fields will be enabled.	No
Is Reconciliation file encrypted?	This option indicates whether the reconciliation file is encrypted. Enable this toggle if the reconciliation file has been encrypted. When turned ON, the system will decrypt the file using the configured encryption key before processing. If left OFF, the file will be treated as a plain (unencrypted) file.
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.
Specify control and metadata value length	For a fixed-length file, specify the length of the Control or Metadata Value field.
Specify control and metadata name length	For a fixed-length file, specify the length of the Control or Metadata Name field.
Metadata Record Type Code	Specify the code used to identify metadata records in the file. For example, METADATA.
Specify Record Type Length	For a fixed-length file, specify the length of the Control or Metadata Value field.
Does the reconciliation include descriptive metadata Yes	Enable this option to extract and store descriptive metadata available in the control file. Metadata must be provided in the same key-value format as control information. If the Separate file is Yes, provide all the file related information.
Specify control and metadata value length	For a fixed-length file, specify the length of the Control or Metadata Value field.
Specify control and metadata name length	For a fixed-length file, specify the length of the Control or Metadata Name field.
Metadata Record Type Code	Specify the code used to identify metadata records in the file. For example, METADATA.
Specify Record Type Length	For a fixed-length file, specify the length of the Control or Metadata Value field.
Metadata Record Type Code Behaviour Configuration	-
Data, control and metadata are present in the same file	Data, Control and Metadata Record Type Codes are mandatory and must have different values
Reconciliation details are present in a separate file	Control and Metadata Record Type Codes must be unique. If one Record Type Code is specified, the other must also be specified.
Controls	-
Control Name	Specify the name of the control.

Copyright © 2000, 2026, Oracle and/or its affiliates.

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 (Optional)	Threshold controls allow users to define acceptable limits for error records during batch execution. This helps enforce stricter validation by ensuring the batch fails when the number of rejected records exceeds the configured limit. Users can configure the threshold using one of the following options: • **Absolute Value** – Defines the maximum number of rejected records allowed. The absolute percent difference is matched against this threshold value. • **Percentage** – Defines the maximum percentage of rejected records allowed relative to the total number of processed records. The reconciliation difference in percent is matched against this threshold value. Threshold rules can be configured in the **Control** tab of the External Data Descriptor (EDD). These rules define the acceptable limits for rejected records during batch execution. **Table 2-2 Example** **Control Name** / **Control Type** / **Threshold Type** / **Threshold Value** Test / Count / Absolute / 10 • Up to **10 rejected records** are allowed during processing. • If the number of rejected records **exceeds 10**, the **PMF execution fails** even if valid records are loaded into the target table.
Threshold Value	Specify the difference value in percent or absolute.

Copyright © 2000, 2026, Oracle and/or its affiliates.

> **Note**
>
> - If descriptive metadata is enabled but metadata records are not available in the file, connector execution continues with a warning.
> - Metadata records are stored separately and do not affect control reconciliation.
> - The Metadata records will be ingested into the Execution File Metadata table.

11. Click on the **Transformation** tab at the top of the EDD configuration screen.

Table 2-3 Transformation Tab Description

Fields	Description
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.
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.

Copyright © 2000, 2026, Oracle and/or its affiliates.

12. Click **Save as Draft** to store your progress or **Save** to finalize the descriptor once all required fields are completed.

2.1.1 Defining an External Data Descriptor (EDD)

To define a new **External Data Descriptor (EDD)**, 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.
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**.

2.1.2 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**.

2.1.3 Modifying and Viewing an External Data Descriptor (EDD)

You can edit or view existing EDDs.

Copyright © 2000, 2026, Oracle and/or its affiliates.

> **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**.

2.1.4 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 page, click the **Dependencies** icon to check if the order has any dependency.

   A window opens showing whether the order has any linked or prerequisite dependencies.
2. If there are no dependencies, then click **Delete** corresponding to the EDD you want to delete.
3. Acknowledge the confirmation message.

   The EDD details are deleted.

2.1.5 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 2-4 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).

Copyright © 2000, 2026, Oracle and/or its affiliates.

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.

Copyright © 2000, 2026, Oracle and/or its affiliates.

3

Catalog Lookup Object

The **Catalog Lookup Object** allows users to retrieve existing data from the Data Foundation catalog during data ingestion instead of requiring the source system to provide the same information again.

The Catalog Lookup object enables users to create and manage lookup definitions within data services definitions. With Catalog Lookup, users can define **key-value mappings** using entities such as **Dimensions, Facts, or Maps**. A key field from the incoming source data is used to search the catalog, and the corresponding target value is retrieved and populated.

3.1 Browsing Catalog Lookup Objects

The Catalog Lookup Objects Summary page in Data Services displays all the available Catalog Lookup Objects created in the system. This page enables users to search, view, manage, and organize lookup objects used for retrieving data from target entities.

1. Navigate to **Data Services** section and click **Catalog Lookup Objects**.

   The Catalog Lookup Summary page displays all the available lookup objects.

2. From the summary page, you can do the following.

Field	Description
Search Bar	Allows searching lookup objects by name, description, or editor.
View Options	• Grid View • List View
Sort Option	Sort lookup objects by Name (A to Z or Z to A) and last modified date.
Lookup Object Name	The name of the lookup object.
Description	A short description of the lookup object.
Status	Displays the current status such as **Saved**.
Created	Displays when the lookup object was created and the user who created it.
Last Updated	Displays when the lookup object was last modified and the user who modified it.

Copyright © 2000, 2026, Oracle and/or its affiliates.

Field	Description
Delete	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. If the lookup is referenced by one or more connectors, the **Delete** icon is not displayed. For example, assume a Catalog Lookup is used in a Connector. Then, unless the Connector is deleted, the used Catalog Lookup cannot be deleted. To delete an existing Catalog Lookup, complete the following steps: a. On the Catalog Lookup Summary, click **Delete** corresponding to the Catalog Lookup you want to delete. b. Acknowledge the confirmation message. The Catalog Lookup details are deleted.

> **Note**
>
> Users cannot modify or edit Catalog Lookup Objects.

3. From the Catalog Lookup Summary list, select any catalog lookup object.

   A details panel opens on the right side of the screen. For more information on adding catalog lookup objects, see Ingesting Data into Data Foundation Cloud Service for Banking.

3.2 Defining Catalog Lookup Objects

Catalog Lookup objects can only be defined within the context of a target entity in an ingest connector. Therefore, to create a new catalog object, the user must go to the **Ingest Connectors UI** and follow the below steps.

1. Navigate to **Home** page, under **Data Services** > **Ingest Connectors**.
2. Click **Add** to display the new connector window.

> **Note**
>
> A Connector cannot be created using only a Catalog Lookup Object. At least one Source External Data Descriptor (EDD) must be defined in the connector.

3. In the connector canvas, click **Target** to display all entities.
4. Drag and drop a target entity (for example, Casa) onto the canvas. Ensure the target entity is successfully placed in the data flow.
5. Right-click the target entity on the canvas.
6. From the context menu, select **Create Catalog Lookup**.

   The Create Catalog Lookup dialog box is displayed.

Copyright © 2000, 2026, Oracle and/or its affiliates.

Table 3-1 Create Lookup

Field	Description
Name	Enter a name for the lookup.
Description	(Optional) Enter a description in the Description field.
Context	Review the Context field. This field is auto-populated based on the selected target entity and cannot be edited.
Select the Target Attribute	From the Select Target Attribute list, choose the attribute you want to populate using the lookup. A Business Term (BT) from the Target Entity is selected as the Target Attribute for the lookup.
Lookup Entity Type	Under Lookup Entity Type, select one of the following: • **Dimension**: Uses the SCD populated dimension table as the lookup entity. The system automatically selects the Dimension table Business Key and Numeric Identifier as the default key fields for the lookup. For more information, see Using Catalog Lookup with a Dimension Entity. • **Map**: Uses Mapper History entities as the lookup entity. The system automatically selects the Mapper Dimension source fields as the default key fields for the lookup. For more information, Using Catalog Lookup with a Map. • **Fact**: Uses stage entities from the catalog as the lookup entity. The system automatically selects the Stage table Primary Key as the default key field for the lookup. For more information, Using Catalog Lookup with a Fact Entity. • **Dimension as Sourced**: Select this entity type to use Master Data as sourced. The system automatically selects the Stage Master table Primary Key and Numeric Identifier as the default key fields for the lookup. For more information, Using Catalog Lookup with Dimension as Sourced.
Lookup Entity	From the Select Lookup Entity list, choose the entity that contains the reference data. **Note** Only entities that support the selected target attribute are displayed.

Copyright © 2000, 2026, Oracle and/or its affiliates.

Table 3-1 (Cont.) Create Lookup

Field	Description
Manual Selection of Group by Key Fields	• Key fields are auto-selected based on the selected Lookup Entity Type. • Users can manually select up to 8 Business Terms (BTs) as Lookup Key Fields. – Dimension: Business Key and Numeric Identifier. – Dimension as Sourced: Stage Master Table's Business Key and Numeric Identifier. – Fact: Stage Table's Business Key. – Map: Mapper Dimension's source fields.
Lookup Value Field	From the Lookup Value Field list, select the field whose value will be used to populate the target attribute.

7. Click **Save** to save the Catalog Lookup Object details and a confirmation message is displayed. The newly added catalog lookup object is listed in the **Catalog Lookup Object** Summary page.

   The newly added Catalog Lookup Object is also listed as a new source on the LHS of the Connector User interface in the Data Descriptor list. The Catalog Lookup Objects will appear along with the EDD objects in this list. Each Catalog Lookup Object is represented with an icon marked "CL" in dark blue color to distinguish it from the EDD objects.

   To know more about using catalog lookup object, see Ingesting Data into Data Foundation Cloud Service for Banking.

3.2.1 Using Catalog Lookup with a Dimension Entity

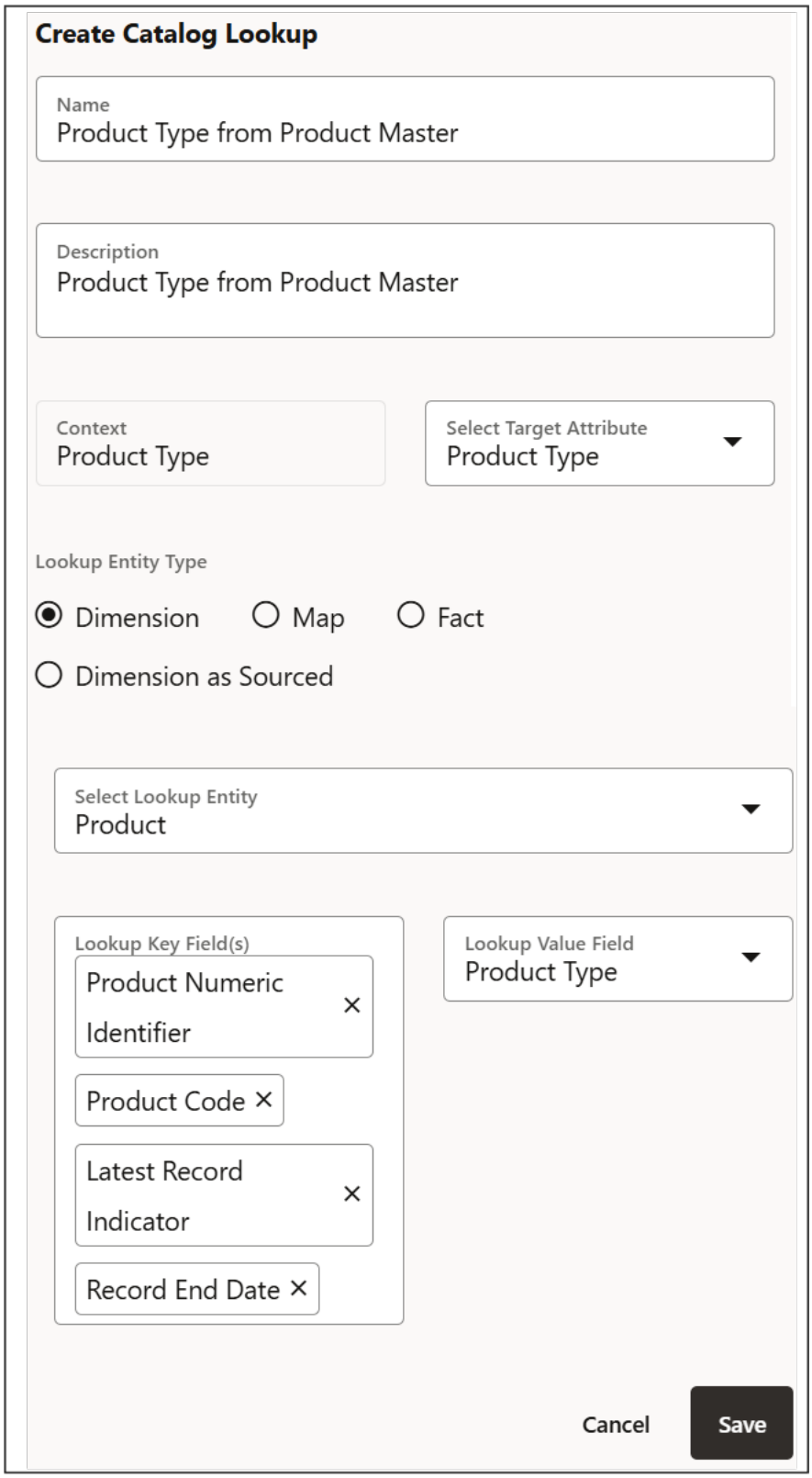
Used to fetch attributes from **dimension tables**.
Example:

- Product Dimension
- Customer Dimension
- Branch Dimension

1. Create a Catalog Lookup Object in the Data Service and select the required Dimension entity (for example, Product Dimension).

Copyright © 2000, 2026, Oracle and/or its affiliates.

Figure 3-1 Dimension as an Entity

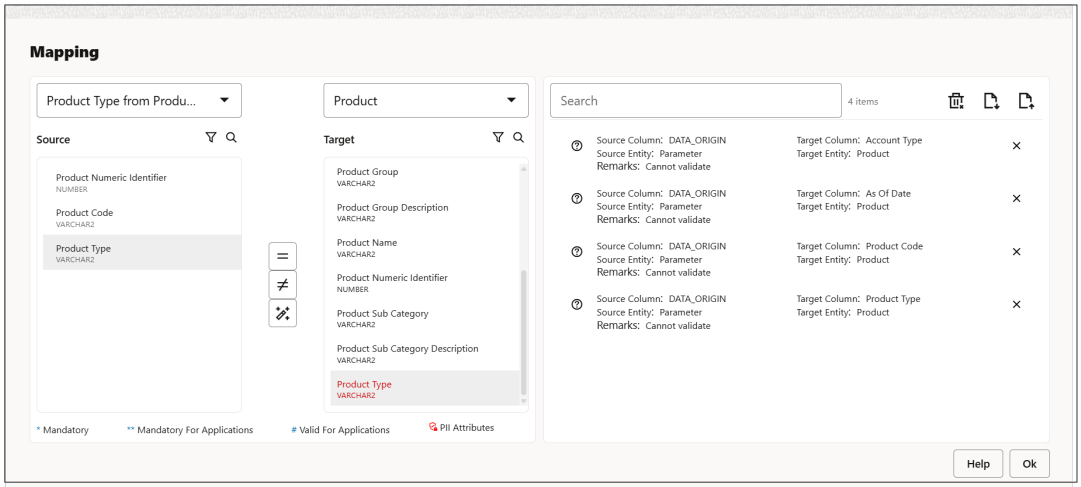


2. Define the key field (for example, Product Code) and select the lookup field to retrieve (for example, Product Type).

Copyright © 2000, 2026, Oracle and/or its affiliates.

3. If the dimension contains historical records, include filters such as Latest Record Indicator or Record Start/End Date to avoid duplicate results.
4. In the Connector Source configuration, add the Source EDD and the Catalog Lookup Object.
5. Define a join condition between the source field and the lookup key field (for example, Source Product = Product Code). Up to 8 Business Terms can be selected as Lookup Key Fields.
6. In Mapping, map the retrieved lookup value (for example, Product Type) to the required target field.

Figure 3-2 Mapping the Dimension Entity



3.2.2 Using Catalog Lookup with a Map

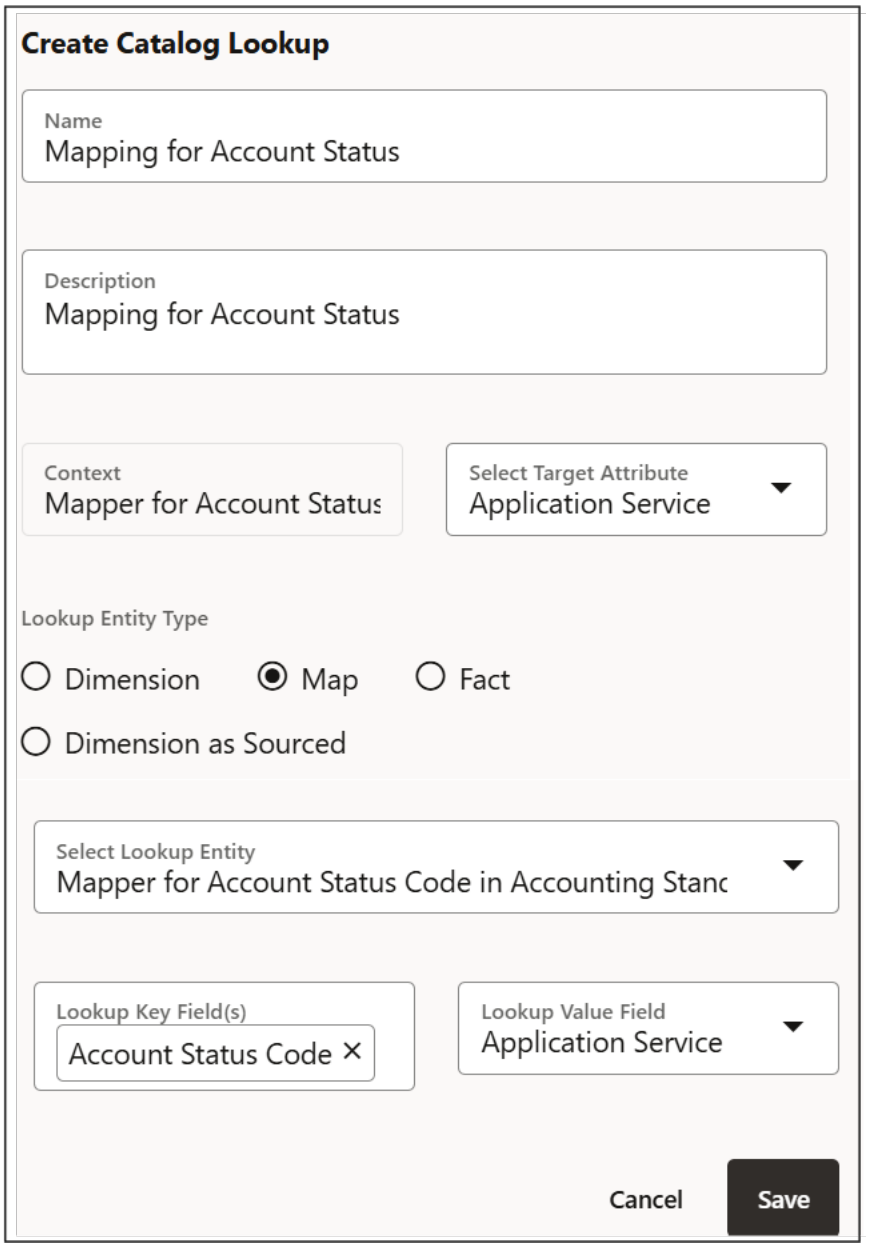
Used when lookup data is stored in **mapping tables**.
Example:

- Currency Mapping
- Country Mapping
- Code Translation Tables

1. Create a Catalog Lookup Object pane, select the required lookup entity type as Map (for example, Account for Mapper Status).

Copyright © 2000, 2026, Oracle and/or its affiliates.

Figure 3-3 Map Lookup Entity Type



2. Define the key field (for example, Account Status Code and select the lookup field to retrieve (for example, Application Service).
3. In the Connector Source configuration, add the Source EDD and the Catalog Lookup Object.
4. Add a Aggregation condition between the source field and the lookup key field (for example, Mapping for Account Status = Account Status (ADI).
5. In Mapping, map the retrieved lookup value (for example,) to the required target field.

3.2.3 Using Catalog Lookup with a Fact Entity

Used to retrieve information from **fact tables**.
Example:

Copyright © 2000, 2026, Oracle and/or its affiliates.

- Transaction Facts
- Balance Facts

1. Create a Catalog Lookup Object pane, select the required Fact entity (for example, Fact Entity Allocation).
2. Define the key field (for example, Allocation Fact and select the lookup field to retrieve (for example, Fact).
3. In the Connector Source configuration, add the Source EDD and the Catalog Lookup Object.
4. Add a Aggregation condition between the source field and the lookup key field (for example, Entity Allocation Factor = Allocation Fact (ADI).
5. In Mapping, map the retrieved lookup value to the required target field.

> **Note**
>
> If the source data type and target data type selected do not match, a warning sign is displayed.

6. Click **Auto-Map** to match the source and target data types.

3.2.3.1 Using Catalog Lookup for Dimension, Map, and Fact (Without Aggregation)

Users can configure Catalog Lookup Objects to perform lookups on **Dimension**, **Map**, and **Fact entities (without aggregation)**.

- Catalog Lookup Objects appear as selectable sources, similar to EDD.
- Users can select and apply a Catalog Lookup Object within a connector.

The following features are supported when configuring the lookup:

- **Join**
- **Filter**
- **Left Outer Join (Lookup)**
- **Expression**
- **ID Generator**

3.2.3.2 Using Catalog Lookup with Aggregation for Fact Entities

Users can also configure Catalog Lookup Objects with aggregation logic for Fact entities to perform grouped lookups..

- The aggregated Key Value Business Term can be used during mapping to the target entity.
- Key Fields ensure correct lookup execution across connectors.
- The same Catalog Lookup Object can be reused across multiple connectors with different group-by combinations.
- Users can select and apply a Catalog Lookup Object within a connector.

Copyright © 2000, 2026, Oracle and/or its affiliates.

3.2.4 Using Catalog Lookup with Dimension as Sourced

Uses master data directly from the stage master table to fetch attributes from **dimension as sourced tables**.

1. Create a Catalog Lookup Object in the Data Service and select the required Dimension as Sourced (for example, Data Source for Dimension).
2. Define the key field (for example, Product Code) and select the lookup field to retrieve (for example, As of Date).
3. In the Connector Source configuration, add the Source EDD and the Catalog Lookup Object.
4. Add a Aggregation condition between the source field and the lookup key field (for example, Data Source for Dimension as Source = Data Source Numeric Identifier (ADI).
5. In Mapping, map the retrieved lookup value (for example, Data Source) to the required target field.

Copyright © 2000, 2026, Oracle and/or its affiliates.

4 Data File Encryption

In scenarios that require encrypted data files to be exchanged between applications and servers, DFCS now offers support for ingesting encrypted source files as part of this release. Administrators are given full control over encryption key management, including the ability to perform key rotation.

Encryption is optional and remains disabled by default. It becomes available only after an Administrator configures a valid encryption key.

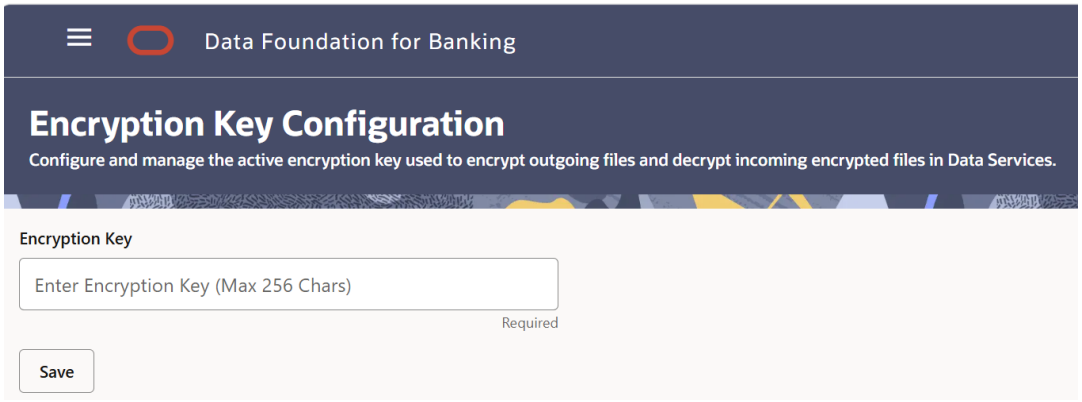
Encryption applies to the entire file, not to individual columns. Currently, the system uses one active encryption key for all encryption and decryption activities.

4.1 Configuring Administrator Key for Data File Encryption

The platform uses an encryption key to secure files during data processing. This key is set up and managed only by Admin users and is used for both encrypting outgoing files and decrypting incoming files. Encryption options are available only if an Admin has configured a valid key.

1. Navigate to the **DFCS Home Page** > **LHS** > **Administration** > **Configuration** > **Encryption Key Configuration**.
   The **Encryption Key Configuration** screen is displayed.
2. In the **Encryption Key** dialog box, enter a custom key.

Figure 4-1 Encryption Key Configuration



> **Note**
>
> - The key must be in Base64-encoded `KEY:IV` format with a maximum character limit of 256.
> - When a new key is configured, previously encrypted files cannot be decrypted using the old key.

Copyright © 2000, 2026, Oracle and/or its affiliates.

3. Click **Save**. A key is generated to encrypt files in EDD and Data Extraction.

4.2 External Data Descriptor - Encrypted Files

When working with EDD, you can specify whether the incoming data file is encrypted so that the system can process it correctly.

By default, encryption is turned **OFF**, and the system expects files to be in plain text. If your incoming file is encrypted, you must enable the **Is data file(s) encrypted?** option in the **Data** tab. Once enabled, the system will automatically decrypt the file using the configured encryption key and then load the data.

If your data file is provided within an **archive (such as a ZIP file)**, ensure that the individual files are already encrypted before they are archived. The system does not encrypt or decrypt the archive itself.

For **control files**, if they are provided separately from the data file, you can enable the **Is control file encrypted?** option in the **Control** tab. This ensures the control file is decrypted before processing. However, if both data and control information are contained in the same file, this option will be disabled, and the encryption setting defined for the data file will apply to the entire file.

For more information, see External Data Descriptor (EDD).

4.3 Troubleshooting Encryption Error

- **Encryption Errors**: Unable to encrypt the file. Please make sure the Encryption Key entered by Admin is valid.
- **Decryption Errors**: Unable to decrypt the file. Please make sure the Encryption Key entered by Admin is valid.
- **Invalid Key Format**: Invalid encryption key format. Please provide the key in Base64-encoded KEY:IV format.

Copyright © 2000, 2026, Oracle and/or its affiliates.

5
Data Trace

Data Trace provides visibility into data flows within the platform across Entity Data Descriptors, Ingest Connectors, dimension management jobs, and Application Data Services. This view of data flow assists in easier troubleshooting and impact analysis.

This feature improves operational efficiency by introducing transparency into data movement. It reduces troubleshooting time, accelerates root cause analysis, and simplifies impact assessment for changes.

Use Data Trace to view the upstream lineage of a target data object. Data Trace shows the source objects, connectors, and processes that supply data to the target object. You can trace the lineage of an entity, external data descriptor (EDD), application data service (ADS), or slowly changing dimension (SCD) object.

Data Trace identifies each source using one of the following connector categories:

- Application Connector - a connector associated with an application integration.
- Ingest Connector - a connector associated with ingesting data into the platform.

> **Note**
>
> Before you can view trace diagrams, an administrator must run the Data Trace process at least once. Run the process again after adding connectors or entities.

You can trace data backward through all available upstream levels until you reach root-level source objects with no upstream dependencies.

5.1 Viewing Data Trace from Application Connectors

Application Connectors move data into staging entities from application-catalog sources. To view the trace:

1. From the Data Foundation for Banking Home page, in the **Data Services** panel, click **Data Operations**.

   Process Modeller opens.
2. In the Search field, enter **Data Trace**.
3. Select the **Process** tab.
4. Locate the process with Process ID and Process Code DF_DATA_TRACE, and Process Name Data Trace.
5. In the Actions column, click **More Actions** (), and then select **Execute Run**.
6. In the Execution panel, select an execution type.

   Select **Without Parameters** to use the default configuration.
7. If required, enter an object ID.
8. Enter the **Application Params (in JSON format)** parameters.

Copyright © 2000, 2026, Oracle and/or its affiliates.

9. Click **Execute**.
10. To monitor the process, select **Process Flow Monitor** from the **More Actions** menu.
11. On the Data Foundation for Banking Home page, under **Data Services**, select **Application Connectors**.
12. Locate the connector to trace. Use the search field or filter the list by Foundation Catalog, Application Catalog, Created By, or Seeded/Custom.
13. In the connector row, click the **View Data Trace** icon. The icon appears as a grid next to the More Actions (⋮) menu.
14. Selecting View Data Trace opens the Data Trace diagram for the connector's target entity.
15. Click on PMF Execution to start the execution. For more information, see Executing Run Pipeline.

    After the process completes successfully, Data Trace diagrams are available from the Application Connectors and Ingest Connectors pages.

> **Note**
>
> The Data Trace process includes the preconfigured DataTrace activity. Do not modify this activity.

When a source object is supplied by another connector or process, Data Trace includes that upstream relationship in the same diagram.

For example, a diagram can show an SCD process that supplies a Dimension Master, which is supplied by an Application Connector. Data Trace continues to resolve each branch until it reaches a root-level source object.

5.2 Viewing Data Trace from Ingest Connectors

Ingest Connectors load data into the platform from external data source templates. To view an Ingest Connector data trace:

1. From the navigation menu, select Ingest Connectors.
2. In the Search field, enter the connector name. In the connector row, click the View Data Trace icon in the Actions column.

   The Data Trace diagram opens and displays the data source template that supplies the connector and the Ingest Connector target object that it loads.

Copyright © 2000, 2026, Oracle and/or its affiliates.

6

Ingest Connectors

Ingest Connectors serve as configurable components that enable the mapping of one or more **External Data Descriptors (EDDs)** to an **Application Data Interface (ADI)**. They are designed to support both inbound and outbound data exchange, ensuring seamless integration between external systems and the **Financial Services Data Foundation**.

1. Navigate to **Home** page, under **Data Services** > **Ingest Connectors**.

6.1 Icons and Description

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

Table 6-1 Icons and Description

Icon	Description
	Click the **Source** button to view the list of all External Data Descriptors and Catalog Lookup Objects (CLO) 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/ CLO, you can drag the desired EDD/ CLO 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.
	The Filter component is used to 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.

Copyright © 2000, 2026, Oracle and/or its affiliates.

Table 6-1 (Cont.) Icons and Description

Icon	Description
	The Expression component is used to define 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.

6.2 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 Services**, and then click **Add Connector**.

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

Copyright © 2000, 2026, Oracle and/or its affiliates.

The **New Connector** window is displayed.

4. To define a connector, you must have a source with EDD, Catalog Lookup Objects and a target, which is ADI.
5. Click the **Source** button to open the Data Descriptors panel, which will display all available EDD/ Catalog Lookup Objects defined.
   a. You can use **Search** to quickly locate the required CLO (for example, *Accounting Entries*, *Accounting Entries Header*, or *Catalog Lookup – Left Outer Join*).
   b. Drag and drop the selected EDD/ CLO. The Catalog Lookup Object is added to the canvas and is ready to be mapped.
6. Select the default File Extracts file type EDD/ Catalog Lookup and drag it on to the Connector canvas.
   - Ingest Connectors support creating derived expressions using transformed columns from the EDD and derived business terms from Catalog lookup.
   - Users can define derived columns by referencing transformed columns or derived business terms in their expressions.
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 create catalog lookup, 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 enabled and should not be selected.

    > **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.

Copyright © 2000, 2026, Oracle and/or its affiliates.

- 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.

6.3 Using Join (Inner 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)/ Catalog Lookup Objects/ 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.

6.4 Using Filter

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

Copyright © 2000, 2026, Oracle and/or its affiliates.

1. Drag and drop the **Filter** component onto the canvas to define a filter for an entity such as **EDD** / **CLO** (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 the 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')`

6.5 Using Lookup (Left Outer Join)

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, then 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 as many entities on the left and right tabs you want to define.

Copyright © 2000, 2026, Oracle and/or its affiliates.

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.

6.6 Using 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**.

6.7 Using 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/ CLO.
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/ CLO and it can have only one output.
4. If you have multiple EDDs/ CLOs to be aggregated then you must select as many number of aggregation components, connect each to the respective EDD/ CLO, 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/ CLO 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.

Copyright © 2000, 2026, Oracle and/or its affiliates.

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.

6.8 Using 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/ CLO and can have only one output.
4. If you have multiple EDDs/ CLOs selected, and you want to have a Transpose (Rows to Columns) component for more than one EDD/ CLO, then you must select as many number of Transpose (Rows to Columns) components, connect each to its respective EDD/ CLO, 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**.

6.9 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/ CLO.
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/ CLOs selected, and you want to have a Transpose (Columns to Rows) component for more than one EDD/ CLO, then you must select as many number of Transpose (Columns to Rows) components, connect each to its respective EDD/ CLO, 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**.

Copyright © 2000, 2026, Oracle and/or its affiliates.

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**.

6.10 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**.

6.11 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. Click **Mapping** component on the canvas. The EDDs, CLOs, 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.

Copyright © 2000, 2026, Oracle and/or its affiliates.

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**.

6.12 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/ CLO 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.

Copyright © 2000, 2026, Oracle and/or its affiliates.

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**.

6.13 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/ CLO 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**.

6.14 Specifying Alias in Connector

Alias refers to an assumed name or alternative name 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 Services as and when required while defining connectors.

The following scenario explains the mechanism:

Copyright © 2000, 2026, Oracle and/or its affiliates.

1. Drag and drop an EDD more than once, for use in separate joins or looks-up, on the **New Connector** canvas. Data Services 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.

6.15 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.

6.16 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.

6.17 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.

6.18 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.

6.19 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

Copyright © 2000, 2026, Oracle and/or its affiliates.

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.

6.20 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.

Financial Services Data Foundation Cloud Service for Banking 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.

Copyright © 2000, 2026, Oracle and/or its affiliates.

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.

6.20.1 Reading the Error Table

Financial Services Data Foundation Cloud Service for Banking 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 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.

6.21 Executing Connectors

Use the Process Modelling Framework to execute a connector.

6.21.1 Executing Connector Using Process Modelling Framework

Refer to the https://docs.oracle.com/en/industries/financial-services/ofs-analytical-applications/data-foundation-cloud/26b/dfado/index.html for more details.

6.21.2 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.

6.22 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 Oracle® Financial Services Data Foundation Cloud Service for Banking User Guide.

6.23 Protection

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

Copyright © 2000, 2026, Oracle and/or its affiliates.

6.23.1 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 built-in 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 Oracle Data Foundation Cloud Service Data Catalog.

Copyright © 2000, 2026, Oracle and/or its affiliates.

7

Application Data Services (DFCS Only)

Application Data Service (ADS) feature of DFCS enables secure, automated data exchange between DFCS and other downstream applications such as Profitability and Balance Sheet Management Cloud Service (PBSMCS). ADS provides pre-built components for data exchange jobs like pre-built Glossary mapping and connectors. Users can view out-of-the box Glossary mapping and Connectors. The solution also supports extension of existing connectors.

This user guide demonstrates:

- Setting up Application Data Services
- Managing Application Data Service Definitions
- Subscribing Application Data Services
- Published Application Data Services

For information on Glossary mapping and connectors for PBSMCS and ASCS see, DFCS Integration with PBSMCS for ADS User Guide and .

7.1 Setting up Application Data Services

Setting up Application Data Services involves below steps:

- Create A New User in DFCS IDCS Console for ADS Subscription
- Assign User Access for ADS Subscription
- Get the OAuth Client ID and Client Secret for the User
- Application Data Services Version Compatibility

7.1.1 Create A New User in DFCS IDCS Console for ADS Subscription

Steps to create a new user in DFCS IDCS console

1. Enter the IDCS URL in the browser's URL Address Bar. The Oracle Cloud Account Sign In Window appears.
2. Log in to Oracle Identity Cloud Service (IDCS).
3. Select Oracle Cloud Services. For more information, see Accessing the IDCS Console in the Identity Management Guide.

Copyright © 2000, 2026, Oracle and/or its affiliates.

> **Note**
>
> - Ensure the DFCS Cloud Administrator who creates your account has granted you administrative privileges to access the IDCS Console.
> - For more information on the privileges available, see User Groups and Privileges.
> - If the DFCS Cloud Administrator has granted you only Identity Management privileges and no other DFCS privilege, you are automatically redirected to the IDCS Console specific to DFCS after logging in successfully.
> - After a User signs in to the DFCS Cloud Service, the User to User-Group Mapping created in the **IDCS Console** will onboard into the Master and Mapping Tables.
> For more information about how to Unmap a User from a Group in the **IDCS Console**, see the **Create Application Users** section in the Oracle® Financial Services Data Foundation Cloud Service for Banking Getting Started Guide.

7.1.2 Assign User Access for ADS Subscription

Users must be mapped to below user group to access Data Services Subscriptions, ADS Run History, create Scheduler batches and execute them.

Table 7-1 User Group

User Group	Function Code	Role Name	Role Description
ADMIN	DFBADSSCBR,DFBADSEXEC,DFBADSVW	Application Data Service Administrator	This role helps to Subscribe, Update, View and Execute the Application Data Service
BATCH EXECUTOR	DFBADSEXEC, DFBADSVW	Application Data Service Executor	This role helps to Execute the Application Data Service
BATCH VIEWER	DFBADSVW	Application Data Service Viewer	This role helps to View the Application Data Service

After upgrading to 25C or later, unmap DFCS_PBSM_INTEGRATION from users if it remains assigned. At least one user must be assigned to the required groups; otherwise the subscription cannot be used.

Similarly, a **DFCS user** must be mapped to the integration groups listed below.

- IDNTY_AUTH
- IDNTY_ADMN
- DFBDATANALST
- DFBINTANLST
- DFBPRCADMIN

Copyright © 2000, 2026, Oracle and/or its affiliates.

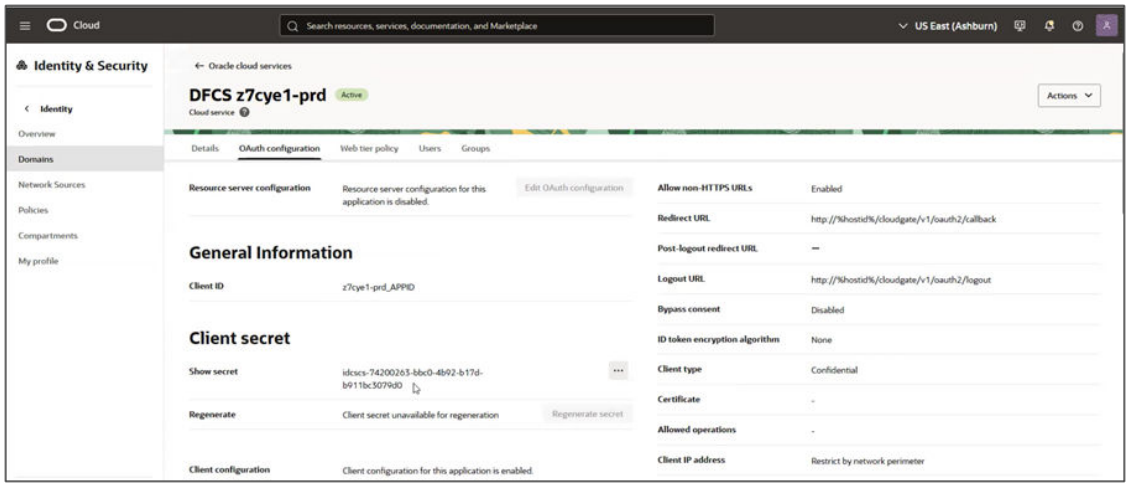
> **Note**
>
> If no user is assigned to the group, the system cannot be used for subscription. For more information on user group see, User Groups and Privileges.

7.1.3 Get the OAuth Client ID and Client Secret for the User

To get the OAuth Client ID and Client Secret, follow these steps:

1. Enter the IDCS URL in the Browser's URL Address Bar. The Oracle Cloud Account Sign In Window appears.
2. Log in to Oracle Identity Cloud Service (IDCS).
3. From the LHS, click **Domain** and select the DFCS provisioned domain.
4. Select **Oracle Cloud Services**. For more information, see Accessing the IDCS Console from Administering Oracle Identity Cloud Service.
5. From the **Oracle Cloud Services** window, search and select the required DFCS Service from the list.
6. Click the **OAuth Configuration** tab. The Client ID and Client Secret Details are displayed in the General Information section.
7. Copy the Client ID and Client Secret.

Figure 7-1 Oracle Cloud Service - Client Secret



7.1.3.1 ADS Credential Validation through the IDCS Integrated Application

To use ADS, you need to set up secure credentials using an Integrated Application in IDCS. This is a one-time setup that replaces the older Oracle Cloud Services (OPC) method.

> **Note**
>
> Tenants with existing credentials can continue using the older automatic Oracle Cloud Services (OPC), based setup until further notice.

Copyright © 2000, 2026, Oracle and/or its affiliates.

Unlike the older Oracle Cloud Services (OPC) setup, the new Integrated Application approach requires the administrator to manually create an application in IDCS. Once created, a Client ID and Client Secret are generated.

- **First-time Subscription:** Enter the Client ID and Client Secret from the Integration Application, along with any other required credentials.
- **Existing Subscription:** Update your subscription by replacing the current credentials with the new Client ID and Client Secret generated from the Integration Application.

Adding a Confidential Application

1. Go to **Oracle Cloud Console** and navigate to **Identity & Security** > **Domains**.
2. On the **Domains** list page, select the domain in which you want to make changes. Select your domain to **Open Details** tab.
3. On the details page, select **Integrated applications**. A list of applications in the domain is displayed.
4. Select **Add application**.
5. In the **Add application** window, select **Confidential Application**.
6. Click **Launch workflow**.
7. In the **Add application** details page, use the following table to configure application details and the display settings.

Option	Description
Name	Enter a name for the confidential application. You can enter up to 125 characters. It is recommended to enter the tenant name. For applications with lengthy names, the application name appears truncated in the **My Apps** page. Consider keeping application names as short as possible.
Description	Enter a description for the confidential application. You can enter up to 250 characters.

8. On the **Client configuration OAuth** pane, select **Configure this application as a client now**.

   Use the following table to provide the information needed to configure this application as a client.

Option	Description
Resource owner	Use when the resource owner has a trust relationship with the confidential application, such as a computer operating system or a highly privileged application, because the confidential application must discard the password after using it to obtain the access token. **Note** *DFCS_PBSM_INTEGRATION* must be mapped to the application.

Copyright © 2000, 2026, Oracle and/or its affiliates.

Option	Description
Client credentials	Use when the authorization scope is limited to the protected resources under the control of the client or to the protected resources registered with the authorization server. The client presents its own credentials to obtain an access token. This access token is either associated with the client's own resources, and not a particular resource owner, or is associated with a resource owner for whom the client is otherwise authorized to act
Refresh token	Select this grant type when you want a refresh token supplied by the authorization server, and then use it to obtain a new access token. Refresh tokens are used when the current access token becomes invalid or expires and don't require the resource owner to reauthenticate.

9. Click **Submit**.
10. Select **Finish.**
11. Record the **Client ID** and **Client secret** in the Application added dialog box.

    To integrate with your confidential application, use this **Client ID** and **Client Secret** as part of your connection settings.
12. Select **Close.**

    The new application's details page is displayed.
13. To Update Subscription UI, open the ADS Subscription UI and enter the following details.
    - Domain URL
    - Client ID
    - Client Secret
    - Username
    - Password

    > **Note**
    >
    > Do not check in the **Integrated App Migration** check box.

14. Click **Update Credentials**.
15. To test connection, click **Test Connection**. The system validates the subscriber, publisher.
16. If successful, the connection is established, credentials are saved.

> **Note**
>
> - OPC to Integrated App Migration is one-time only
> - You cannot revert to old (OPC) method

Regenerating a Client Secret for Confidential Applications

Copyright © 2000, 2026, Oracle and/or its affiliates.

When you create a confidential application, you use a **Client ID** and a **Client Secret** as part of your connection settings. You can regenerate your **Client Secret** at any time for a confidential application using the identity domains console. For more details, see Adding a Confidential Application.

1. On the **Domains** list page, select **Integrated applications** tab. A list of applications in the domain is displayed.
2. Select the confidential application for which you want to regenerate a client secret.
3. Depending on the options you see, do one of the following:
   - Select the **OAuth configuration** tab and then select **Regenerate secret**, or
   - In the **General information** section of the application detail page, select **Regenerate**.
4. Confirm your choice to regenerate the secret. The new Client Secret appears in the **New Client Secret** dialog box.
5. Update the ADS subscription credentials with the new **Client Secret**.

7.2 Managing Application Data Service Definitions

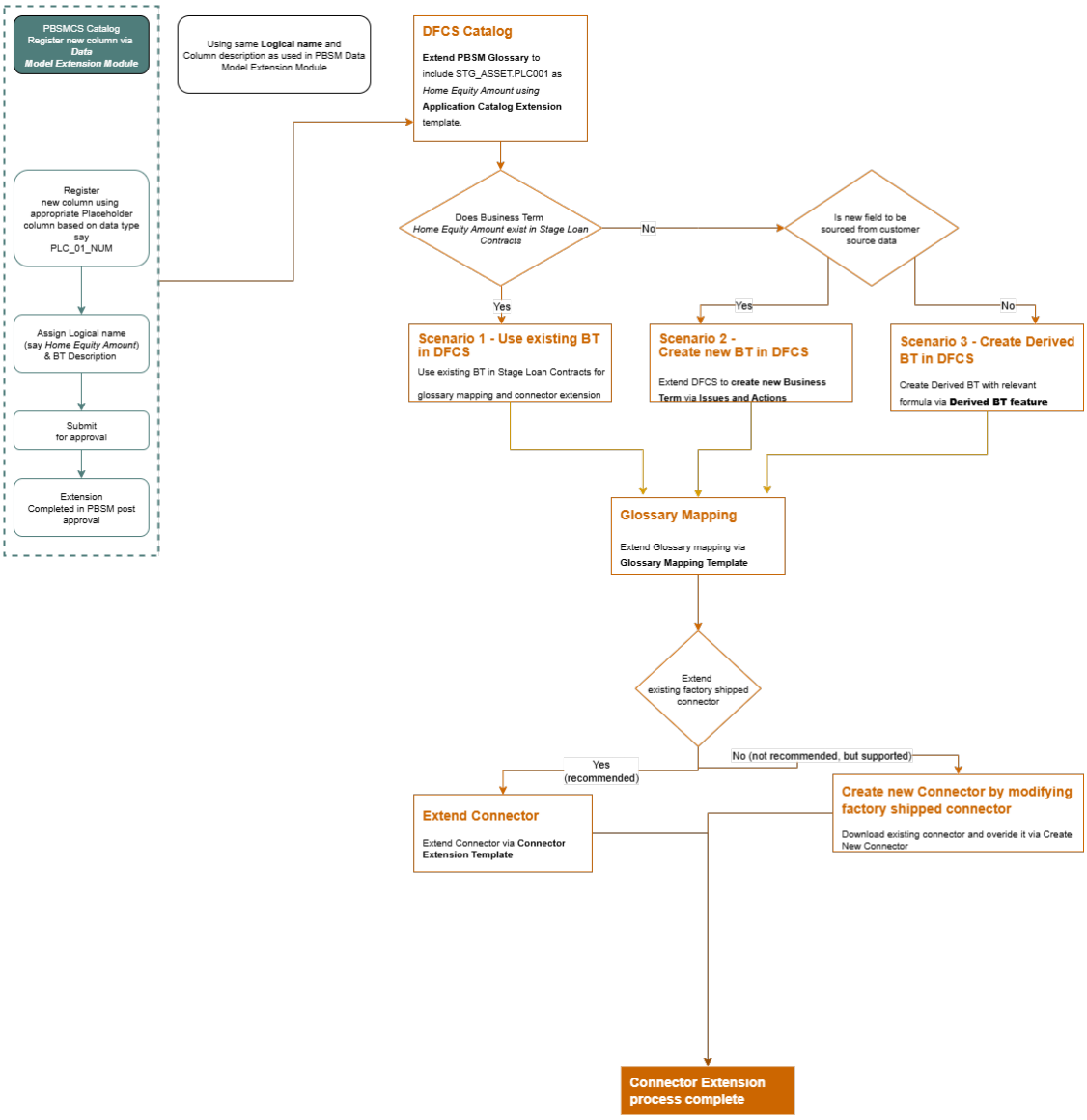
The Catalog section on the DFCS home page provides access to Application Data Services (ADS) related glossary mappings and connectors. Users can browse available solutions to view out-of-the-box glossary mappings and connectors. Users can also extend or override existing connectors as needed.

For example, a bank may need to add an additional attribute to a PBSM entity and include it in the ADS connectors.

The diagram outlines the steps for extending connectors and business terms (BT) in DFCS:

Copyright © 2000, 2026, Oracle and/or its affiliates.

Figure 7-2 DFCS ADS Connectors



This below list outlines the key steps followed to browse and manage ADS definitions.

1. **Register New Column**: Register a new column in the OCI application Catalog with a logical name and description, then submit for approval.
2. **Extend DFCS Catalog**: Add the new column to the OCI application glossary using the Application Catalog Extension template.
3. **Business Term Check**: This step results in one of the following three scenarios:

Scenario	Description	Required Action
Scenario 1	The newly added Business Term in the ADS application already exists in the DFCS entity.	Create a Glossary Mapping and extend the existing connector.

Copyright © 2000, 2026, Oracle and/or its affiliates.

Scenario	Description	Required Action
Scenario 2	The newly added Business Term in the ADS application doesn't exist and has no data points available for derivation (data must be sourced from the customer).	Create a new Business Term in DFCS, then perform Glossary Mapping and extend the existing connector.
Scenario 3	The newly added Business Term in the ADS application doesn't exist but can be derived from existing DFCS data points.	Create a Derived Business Term and override the existing connector by creating a new connector using the current one as the base.

4. **Extend Connector**: Extend or modify the existing connector as needed.
5. **Completion**: Finalize the connector extension process.

Refer to Oracle® Data Foundation Cloud Service Data Catalog for **Managing Application Data Service Definition**.

7.3 Subscribing Application Data Services

This section describes configuring OCI Application (for example, PBSMCS, ASCS) to receive data from DFCS by establishing a subscription, enabling the data import process from the publisher.

- Application Data Services Version Compatibility
- Steps to Subscribe to Application Data Services
- How to Refresh Extensions and Updates from DFCS

7.3.1 Application Data Services Version Compatibility

This section explains how applications interact with DFCS based on glossary version compatibility. It covers:

- First-time subscription rules
- Upgrade of Application Data Services
- Version Matrix Reference

1. First-Time Subscription

An application can subscribe to Application Data Services in DFCS only if the glossary version in DFCS is the same or lower than the application version.

> **Note**
>
> - If the DFCS glossary version is **higher** than the app version, Subscription is **NOT** allowed.
> - **Check for Extension** is initially disabled. Once DFCS subscription is successfully registered, **Check for Extension** becomes enabled.

Example

Copyright © 2000, 2026, Oracle and/or its affiliates.

- DFCS **26D** contains Application Data Services for any OCI application such as **PBSMCS**, with glossary version **26B**.
  - OCI application (PBSM) versions **26B**, **26C**, and **26D** can subscribe to DFCS 26D.
- DFCS **26A** contains glossary version **26A** for PBSMCS.
  - Any OCI application **26A** can subscribe to DFCS **26A**.

2. Upgrade of Application Data Services

If the App version is same or higher than the DFCS glossary version, the App can subscribe to the latest Application Connectors.

However, if DFCS is upgraded and its glossary version for the app is **higher** than the app version, the app will continue using the **previously subscribed** connector version. In this case, the App **cannot access new extensions** until it is upgraded to the latest compatible DFCS glossary version.

> **Note**
>
> The ADS Connector supports multiple overrides for a given connector configuration. There are no restrictions on the number of overrides that can be defined.

Example

- If DFCS upgrades to **26A** and includes a OCI application (PBSMCS) glossary for **25D**, and the OCI application has not yet upgraded to 26A, the OCI application will continue to receive upgrades, because the glossary version (25D) is not higher than the app version.
- If DFCS upgrades to **26A** and includes a OCI application (PBSMCS) glossary for **26A**, and the OCI application remains at **25D**, the OCI application will continue using the **25D** Application Data Services it had before the DFCS upgrade. In this case, the OCI application will not receive new upgrades until it upgrades to **26A**.

3. Version Matrix Reference

The ADS Connector can support multiple overrides within one connector configuration. This allows one published DFCS glossary version to work with multiple PBSM application versions at the same time.

The table below shows the current version compatibility matrix for DFCS publisher version 26.8.1 and the subscriber application (PBSM).

Table 7-2 Version Matrix for Publisher and Subscriber

Publisher ADS Version (DFCS)	Subscriber ADS Version (PBSM)
26.8.1	26.5.1
26.8.1	26.6.1
26.8.1	26.3.1
26.8.1	26.1.1
26.8.1	26.8.1

Each row with **F_LATEST_RECORD_INDICATOR** set to **Y** represents an active override that is currently in effect. In this example, subscriber application **PBSM** versions **26.1.1**, **26.3.1**, **26.5.1**, **26.6.1**, and **26.8.1** are all valid, currently active overrides against publisher **DFCS 26.8.1**.

Copyright © 2000, 2026, Oracle and/or its affiliates.

7.3.2 Viewing Data Services Subscription Dashboard

The Data Services Subscriptions page lists every application data service your environment is subscribed to, organized by category (for example, Customer or Dimensions). Use this dashboard to check which version of a data service is active, how many attributes it delivers, and the underlying code used to identify it in connectors and technical configuration.

To fill in subscription details of Data Foundation Services, select Admin Tools, select Data Foundation Integration, and then select Data Services Subscriptions.

The Data Services Subscriptions dashboard is displayed.

The dashboard displays the following details: Application Data Service:

- **Application Data Service**: The name of the subscribed data service, grouped by category (for example, Customer, Dimensions). Displays the total number of ADS or Connectors present.
- **Version**: The version of the data service currently active in your subscription.
- **Type**: The Type column identifies the direction in which an Application Data Service exchanges data between DFCS and a SaaS application. There are two type:
  - **Inbound Type**: Data is sent from DFCS to a SaaS application. The connector type that is
  - **Outbound Type**: Data is sent from a SaaS application to DFCS.
- **Description**: A short description of what the data service provides.
- **Application Entities**: The source entity or entities the data service is built from (for example, Staging Customer).
- **No. of Attributes Received**: The number of attributes delivered by the data service.
- **ADS Pipeline Name**: Displays the name of the ADS pipeline associated with the data service. The PMF name uses the connector name instead of the connector code.

> **Note**
>
> The PMF name is generated using the connector name, not the connector code.

- **ADS Pipeline Code**: Displays the unique internal code that identifies the ADS pipeline. Use this code in connector configuration. Hover over the info icon next to a data service to view its code.

> **Note**
>
> The Code that is displayed in the info popup (for example, PBSM_ADS_Q5K6NG_PRD_STG_CUSTOMER_MASTER_PARTY_MASTER) is the identifier used when referencing this data service, such as in connector configuration or support requests.

- **Publisher**: The Data Foundation Cloud Service tenant name. This option will be available if you have subscribed for Data Service subscription.
  You can expand each of the Application Data Service Name to view the mapped subsets.
- **Preview Data**: Select Preview Data to inspect a sample of the records delivered by a data service.

Copyright © 2000, 2026, Oracle and/or its affiliates.

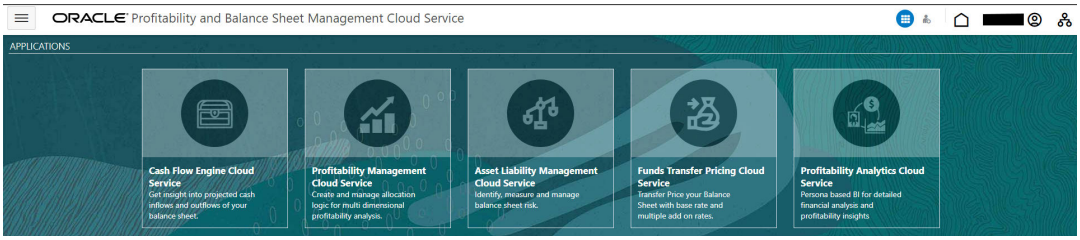
7.3.3 Steps to Subscribe to Application Data Services

This section outlines the steps for first-time users to subscribe and for existing users to upgrade their current subscription.

> **Note**
>
> Before you begin the subscription, ensure the user groups are mapped. For more information, see User Groups and Privileges section in the *Identity Management* guide.

1. Start with subscribing FSGIU SaaS application, say, **Profitability and Balance Sheet Management Cloud Service**.
2. The below steps provide an example of the Profitability and Balance Sheet Management Cloud Service Console. Note that the process remains the same for any FSGIU SaaS application.

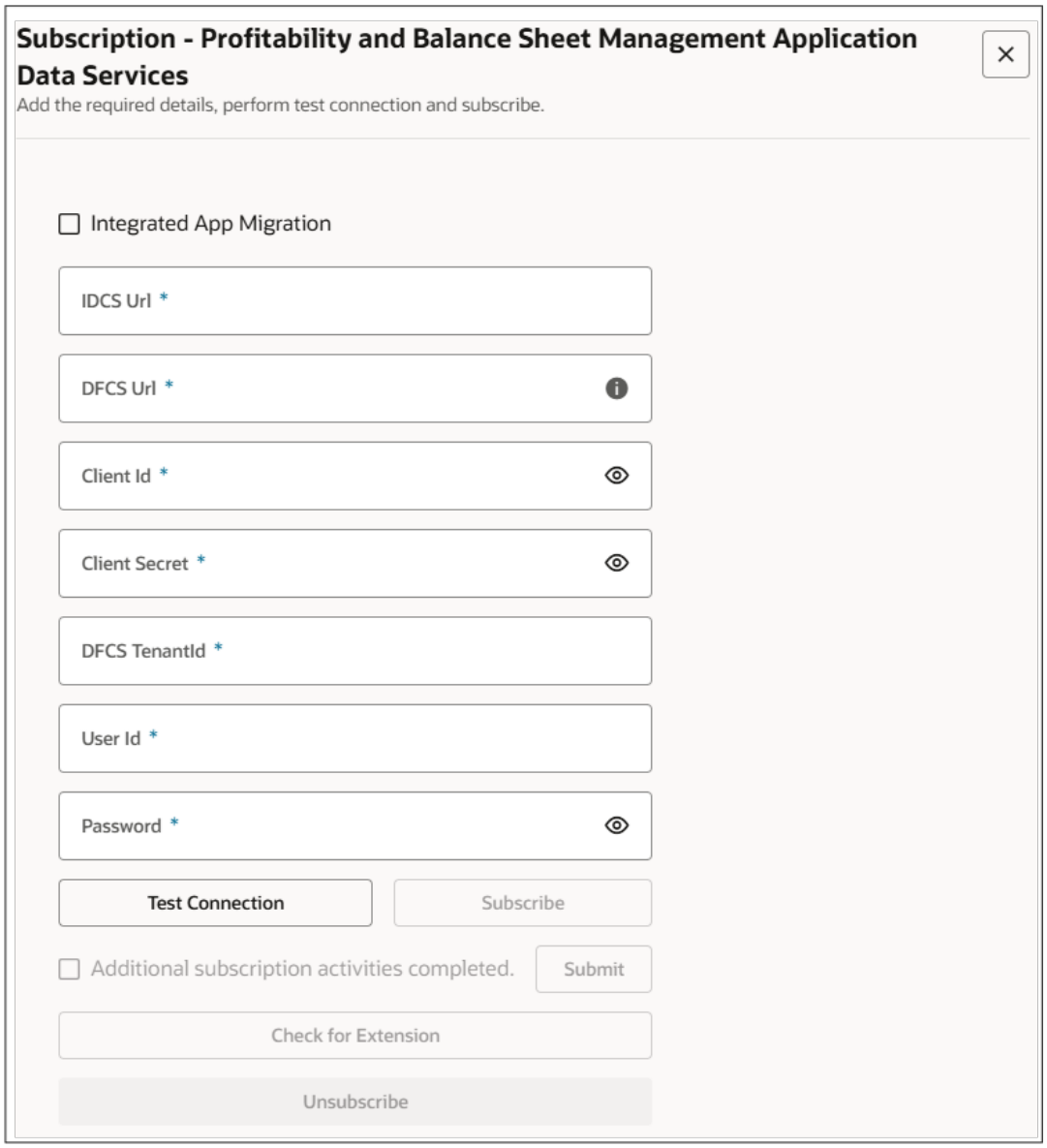
Figure 7-3 Profitability and Balance Sheet Management Cloud Service Console



3. Select a specific service, say, **Funds Transfer Pricing Cloud Service** option and then from the LHS, click **Admin Tools → Data Foundation Integration → Data Services Subscriptions**.
4. Click **Add Subscription** if you are subscribing for the first time. The Subscription pane is displayed.

Copyright © 2000, 2026, Oracle and/or its affiliates.

Figure 7-4 Add Subscription Pane



- **For First-Time Subscription**:
  a. Enter the following.

Copyright © 2000, 2026, Oracle and/or its affiliates.

Table 7-3 Application Data Service

Field	Description
IDCS URL	To obtain the IDCS URL, the user must log in to the DFCS Identity Cloud Service (IDCS) instance that is used for authentication. This enables the OCI application to authenticate securely with DFCS. **Note** For more information on how to obtain the IDCS URL, see Get the OAuth Client ID and Client Secret section in the Application Data Services (DFCS Only) chapter.
Integrated App Migration	Select the **Integrated App Migration** check box when using Oracle Cloud Services (OPC) or the Integrated Application method during subscription. For more information, see ADS Credential Validation through the IDCS Integrated Application section. **Note** If you use the Confidential App, you must enable this setting; otherwise, an error will occur.
DFCS URL	The base URL of the **Data Foundation Cloud Service (DFCS)** instance. It specifies the source environment from which the OCI application will receive data. Hover over the info icon to view the sample URL. **Note** For more information on how to obtain the IDCS URL, see Get the OAuth Client ID and Client Secret section in the Application Data Services (DFCS Only) chapter.

Copyright © 2000, 2026, Oracle and/or its affiliates.

Table 7-3 (Cont.) Application Data Service

Field	Description
Client ID	A unique identifier registered in IDCS for the application (PBSMCS). It is used as part of the OAuth2 authentication process when establishing a connection with DFCS. You can update the Client ID at any time, including after subscription, based on your business requirements. **Note** For more information on how to obtain the IDCS URL, see Get the OAuth Client ID and Client Secret section in the Application Data Services (DFCS Only) chapter.
Client Secret	A confidential key associated with the Client ID. It is used to authorize the OCI application to access DFCS data securely. You can update the Client Secret at any time, including after subscription, based on your business requirements. **Note** For more information on how to obtain the IDCS URL, see Get the OAuth Client ID and Client Secret section in the Application Data Services (DFCS Only) chapter.
DFCS Tenant ID	The unique identifier for your DFCS tenant environment. It ensures that the connection and data transfer are scoped correctly to your organization's DFCS instance. **Note** DFCS and the corresponding OCI application, say PBSMCD or ASCS must use the same tenant type (**-prd** or **-nprd**). Cross-tenant usage is not supported. For example, if DFCS is a production tenant then the subscriber tenant must also be a production tenant.

Copyright © 2000, 2026, Oracle and/or its affiliates.

Table 7-3 (Cont.) Application Data Service

Field	Description
User ID	The DFCS application user's login ID authorized to access and retrieve data from the DFCS environment. This user must have the appropriate roles and permissions for data publishing. **Note** The User ID is case-sensitive and must be created in lowercase letters.
Password	The corresponding password for the DFCS User ID. It is used during authentication to validate the user's access when connecting PBSMCS to DFCS.

Note

For an existing subscription, IDCS URL, DFCS URL, Client ID, Client Secret, and DFCS Tenant ID cannot normally be edited because they were established during the initial subscription.

b. Click **Test Connection** to verify the connection.

Note

This checks if the current credentials are working or if there's a connection issue. The **Update Credentials** appears on the screen.

c. Once the credentials are entered correctly, you can click **Update Credentials** to save your changes.

d. If successful, a message *"Operation successfully connected"* appears at the top of the pane, and the **Subscribe** button is enabled.

e. If the test fails, recheck and verify the entered details.

f. Click **Subscribe**.

g. When the subscription starts, a message *"Please Contact Oracle Support to Complete Additional Activities"* will appear.

Note

Subscription completion may take a few minutes.

h. If you've already subscribed, you can unsubscribe from the service or make changes to your subscription, click **Unsubscribe**. The pop-up message informs

Copyright © 2000, 2026, Oracle and/or its affiliates.

you that unsubscribing from the **DFCS Tenant** is **permanent** and **cannot be undone**.
You have two options:

- **Cancel**: If you change your mind and do not want to unsubscribe, click the **Cancel** button. This will close the prompt and allow you to continue using the service.
- **Yes, Continue**: If you're sure you want to unsubscribe, click the **Yes, Continue** button. This will confirm the action and unsubscribe you from the service permanently.

> **Note**
>
> When the PBSM wallet is re-uploaded with the G2 certificate to avoid ADB batch failures with SSL/TLS certificate errors (ORA-17002: PKIX path building failed), the same must also be updated on the DFCS end. To upload the wallet, raise a Service Request (SR) with Oracle Support.

i. After the subscription is completed, raise a **Service Request (SR)** with the DFCS Tenant details and Subscriber (OCI Application) Tenant details to finalize the additional subscription process.

j. Once the SR is processed and confirmation is received, select the **Additional Subscription Activities Completed** checkbox.

k. Click **Submit** to complete the subscription.

- **For Editing an Existing Subscription**:

a. Update the **User ID** and/or **Password** if required.

b. Click **Update Credentials** to save the changes.

> **Note**
>
> If you are already subscribed, click **Edit Subscription** to update the User ID and Password. Note that fields such as **IDCS URL, DFCS URL, Client ID, Client Secret,** and **DFCS Tenant ID** cannot be edited, as they are mapped/ created by the admin during the initial subscription. For more information, see Accessing the IDCS Console.

You can also click **Check for Extension** if extensions have been already published.

7.3.4 How to Refresh Extensions and Updates from DFCS

After subscribing to Application Data Services (ADS), you may need to apply **extensions** to include additional data elements or custom mappings. The following steps outline how to verify, extend, and execute the required processes.

1. **Upgrade of Application Data Services**: Subscribers must perform an upgrade based on the following scenarios:
   - An existing connector is extended
   - A **new custom connector** is created in DFCS

Copyright © 2000, 2026, Oracle and/or its affiliates.

- There is a **version upgrade** of ADS that is compatible with the current PBSMCS release

2. **Check for Extension (ADS - Subscriber)**: The **Check for Extension** feature allows users to identify and update the newly added connectors.
   - After completing the ADS Extension at the DFCS setup, click **Check for Extension**. A confirmation message appears: **Do you want to proceed with the extension?**
   - Click **Confirm** to proceed.
   - Then click **Submit** to finalize the confirmation.
3. **Refresh Application Data Services**:
   - Click **Extend ADS** to trigger the extension of Application Data Services.
   - After the extension is triggered, **refresh the page once** so the updated extensions are visible in the UI.
   - Refresh does not upgrade PBSMCS to a higher DFCS version.
   - This process updates the existing **PMF** specific to the extensions.

7.4 Published Application Data Services

The **Data Services Publications** screen in DFCS displays all active subscribers receiving data via Application Data Services (ADS). This helps validate that the subscription from the selected **OCI Application** to DFCS has been successfully established.

1. Access the Data Foundation for Banking portal.
2. Click **Administrator → Data Services Publication**.
3. Verify that the OCI Application Data Services appears as a subscriber.
4. Confirm that your PBSMCS or ASCS instance (e.g., PBSMADS E26SJU-PRD) is listed.
5. Review the number of:
   - Application Data Services
   - Entities Data Shared
   - Attributes Data Shared
6. Click the arrow next to the subscriber name to view more details about the shared ADS, version, entities, and attributes.

> **Note**
>
> If a newer version (e.g., 25.9.1) is available while the system is still mapped to an older version (e.g., 25.6.1), the Version tab will display only the latest version (25.9.1) to prevent duplicate entries in the glossary mappings.

7.4.1 ADS Run History

The ADS Run History provides information related to Batches that have been processed to move data from DFCS to any OCI application.

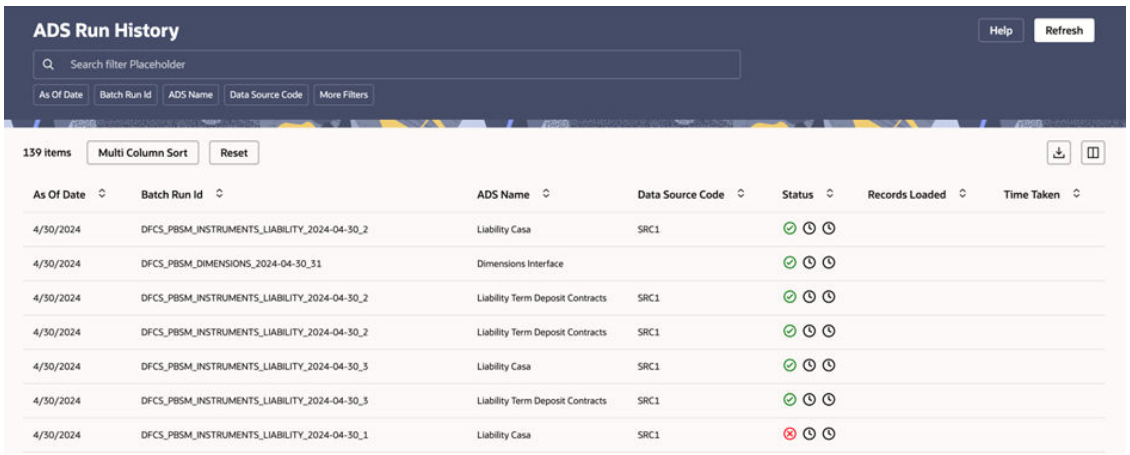
Copyright © 2000, 2026, Oracle and/or its affiliates.

> **Note**
>
> For the user to be able to see the ADS History, the user must be mapped to a group that has the required role (RLADSHISTVIEW). Role is not mapped in the OOTB set up.

To open the ADS Run History, from the LHS menu, , select **Admin Tools**, select **Data Foundation Integration**, and then select **ADS Run History**.

The ADS Run History page is displayed.

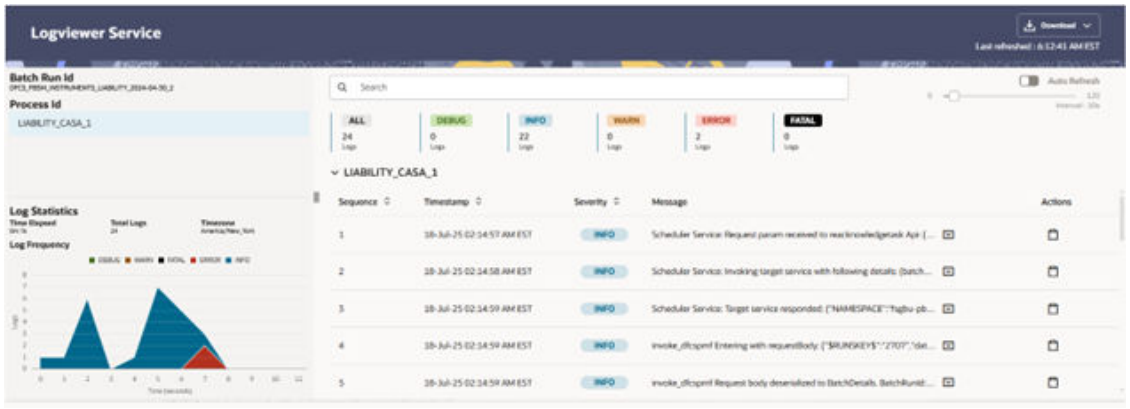
Figure 7-5 ADS Run History



This UI allows you to apply various filters namely, As Of Date, Batch Run Id, ADS Name, and Data Source Code. The default view of ADS Run History displays the mentioned details. You can click **More Filters** which are Data Load Status, Batch Name, Task Name, and Task Id to change the view from default.

You can click the icons under Status to see the status of the batch. The status displays Passed, Failed, and Pending status icons.

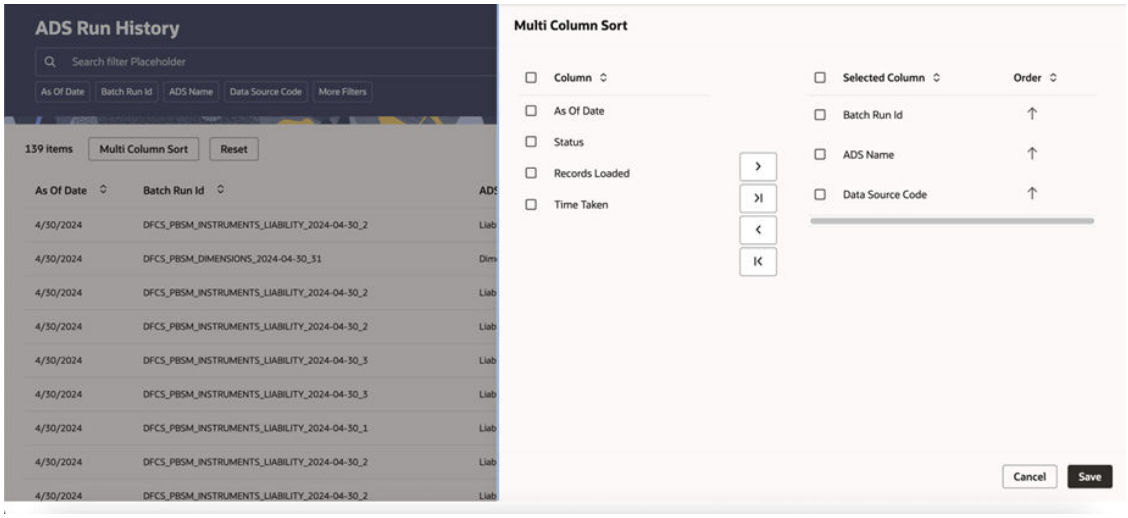
Figure 7-6 Log Viewer Service



Copyright © 2000, 2026, Oracle and/or its affiliates.

The Multi Column Sort allows you to define different columns to be displayed in the UI.

Figure 7-7 Multi Column Sort



Select the required Columns and move them to the **Selected Columns** box and then click **Save**. You can also define the order of the column appearance by selecting ascending or descending setting the order. The ADS Run History UI displays the rows as per selection.

If you click **Reset**, all the filters, columns order will be reset and the ADS Run History UI displays the default view.

You can download the ADS Run History in Excel (.xlsx) format using the **Download** icon.

Click the **Columns** icon to select or deselect the columns you want to display in the ADS Run History data.

Copyright © 2000, 2026, Oracle and/or its affiliates.

8

Integration with Oracle Applications

This section explains the pre-built integration with other Oracle Applications.

8.1 Integration with Oracle Enterprise Data Management Cloud Service

Enterprise Data Management Cloud Service (EDMCS) is an Oracle Cloud application designed to centrally manage and govern enterprise data across business applications. It provides a unified platform for creating, maintaining, and sharing master data such as dimensions, hierarchies, properties, and mappings.

Key features of EDMCS include:

- Centralized data governance with audit tracking
- Support for multiple hierarchies and viewpoints
- Role-based access and workflow approvals
- Integration with Oracle and non-Oracle systems
- Simplified modelling of enterprise structures with drag-and-drop interface

This section outlines the steps required to source dimension-related data from EDMCS into DFCS, a key capability for managing dimensions within DFCS.

You can significantly save the efforts required to manage dimensions in DFCS if those dimensions are already available and maintained in EDMCS within another application, such as *Oracle Cloud ERP*. A dimension from an existing EDMCS application—like *Oracle Cloud ERP*—can be subscribed into the EDMCS application for DFCS. This eliminates the need to manage these existing dimensions independently for DFCS.

> **Note**
>
> When setting up a DFCS application in EDMCS for the first time, you must manually import the existing data for the subscribed dimensions from the source EDMCS application. After the initial import, the subscription will automatically capture new changes and updates (including edits).

For detailed instructions on setting up such subscriptions, refer to the EDMCS User Guide

8.1.1 Pre-Configured Standard Dimensions

The following dimensions are delivered by default in EDMCS and are ready for use:

Copyright © 2000, 2026, Oracle and/or its affiliates.

Table 8-1 Pre-Configured Standard Dimensions in Use

Dimension	Purpose
Product	Defines products or services offered by the organization.
Project	Captures details of specific projects or initiatives.
Data Source	Identifies the origin of data within the system.
Business Segment	Represents major business divisions or market segments.
Line of Business	Groups related products/services under a business line.
Transaction Type	Classifies financial or operational transactions by type.
Organization Unit	Represents organizational entities such as departments or regions.
Product Type	Categorizes products into high-level types or families.
General Ledger	Stores chart of accounts and financial account structures.
Legal Entity	Maintains information about registered legal entities.
Branch	Lists and manages details of organizational branches.

> **Note**
>
> These dimensions can be directly used or customized by adding, renaming, or restructuring nodes to suit specific business requirements. To extend a dimension, update it first in **DFCS**, then in **EDMC**, and make sure the attribute definitions are consistent across both systems.

8.1.2 Validations in EDMCS

EDMCS automatically enforces two critical validation rules when working with **alternative hierarchies** inside a dimension.

1. Leaf Node Consistency Across Alternative Hierarchies (Bottom Node Hierarchy Check)

- **Purpose**: All leaf nodes (lowest-level members without children) must be identical across all alternative hierarchies within the same dimension.
- **When it applies**: Only when a dimension has more than one alternative hierarchy.
- **How it works**: If **Product A** exists as a leaf node in one hierarchy, it must also appear as a leaf node in all other hierarchies for that dimension.
  Example: If *Product A* is a leaf node in one hierarchy, it must also exist as a leaf node in every other alternative hierarchy for that dimension.

No Duplicate Parent–Child Relationships Across Alternative Hierarchies (Intermediate Node Cannot be Shared)

Copyright © 2000, 2026, Oracle and/or its affiliates.

- **Purpose**: The **same parent–child pair** cannot appear in more than one alternative hierarchy for the same dimension.
- **Allowed**: A node can exist in multiple hierarchies, but it **must have a different parent** in each hierarchy.
  Example: If *X* is the parent of *Y* in one hierarchy, this **exact X–Y relationship** cannot be repeated in another alternative hierarchy for the same dimension.

> **Note**
>
> These validations help ensure that alternative hierarchies are **structurally sound**, **consistent**, and **logically distinct** while maintaining alignment at the leaf-node level.

8.1.3 Managing Dimensions and Hierarchies in EDMCS

This section describes how to extend the integration between **Enterprise Data Management Cloud Service (EDMCS)** and **DFCS**, including how to add and manage dimensions, attributes, and hierarchies in EDMCS for use in DFCS.

The service includes a set of preconfigured dimensions in EDMCS. However, depending on your business requirements, you need to configure additional metadata, such as creating a new dimension or adding attributes to an existing dimension.

Review the following guidelines before creating or modifying dimensions and attributes.

- **Dimension Naming**: You can choose the name of the dimension. However, it is recommended to use the same name as the corresponding dimension in **DFCS**.
- **Attribute Naming**: Attribute names must match the names used in **DFCS**. Refer to the **Catalog** for the list of Business Terms (attributes in EDMCS). You can also find the attribute names in the **Download Specification** document.
- **Attribute Data Type**: Attribute data type should be compatible with DFCS Catalog.
- **Date Format**: The date entered must be provided in the date format mm/dd/yyyy of the date attribute in the EDMCS .
- **Hierarchy Rules for Dimensions**: Dimensions with hierarchies must follow the validations listed below:

Validation Rule	Description
Leaf Node Sharing	All leaf nodes must be shared across all hierarchies as leaf nodes.
Intermediate Node Uniqueness	Intermediate nodes must be unique within a dimension.
No Sharing Within Hierarchy	A node can appear only once within the same hierarchy.

The following values are generated automatically using EDMCS expressions.

1. Child Node ID (Child Numeric Identifier): Returns the numeric identifier of the current node using the **DFCS Node ID** property.

```
{
{"statements":
```


Copyright © 2000, 2026, Oracle and/or its affiliates.

```
[{"type":"return","expression":{"type":"node","id":"node","invoke":
{"type":"properties","invoke":
{"type":"propertyValue","id":"ca27fb33-1361-493c-95f6-aabd1218b945"}}}}]}
}
```

2. Parent Node Id (Parent Numeric Identifier)

```
{
{"statements":
[{"type":"if","condition":{"type":"node","id":"node","invoke":
{"type":"properties","invoke":
{"type":"propertyValue","id":"f803899c-8117-41c4-99dc-0f413849656d","invoke":
{"type":"equals","argument":
{"type":"intLiteral","value":1}}}}},"statements":
[{"type":"return","expression":
{"type":"intLiteral","value":0,"invoke":
{"type":"toNumericString","padLength":
{"type":"intLiteral","value":0}}}}],"else":
{"type":"else","statements":[{"type":"return","expression":
{"type":"node","id":"node","invoke":{"type":"parent","invoke":
{"type":"properties","invoke":
{"type":"propertyValue","id":"ca27fb33-1361-493c-95f6-aabd1218b945"}}}}}]}}]}
}
```

3. Node Id (Numeric Identifier)

```
{
{"statements":
[{"type":"return","expression":
{"type":"node","id":"node","invoke":
{"type":"properties","invoke":
{"type":"propertyValue","id":"f85d696d-784a-420f-b9e2-b730a8355b1a"}}}}]}
}
```

4. Hierarchy Code

```
{
{"statements":
[{"type":"if","condition":{"type":"node","id":"node","invoke":
{"type":"properties","invoke":
{"type":"propertyValue","id":"f803899c-8117-41c4-99dc-0f413849656d","invoke":
{"type":"equals","argument":{"type":"intLiteral","value":1}}}}},"statements":
[{"type":"return","expression":
{"type":"node","id":"node","invoke":{"type":"name"}}}],"else":
{"type":"else","statements":
[{"type":"return","expression":
{"type":"node","id":"node","invoke":
{"type":"ancestors","invoke":
{"type":"get","index":{"type":"intLiteral","value":0},"invoke":
```


Copyright © 2000, 2026, Oracle and/or its affiliates.

```
{"type":"name"}}}}}]}}]}
}
```

5. Display Order Number

```
{
{"statements":[{"type":"if","condition":
{"type":"node","id":"node","invoke":
{"type":"properties","invoke":
{"type":"propertyValue","id":"f803899c-8117-41c4-99dc-0f413849656d","invoke":
{"type":"greaterThan","argument":
{"type":"intLiteral","value":1}}}}},"statements":
[{"type":"return","expression":{"type":"node","id":"node","invoke":
{"type":"parent","invoke":{"type":"properties","invoke":
{"type":"propertyValue","id":"b1819319-1484-4dd0-87df-c3d29a05f30d","invoke":
{"type":"indexOf","search":{"type":"node","id":"node","invoke":
{"type":"name"}},"invoke":{"type":"add","argument":
{"type":"intLiteral","value":1}}}}}}}}],"else":
{"type":"else","statements":[{"type":"return","expression":
{"type":"intLiteral","value":0}}]}}]}
}
```

For more information on adding Derived Properties, see Derived Properties.

Validations for Dimensions with Hierarchies

The following are the validations for dimensions with Hierarchies:

- Bottom Node Hierarchy Check- Ensures that the bottom nodes are under each hierarchy (Top Node).

```
{
{"statements":[{"type":"if","condition":
{"type":"node","id":"node","invoke":
{"type":"properties","invoke":
{"type":"propertyValue","id":"e75ab694-50b7-40b8-86d0-50f07d9ae8a3","invoke
":
{"type":"not"}}}},"statements":
[{"type":"comment","value":"If this is not a bottom node then skip the
validation"},{"type":"return","expression":
{"type":"booleanLiteral","value":true}}],"elseifs":
[{"type":"elseif","condition":{"type":"node","id":"node","invoke":
{"type":"allAncestors","invoke":{"type":"filter","predicate":
{"type":"nodePredicate","statements":
[{"type":"node","id":"thisNode","invoke":
{"type":"properties","invoke":
{"type":"propertyValue","id":"f803899c-8117-41c4-99dc-0f413849656d","invoke
":
{"type":"equals","argument":
{"type":"intLiteral","value":1}}}}]},"invoke":
{"type":"size","invoke":{"type":"equals","argument":
{"type":"node","id":"node","invoke":
{"type":"ancestors","invoke":{"type":"get","index":
{"type":"intLiteral","value":0},"invoke":
```


Copyright © 2000, 2026, Oracle and/or its affiliates.

```
{"type":"siblings","invoke":{"type":"size","invoke":
{"type":"add","argument":
{"type":"intLiteral","value":1}}}}}}}}}}}},"statements":
[{"type":"comment","value":"Finally check if it is under all top nodes"},
{"type":"return","expression":
{"type":"booleanLiteral","value":true}}]}],"else":
{"type":"else","statements":[{"type":"return","expression":
{"type":"booleanLiteral","value":false}}]}}]}
}
```

- Bottom Node Hierarchy Check (Remove)- Prevents removing the last location from a hierarchy(Top Node).

```
{
{"statements":[{"type":"if","condition":
{"type":"node","id":"node","invoke":
{"type":"properties","invoke":
{"type":"propertyValue","id":"e75ab694-50b7-40b8-86d0-50f07d9ae8a3","invoke
":
{"type":"not"}}}},"statements":
[{"type":"comment","value":"If this is not a bottom node then skip the
validation"},{"type":"return","expression":
{"type":"booleanLiteral","value":true}}],"elseifs":
[{"type":"elseif","condition":{"type":"node","id":"node","invoke":
{"type":"actions","invoke":{"type":"any","predicate":
{"type":"actionPredicate","statements":
[{"type":"action","id":"action","invoke":
{"type":"type","invoke":{"type":"isRemove"}}}]}}}},"statements":
[{"type":"comment","value":"If it is a Request Remove Action then prevent
it if this is the last location under the top node. \nNote: this requires
visualized mode AND assumes no implicitly shared nodes"},
{"type":"return","expression":{"type":"node","id":"node","invoke":
{"type":"locations","excludeImplicitShares":
{"type":"booleanLiteral","value":false},"sort":
{"type":"booleanLiteral","value":false},"invoke":
{"type":"filter","predicate":{"type":"nodePredicate","statements":
[{"type":"node","id":"thisNode","invoke":{"type":"properties","invoke":
{"type":"propertyValue","id":"b3ef0bbe-8f21-4ba5-ad81-
d5663f6fb6e0","invoke":
{"type":"equals","argument":{"type":"node","id":"node","invoke":
{"type":"properties","invoke":
{"type":"propertyValue","id":"b3ef0bbe-8f21-4ba5-ad81-
d5663f6fb6e0"}}}}}}}]},
"invoke":{"type":"size","invoke":{"type":"greaterThan","argument":
{"type":"intLiteral","value":1}}}}}}}]}],"else":
{"type":"else","statements":
[{"type":"return","expression":{"type":"booleanLiteral","value":true}}]}}]}
}
```

- Intermediate Nodes Cannot be Shared- Ensures that only bottom nodes can be shared, and intermediate roll up nodes cannot be shared between hierarchies.

```
{
{"statements":[{"type":"if","condition":
```


Copyright © 2000, 2026, Oracle and/or its affiliates.

```
{"type":"node","id":"node","invoke":
{"type":"properties","invoke":
{"type":"propertyValue","id":"e75ab694-50b7-40b8-86d0-50f07d9ae8a3","invoke
":
{"type":"equals","argument":
{"type":"booleanLiteral","value":false},"invoke":
{"type":"and","argument":{"type":"node","id":"node","invoke":
{"type":"locations","excludeImplicitShares":
{"type":"booleanLiteral","value":false},"sort":
{"type":"booleanLiteral","value":false},"invoke":
{"type":"size","invoke":{"type":"greaterThan","argument":
{"type":"intLiteral","value":1}}}}}}}}},"statements":
[{"type":"return","expression":
{"type":"booleanLiteral","value":false}}],"else":
{"type":"else","statements":[{"type":"return","expression":
{"type":"booleanLiteral","value":true}}]}}]}
}
```

- No Sharing Within Hierarchy- Within a hierarchy (Top Node) a node can only be present in one location.

```
{
{"statements":[{"type":"return","expression":
{"type":"node","id":"node","invoke":
{"type":"locations","excludeImplicitShares":
{"type":"booleanLiteral","value":false},"sort":
{"type":"booleanLiteral","value":false},"invoke":
{"type":"count","predicate":{"type":"nodePredicate","statements":
[{"type":"node","id":"thisNode","invoke":{"type":"properties","invoke":
{"type":"propertyValue","id":"b3ef0bbe-8f21-4ba5-ad81-
d5663f6fb6e0","invoke":
{"type":"equals","argument":{"type":"node","id":"node","invoke":
{"type":"properties","invoke":
{"type":"propertyValue","id":"b3ef0bbe-8f21-4ba5-ad81-
d5663f6fb6e0"}}}}}}}]},
"invoke":{"type":"lessThanOrEqual","argument":
{"type":"intLiteral","value":1}}}}}]}
}
```

For more information on custom validations, see EDMCS User Guide.

- **Adding New Dimensions**: For more information on adding dimensions, see EDMCS User Guide.
- **Adding a Property to an Existing Dimension**: For more information on adding a property, see EDMCS User Guide.
- **Adding an Existing Property to a Dimension**: For more information on adding an existing property, see EDMCS User Guide.

8.1.4 Steps for Integrating EDMCS with DFCS

This interface allows users to provide necessary credentials and connection details to integrate EDMCS (Enterprise Data Management Cloud Service) with the Data Foundation Cloud Service. It allows DFCS to fetch and utilize dimension metadata for applications.

Copyright © 2000, 2026, Oracle and/or its affiliates.

> **Note**
>
> As a prerequisite, the user is required to raise a Service Request (SR) in My Oracle Support for whitelisting the EDMCS URL. The expected completion time for this request is **3 to 5 business working days**.

1. Navigate to the **DFCS Home Page** > **LHS** > **Administration** > **Data Services** > **Integration with EDMCS**.
   The **EDMCS for Data Foundation Cloud Service** page appears.
2. Enter the credentials to access the EDMCS.

Table 8-2 Share the credentials to access the EDMCS

Field	Description
URL to access EDMCS	Enter the URL for your EDMCS instance in this field. This field is required. (e.g., https://<prod>.oraclecloud.com). Enter 500 or fewer characters.
Version	Enter the version of your EDMCS instance in this field as **v1** only.

3. Click **Save and Get Application**.
4. Once saved, the **Mark As Done** option will be enabled.
5. Select **Authenticate Using?** dropdown. Select the preferred authentication method.
   - Two options are available:
     - **Password**
     - **Access Token**
6. **Option 1: Password Authentication**
   Once **Password** is selected:
   - Provide the **Username** and **Password** for EDMCS.

Table 8-3 Password Authentication

Field	Description
Username	EDMCS service account username. Enter 100 or fewer characters.
Password	Corresponding user password. Enter 255 or fewer characters.

> **Note**
>
> Ensure the user has the necessary EDMCS API access privileges.

7. Option 2: Access Token Authentication
   If **Access Token** is selected:
   - Provide the full access token in the field:

     Specify the token

Copyright © 2000, 2026, Oracle and/or its affiliates.

> **Note**
>
> Token must be valid and scoped for EDMCS APIs. Ensure the token hasn't expired.

8. Click **Save and Get Application** to:
   - Validate the connection
   - Fetch available EDMCS applications and their dimension structures

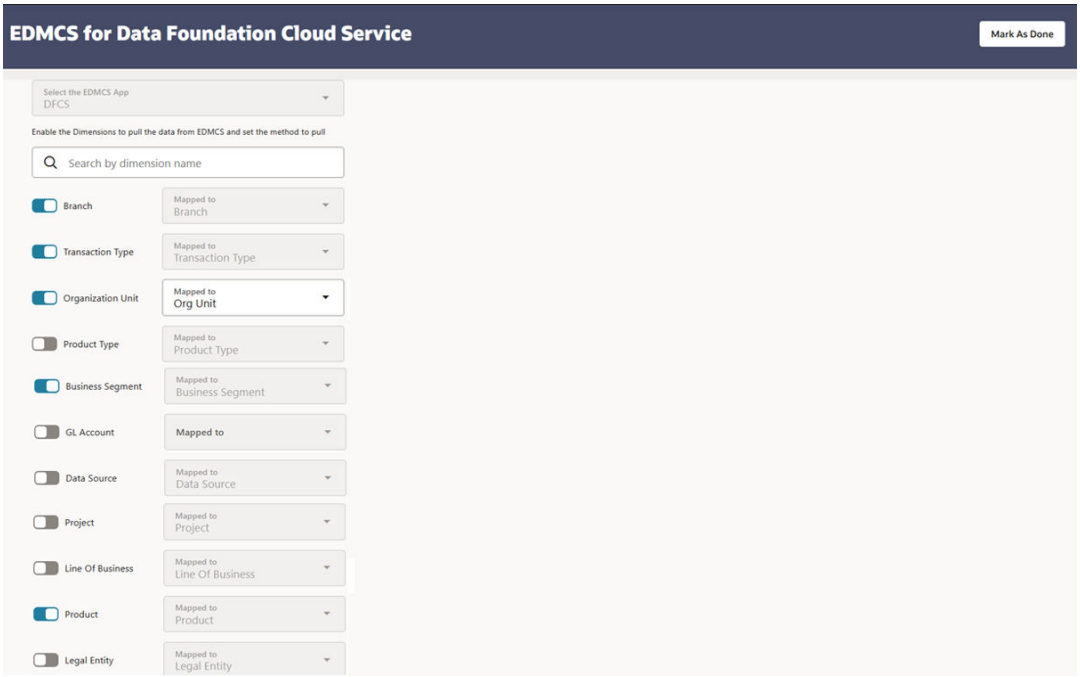
> **Note**
>
> If connection or credentials are invalid, appropriate error prompts will guide the user.

9. Select the required **Dimension**.

> **Note**
>
> Single or multiple Dimensions can be selected.

10. Click **Mark As Done** located at the top right to complete the integration setup.

Figure 8-1 Dimension Mapping



11. After saving successfully, the **PMF** will be created.

Checking PMF Executions

1. Navigate to the **Home** page → **Data Integration** → **Data Operations**.

Copyright © 2000, 2026, Oracle and/or its affiliates.

2. On the Search bar, locate and click on **EDM_DIM_EXPORT_LOAD**.
   All selected dimensions will appear.
3. Click **Execute**.

Figure 8-2 EDMCS Pipeline



4. Select the **Execution Type** and enter the **Object ID**.
   a. If Execution Type is selected as **Without Parameters** then enter the **Object ID** EDMCS fetches the current System date for both the As of Date and the Effective Date parameters.
   b. If Execution Type is selected as **With Parameters**, then enter the **As of Date** and **Effective Date** of the dimensions. EDMCS fetches the dates given in run time PMF parameters response.
5. Click **Execute** again to run the process.
6. Go to the **Process Monitor** page.
7. Check the status for EDM:
   - **Completed** → Process ran successfully.
   - **Failed** → Double-click the failed record to view **Execution Logs** for troubleshooting.
8. Click **Execution Logs** to view the logs.

Validate Data Sourced from EDMCS

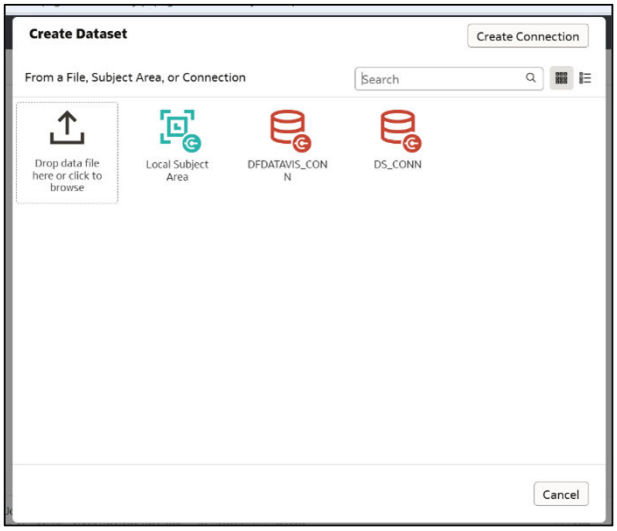
1. From the left menu click **Data Visualization** → **Source Data Visualisation**.
2. Click the **Search** box near the top.
3. From the Source Data Visualisation window, select **Others** or type the dataset name or keyword (e.g. branch) in the search bar and press **Enter**. The dataset page opens showing the data grid.
4. On the dataset page, you can filter the results datasets (e.g., **As Of Date**, **Data Source Code**).
5. Click the three-dots to perform the following actions:
   - **Sort By** — sort the dataset by a chosen column.
   - **Drill to Attribute/Hierarchy** — navigate to related attributes or hierarchies for the selected dimension value.
   - **Edit** — edit dataset metadata or definitions.

Copyright © 2000, 2026, Oracle and/or its affiliates.

- **Export** — export the data.

6. Click **Go back** to return to Oracle Analytics window and choose another dimension.
7. **Create New Dataset**: In the next screen, click on **Create**, then select **Dataset**.
8. **Choose Connection**: In the dataset creation screen, choose the connection **DFDATAVIS_CONN**.

Figure 8-3 Connections



9. **Expand Schema and Select Table**: Expand the **Schemas** section, then expand **DFREP**, and select the **STAGE_PRODUCT_MASTER** table.
10. **View Sourced Data**: The sourced data will now be visible for your selected dimension.

> **Note**
>
> Always ensure that configuration details are accurate before executing any process to avoid failures.

> **Note**
>
> For detailed instructions on Accelerator Applications, refer to Working with Accelerator Applications in the Oracle Cloud Enterprise Data Management guide.

Copyright © 2000, 2026, Oracle and/or its affiliates.

9

Data Ingestion Errors Browser

DFCS is introducing an interface for monitoring data ingestion jobs. It provides a summary, like, number of records processed, number of records successfully loaded, and number of records with error. It also provides a browser for errors identified in data ingestion jobs – both errors in reading the source files as well as errors in populating respective target entities.

Data Flow

The ingestion process follows this flow: **Source File → EDD → Connector → Target Table**

Stages

- **EDD**:
  - Reads data from the source file
  - Performs initial validation and transformation
- **Application Connector**:
  - Loads processed data into the target database table
  - Applies database constraints

How Data Ingestion Works

Data moves in three stages:

Stage	What Happens	Where Errors Can Occur
1. Source File	User uploads a CSV file to the system.	No errors here
2. EDD	The system reads the file and moves data into the application connector.	Data type mismatch, column size mismatch, and similar issues
3. Application Connector to Target Table	The system moves data from the connector into the final database table.	Database violations such as duplicate primary key errors

If a source file has **100 records**:

- **80 records** pass through EDD successfully
- **20 records** fail at the EDD stage

Out of the 80 records that go to the connector:

- **60 records** are loaded into the final table
- **20 records** fail at the connector stage

The failed records are stored separately:

- 20 EDD failures → stored in EDD error table
- 20 connector failures → stored in connector error table

3.2 Application Connector-Level Errors

Copyright © 2000, 2026, Oracle and/or its affiliates.

Errors occurring while inserting data into the database.

Examples:

- Primary key violation
- Constraint violations
- Duplicate records

These records:

- Fail at connector stage
- Stored in **Application Connector error table**

Important Execution Scenarios

Pipeline Stopped and Restarted

If a pipeline execution is stopped before it finishes and then resumed, the system creates a second instance of the same execution. Both instances share the same pipeline name, as-of date, and Run SKey, but they have different internal instance IDs.

In this case, the page automatically shows the errors from the latest (most recent) instance only. You do not need to do anything extra.

> **Note**
>
> *This means the page may not show errors from the earlier stopped instance. If you need to review both, contact your system administrator.*

Threshold Value Set to Zero

Each application connector has a threshold setting. This setting controls how many record failures are allowed before the pipeline stops.

Threshold Value	What Happens
0 (zero)	The pipeline stops as soon as the very first record fails. No error records are saved to the database.
5% (example)	The pipeline allows a small number of failures before stopping. Those failed records are saved and visible on this page.

If the threshold is set to zero, the pipeline will stop as soon as the first error occurs. In this case, no records are saved in the error table. So, even if the pipeline status shows as Failed, you will not see any records on the error page.

> **Note**
>
> *The threshold value is set by the user when configuring the connector. If you are seeing unexpected behavior, check the threshold value for that connector.*

Copyright © 2000, 2026, Oracle and/or its affiliates.

9.1 Accessing Data Ingestion Errors Dashboard via DFCS

Data ingestion pipelines receive files submitted by users and load the data into target database tables. The ingestion process runs through multiple stages. At each stage, the data is validated and filtered before it moves to the next stage.

1. Log in to the DFCS application.
2. Navigate to **Data Ingestion Errors** under **Data Services**.
   The Data Errors window appears.
3. Select the following.

Action	Description
Select a Pipeline	The dropdown shows all pipelines available in the tenant. Select the pipeline you want to check. **Note** *Only pipelines that have been executed will appear in this list.*
Select the As-of Date	The As-of Date dropdown shows the date of the source file used in the execution. Select the date you want to review.
Select the Run SKey	A pipeline can be executed multiple times. Each execution gets a unique number called a Run SKey. Use this dropdown to select the specific execution you want to check. **Example**: If you ran the same pipeline three times on the same day:

Run SK	Execution Time	Use This To
96	Morning run	Check errors from the morning run
97	Afternoon run	Check errors from the afternoon run
98	Night run	Check errors from the night run

4. Click **Submit**.
5. The Summary Page loads automatically, the page shows a table of application connectors and EDD for that execution.
6. Use search bar to find connectors (case-insensitive).
   You will see:

What You See	What It Means
Red error icon next to EDD	Records failed at the EDD stage.

Copyright © 2000, 2026, Oracle and/or its affiliates.

What You See	What It Means
Red error icon next to Connector	Records failed at the connector stage.
No icon (icon is greyed out)	No errors at this stage

7. Click the red icon to view the failed records for that stage. The table shows each failed record and the reason it failed.

Column	Description
Connector Name	Name of the connector
Run SKey	Execution ID
Status	SUCCESS / FAILED
No of EDD	Number of EDDs
No of Error EDD	Number of EDDs with errors
Total Number of Records	Total records processed
Number of Records Loaded	Successfully loaded records
No of Errors	Failed records

8. You can download the error records to a CSV file by clicking the **Export.**

Export Detail	Information
File format	CSV file
File name	**Error Type_EDD/ Connector Name_Timestamp** format
Maximum rows	1,000 records per export

> **Note**
>
> *If you have more than 1,000 error records, only the first 1,000 will be exported. Contact your administrator for more details.*

Searching Within a Pipeline

If a pipeline has many connectors, you can use the search box on the page to filter the connector list. Type part of the connector name to narrow down the results.

Example: A pipeline may have 184 connectors. Typing part of the connector name in the search box will show only matching connectors.

9.2 Understanding EDD (External Data Descriptor) within a Application Connector

Each connector can have one or more EDDs. An EDD is the first processing step inside a connector. It reads data from the source file and passes it into the connector for further processing.

On the Data Ingestion Errors page, connectors are shown at the top level. When you expand a connector row, you can see the EDDs that belong to it.

9.1 Connector Level vs. EDD Level

Copyright © 2000, 2026, Oracle and/or its affiliates.

Level	What It Shows	Where You See It
Connector level	The overall status of each connector in the pipeline	Main table on the page
EDD level	The individual EDDs within a connector and their error details	Shown when you expand or click a connector row

> **Note**
>
> To differentiate between **Connector errors** and **EDD errors**, check the name displayed at the top of the error details table.
>
> - If it shows a **connector name**, the errors are at the connector level
> - If it shows an **EDD name**, the errors are at the EDD level

What Happens When a Connector Is Not Executed

If a connector was not executed during the pipeline run, the page shows the following:

What You See	What It Means
No entry (empty row) for the connector	This connector was not executed. It was skipped during this pipeline run.
NA shown for the EDD inside that connector	Because the connector did not run, the EDD inside it also did not run. The status shows as **Not Available (NA)**.
Zero values for an executed connector	A connector that ran but had no errors will show zero counts, not NA.

Example: If a pipeline has five connectors and only three are executed:

- The first three connectors will show actual results, including record counts and errors
- The remaining two connectors will show no data (empty fields)
- Their EDD details will appear as **Not Available (NA)**

> **Note**
>
> **NA** means the connector or EDD was **not executed** during that run. It does **not** indicate an error.

What Happens When Errors Occur at the EDD Level

If a record fails at the **EDD stage**, it does not move to the next step. The pipeline stops processing that record at EDD itself.

This means:

- The error is captured and stored in the **EDD error table**
- The record does **not** go to the connector
- The record does **not** reach the final target table
- Only **EDD-level errors** are shown for that record

Copyright © 2000, 2026, Oracle and/or its affiliates.

- There will be **no connector-level error** for the same record

Example: A pipeline has **100 records**:

- **20 records** fail at the EDD stage → stored as EDD errors
- These 20 records do **not** move forward
- Only the remaining **80 records** go to the connector stage

9.3 Common Error Types

This section lists the error codes that appear at the EDD and Application Connector levels. It helps you understand the issue and fix it. For each error, it explains what went wrong and gives simple steps to resolve it, so you can easily troubleshoot problems during data integration.

Errors are divided into two types:

1. **EDD / Source File errors** – issues with the data file you uploaded
2. **Application Connector errors** – issues when sending data to the target system

EDD / Source File Errors

These errors happen before the data reaches the target system. They are caused by issues in the uploaded source file, archive, or control file, such as a missing file, incorrect column format, unsupported archive type, or data mismatch during reconciliation.

Error Code	Error Message	What to Do
DIE0007	Number of columns in the source file are less than expected	Check your source file for missing columns or wrong delimiters. Compare the file layout to the expected EDD structure.
DIE0004	Invalid date format in the source file	Open the source file and verify that all date columns use the format required by the EDD (e.g., DD/MM/YYYY vs MM/DD/YYYY).
DIE0010	No data files found	Confirm the file name matches the expected pattern and that the correct file is included in the archive.
DIE0019	Unable to locate the file in the Object Store	Re-upload the source file and ensure the file path/placeholder used in the configuration is correct.
DIE0016	Unsupported ZIP compression method. Only deflate is supported	Re-create the ZIP file using standard Deflate compression (e.g., using WinZip, 7-Zip with Deflate, or standard OS zip tools).
DIE0014	Reconciliation check failed — control record mismatch	Compare your control file values against the source file. Recount records or recalculate totals and regenerate the control file.
DIE0012	Cannot reconcile — expected control value is null	Review the control file and ensure all expected fields are populated with valid values.

Copyright © 2000, 2026, Oracle and/or its affiliates.

Error Code	Error Message	What to Do
DIE0013	Actual control value is null — defaulting to zero for control check	Check the source file for missing values in control columns. This may be a warning only — review if downstream totals are affected.
DIE0009	Invalid threshold specified	Review the threshold settings in the data integration configuration. Contact your administrator if you do not have access to change these settings.

Application Connector Errors

These errors occur when the system attempts to insert or update records in the target database. They are typically caused by data quality issues in the source — such as duplicate records, missing mandatory values, incorrect data types, or values that exceed column limits.

Error Code	Error Message	What to Do
DIE0001	Cannot insert Null into a Not-Null column	Identify the column flagged in the error log. Ensure the source data contains values for all mandatory fields before re-uploading.
DIE0002	Duplicate records encountered while loading into target table	Remove duplicate rows from the source file. If records already exist in the system, check whether a delta/update load should be used instead of a full load.
DIE0003	Cannot insert non-numeric value into a number column	Review the source data for the affected column and remove any non-numeric characters (e.g., currency symbols, spaces, text).
DIE0005	Invalid date format due to incorrect mappings	Check the connector mapping for date columns. Verify that the source column being mapped contains valid dates in the expected format.
DIE0006	Value too large — numeric precision overflow	Check the source data for unexpectedly large numeric values. Confirm the target column's precision settings with your administrator.

Copyright © 2000, 2026, Oracle and/or its affiliates.

10
Data Extraction

The Data Extraction module is part of Data Services in the Data Foundation for Banking application. Use Data Extraction to define and manage extracts that select attributes from the Data Catalog and write them to files, optionally filtered, for use by downstream reports and other consuming systems. An extract can include Derived Terms alongside Business Terms, so calculated and derived attributes can be delivered together with standard catalog attributes. Only Non-Aggregated Derived Terms can be included in an extract.

10.1 Accessing Data Extraction

1. From the application Home page, select the menu icon (≡) in the top-left corner to open the navigation panel.
2. In the navigation panel, select Data Services to expand the section.
3. Select Data Extraction from the expanded list of options. The Summary page is displays the count o how many definitions are saved. Use Search to filter by name as you type. Use Sort to change the order; default is Name, A–Z. Refresh reloads the list. Add opens a blank form for a new definition. Each card shows the name, short label, Saved status, progress, last modified details, and a delete icon.

> ⓘ **Note**
>
> You can also navigate to **Data Services** from the Home page, click on **Data Extraction**. This will display the basic details of the extraction definitions.

Understand Extraction Behavior: DFCS allows user-configured extraction of data from **Staging** and **Results entities** using **Connectors**.

As Of Date Behavior: If the **Extraction Entity** contains the **As Of Date** column, the data will be filtered based on the **PMF execution date**. If no date is selected during PMF Execution, the data will be filtered based on the **current date**.

Default Filter for Active Nodes: The default filter for dimension extraction is "Execution Date between Record Start Date and Record End Date", instead of the "As of Date."

No Need to Provide As Of Date Filter: Since the PMF execution date is used as the default for extraction, you are **not expected** to provide a filter on the **As of Date** during connector creation.

10.2 Adding a Data Extraction

To add a Data Extraction, follow these steps:

1. From the **DFCS Home** page, select **Data Extraction**.

   The **Summary** page is displayed with a list of data extraction definitions.
2. Click **Add**. The **Specify Details** window appears.

Copyright © 2000, 2026, Oracle and/or its affiliates.

Table 10-1 New data extraction definition

Field	Description
Name	A unique name for the data extraction definition. For Example: Extract CASA.
Select Grain	Select the **Select Grain** from the dropdown list, then choose the appropriate business subject area. For Example: Customer Account. Accounting Entries, Corporate Actions, Exchange Rates, Management Ledger.
Select Entity	The specific data source within the chosen grain (for example, Casa, Cards, Term Deposit Contracts, Trading Account). The list of entities available depends on the grain selected. Select the Select Entity from the dropdown list, then choose the data source to extract from.
Description	A free-text explanation of what the extraction is for. This field is optional.
Filter Expression	An optional expression used to filter which records are included in the extraction. Select the pencil icon to build or edit the expression. Enter the required filter expression for the Data Extraction. In addition, you can also add Parameters. For Example: [Casa].[Account or Contract Number] is not null. This filters out rows where the account or contract number is not null. **Note** For filtering data on Dimension Codes, you can specify member values for dimension surrogate key attributes only for the result table entities.

3. Click **Validate** to verify the correctness of the SQL Expression and click **OK**.

   The filter expression is displayed in the **Expression** text box.

4. Navigate to the **Select Attributes** page.

   The **Attributes list** displays the list of attributes that can be associated with the Data Extraction. The attribute list shows the combined attributes for all the tables selected.

5. Select the attributes and click **OK**.

   Select or unselect the attributes by clicking the respective icons. You can also drag and drop to rearrange the attributes in the **Selected Attributes** section.

Note

You can select a maximum of 998 attributes.

Copyright © 2000, 2026, Oracle and/or its affiliates.

> **Note**
>
> You can click Edit icon and rename the header attribute name, if required.

> **Note**
>
> Post upgrade, attributes of existing result entity extract connectors must be de-selected and selected again to auto rename attributes and validation on duplicate columns in selected attributes.
>
> For new connectors, attributes will be automatically renamed when selected. Auto-renaming of attributes to its corresponding dimension code attributes in the user interface is done only for Result table surrogate keys.
>
> Out-of-the-Box or custom attributes common across multiple dimension entities will not be displayed in the Select Attributes list. Related entities linked to the primary entity that do not have child attributes will not be displayed from the connector definition.
>
> Data Extraction for Derived Terms is supported only for staging entities.

> **Note**
>
> Only Non-Aggregated Derived Terms can be selected for extraction. A Non-Aggregated Derived Term is calculated at the same grain as the source entity (for example, a ratio or flag calculated per record), rather than summarized across records. Aggregated Derived Terms are not available for selection

6. Navigate to the **Describe File** page.
7. In **Specify Data file names**, enter the file name. This is a mandatory field.

   For example: test_file_data.txt

   > **Note**
   >
   > The file extension must be either .txt or .csv.

8. Select **Column delimiter** from the drop down list.

   For Example: Comma
9. Specify the **Text qualifier**.

   For Example: Double quotes can be used, prefixed, and suffixed with text.
10. Select the **file format**.

    For Example: Fixed Length or Delimited.
11. Select the **Record delimiter**.

    For Example: Unix
12. Toggle the **Header** button if you want a header to be displayed in the extracted file.

    This field is optional.

Copyright © 2000, 2026, Oracle and/or its affiliates.

> **Note**
>
> The extracted file will show Result, Dimension, Integration, and Staging tables. Currently, Processing layer tables are not supported.
>
> In case two physical entities contain the same logical name, the order of preference for the entity will be in the following order:
>
> - 1 - Result
> - 2 - Dimension
> - 3 - Integration
> - 4 - Staging

> **Note**
>
> The default file size threshold for splitting volume data is 6.5 GB.

13. Toggle the **Generate a control file** button if you want to generate a control file. The control file provides verification/validation information for use with extracted data files.

    This field is optional.

    In the **Specify control file name** field, enter the file name. This is a mandatory field. The file extension must be either .txt or .csv.

    The parameters you entered for **Specify Data file names** are copied to the Control file properties. You can modify based on your requiements. By default, the name is added as **< Specify Data file names >_control.csv/.txt** . For more details on fields such as Select file format, Select Column delimiter, Select Record delimiter, and Specify Text qualifier, see the above step.

    To add the controls, click Add button.

    - Add the control name in the text box.
    - Select the aggregation type from the dropdown. The supported options are Count, Avg, Sum, Min, and Max.
    - Select the attribute from the drop-down. The Attribute field is disabled when the aggregation type is selected as Count. The numerical attributes supports the aggregation type Sum, Average, Min, and Max. The Date attribute supports the aggregation type Min and Max only.

    You can add multiple controls to the file by clicking on Add button.

    The Control file is available in the Archived Extract file.

    > **Note**
    >
    > The following are the behavior of the Control file in this release:
    >
    > - There is no validations performed while generating a control file.
    > - Header will not be generated in case of control file during data extraction and you will not have an option to add the header.

Copyright © 2000, 2026, Oracle and/or its affiliates.

14. Enter the **Date Attributes Format** with a default value set to the current default Java date format.
15. Enter the **Date Time Attributes Format** with a default value set to the current default Java date format.

    For all the Java Date formats supported by Simple Date format class in Java SE 21, see Simple Date FormatSimple Date Format.
16. Click **Save**.

    The Data Extraction Definition is created.
17. After the Data Extraction is saved, the PMF process needs to be configured and executed.

    The Data Extraction file is created.
18. Download the file from the File Operations screen. For more details, see Generating PAR URL for File Operations.

10.3 Copying a Data Extraction

To copy a Data Extraction connector, follow these steps:

1. From the DFCS Home page, select **Data Extraction**. The Summary page is displayed with a list of data extraction definitions.
2. Click the **Copy** icon. The Copied data extraction Details window appears.
3. Enter the name and the Description.
4. Enter the required filter expression for the Data Extraction. In addition, you can also add Parameters.
5. Click **Validate** to verify the correctness of the SQL Expression and click **OK**. The filter expression is displayed in the Expression text box.
6. Navigate to the **Select Attributes** page. The Attributes list displays the list of attributes that can be associated with the Data Extraction. The attribute list shows the combined attributes for all the tables selected.
7. Select the attributes and click **OK**. Select or unselect the attributes by clicking the respective icons. You can also drag and drop to rearrange the attributes in the **Selected Attributes** section.
8. Navigate to the **Describe File** page.

    For more information, see section.
9. Click **Save**. The copied Data Extraction Definition is created.

Copyright © 2000, 2026, Oracle and/or its affiliates.

11

Metadata Report

The Metadata Export feature lets you export the metadata mapping for the Data Service Objects into one Excel workbook. The workbook can be used with a metadata lineage tool to help you understand how data moves and is transformed across your organization. You can export metadata for Ingest Connectors, Slowly Changing Dimensions, Process Connectors, Application Connectors, and Data Extracts.

This chapter shows you how to find Metadata Export, create an export, generate the workbook, and download the results.

11.1 Accessing Metadata Export

Metadata Export is part of the Data Services module.

- From the DFCS home page, navigate to **Data Services** > **Export Metadata**.

  This opens the Metadata Export summary page, which lists every export that has been defined.

11.2 The Metadata Export Summary Page

The Summary page lists all metadata exports created by you and other users, along with their current status. Use the search bar to find an export by its name, Data Service type, or status.

For each export, the page shows:

- Export Name: The name entered when the export was created.
- Type: The Data Service type included in the export, such as Ingest, Process, SCD, or Application.
- Status: The current stage of the export process.
- Created On and Created By: When the export was created and the user who created it.
- No of Objects: The number of Data Service objects included in the export.

Export statuses

Status	What it means
Saved	The export definition has been created but has not yet been generated.
Generating	The report is being generated in the background. You can continue working while it runs.
Generated	The report was generated successfully and is ready to download.
Partially Generated	Some objects were generated successfully, but others were not. Download the logs to identify the failed objects.
Failed	The export could not be generated. Download the logs for error details.

Actions menu

Copyright © 2000, 2026, Oracle and/or its affiliates.

Click the ••• icon at the right of any row to open the actions menu for that export. The options available depend on the export's current status:

- Edit: Update the export description or change the selected objects. The export name and Data Service type cannot be changed.
- Delete: Permanently remove the export definition from the Summary page.
- Generate: Start creating the metadata report. This option is available when the export status is Saved or Ready.
- Download: Download the generated report or its log file. This option is available after the export has finished generating.

11.3 Creating a New Metadata Export

Follow these steps to define a new export.

1. On the Metadata Export summary page, click **+ Create New Export**.
2. Enter an **Export Name**. This is required and must be unique.
3. Optionally enter an Export Description that explains the report purpose or scope.
4. Select a **Metadata Type** from the dropdown — for example, Ingest Connectors, Slowly Changing Dimensions, Process Connectors, Application Connectors, or Data Extracts. Once saved, the Export Name and Metadata Type cannot be changed.
5. Review the objects shown in Available. Select the objects to include, then use **>** to move selected objects to Selected. Use **>>** to move all available objects. Use **<** or **<<** to remove one or all objects from Selected.
6. Select **Save**. The new report appears on the list page with the Saved status.

   If you need metadata from more than one Data Service Type in the same report, create separate exports per type, or contact the administrator, since this determines whether the Metadata Type field allows more than one selection.

11.4 Viewing or Editing an Export

From the summary page actions menu:

- Choose View to open the export in read-only mode and review the objects it includes.
- Choose Edit to change the Export Description or adjust the Selected objects list. The Export Name and Metadata Type are locked once the export has been saved and cannot be edited.

Save your changes to update the export, or Cancel to leave it unchanged.

11.5 Generating a Report

When an export is in Saved status, it is ready to be generated.

1. Open the actions menu for the export and choose **Generate**.
2. Report generation runs in the background, so you can continue working in DFCS while the report is being created.
3. The export status updates automatically as generation progresses:
   - Saved → Generating → Generated (or Partially Generated, or Failed).

Copyright © 2000, 2026, Oracle and/or its affiliates.

> **Note**
>
> Metadata Report Name and Data Service Type cannot be changed after creation. To change either, create a new report.

4. Refresh the Summary page, or select the Refresh icon in the top banner, to view the latest export status.

11.6 Downloading the Report and Logs

Once an export is no longer Saved or Generating, the Download option become available from the actions menu:

- Download Report: downloads the metadata export as an Excel workbook.
- Download Logs: downloads a consolidated Excel file with the generation log for every object included in the export. Use this file to identify why an object was skipped or failed, especially when the export status is Partially Generated or Failed.

Copyright © 2000, 2026, Oracle and/or its affiliates.

12

Dimension Transformation

The Dimension Transformation in the Data Foundation for Banking is a data ingestion tool that loads dimension data (members, attributes, and hierarchies) from staging tables into PBSM interface tables using a PMF pipeline. It supports structured loading via preparation tables and integrates with cloud services where the loaded data can be viewed. The Data File Specification is not applicable to Dimension Data Transformation. The file format and the file names are static in nature.

Purpose of the Dimension Transformation

The Dimension Transformation performs the following key functions:

- Loads **Dimension Members** and their **Attributes** from the staging area into the Dimension Tables registered with the OFS Cloud Service framework.
- Creates **Hierarchies** for Key Dimensions in the Cloud Service.
- Loads **Hierarchical Relationships** between Key Dimension Members into the Cloud Service.

The following are the features of Dimension Transformation:

- Loading Simple Dimensions from Staging Tables.
- Multiple Hierarchies can be loaded from Staging Tables.
- Validations of Members and Hierarchies are similar to that of being performed within the Cloud Service Screens.

Pre-requisites

Before initiating the Dimension Transformation:

- Upload the **Data Files** containing dimension details.
- Ensure that each file includes a valid **Dimension Identifier** for both key and simple dimensions.

System Components and Workflow

The dimension loading process involves the following stages:

- **Source Systems**: These are the systems where raw and reference data is sourced:
  - **Stage Dimensions Master**: Stores raw dimension data received from upstream transactional or master systems.
- **Dimensions Reference**: Contains metadata for validation and enrichment of staging data.

1. **Step 2: Launch Dimension Transformation Screen**
   a. Click the **LHS** menu.
   b. Expand **Administration**.
   c. Expand **Configuration**.
   d. Scroll down within the **Configuration** menu and select **Dimension Transformations**. The **Dimension Transformations** page displays the available dimension list.

Copyright © 2000, 2026, Oracle and/or its affiliates.

> **Note**
>
> Dimension transformations cannot be migrated through Object Migration. You must manually create and configure the required transformations in each environment.

12.1 Adding a Custom Dimension

Dimension Transformations enables you to register a custom dimension entity from the Data Catalog for loading and maintaining in DFCS. After registration, the custom dimension is managed in the same manner as an out-of-the-box (OOTB) dimension.
When you add a custom dimension, the application automatically identifies key attributes, including Numeric ID, Code, Name, Description, and related technical columns. If available, the application also identifies the dimension hierarchy. Manual column mapping is not required.

This section describes how to use the Dimension Transformations page to view existing dimensions and add a custom dimension.

> **Note**
>
> Only entities registered in the Data Catalog as custom dimensions are available in the Dimension list. If the required entity is not listed, contact your Data Catalog administrator to register it.

Accessing the Dimension Transformations Screen

The Dimension Transformations page lists all dimensions available in the environment, including out-of-the-box (OOTB) and custom dimensions. For each dimension, the page displays a summary of its attributes, translations, and hierarchies. To open the Dimension Transformations page, navigate to Administration or Configuration from the left navigation menu in Data Foundation for Banking. The available navigation path can vary by environment and user role. The Dimension Transformations page displays the total number of dimensions—for example, 29 items—and details for each dimension.

Field	Description
Search	Searches for a dimension by name or code.
Created By and Source filters	Filters the list by the user who created the dimension or by source type.
Refresh	Reloads the dimension list.
Add Dimension	Opens the wizard to register a custom dimension.
Sort By	Changes the list sort order, for example, Name: A to Z.
Dimension row	Displays the dimension name, a brief description, dimension code, source (Seeded for OOTB dimensions or Custom), created by, and the number of attributes, translations, and hierarchies.

Step 1 : Add New Dimension

1. On the Dimension Transformations page, click Add Dimension.
2. From the Dimension list, select the custom entity to load.

Copyright © 2000, 2026, Oracle and/or its affiliates.

Only entities registered as custom dimensions in the Data Catalog are displayed. After an entity is configured, it is no longer available in this list.

3. In the Dimension Code field, enter a code for the dimension.

> **Note**
>
> The dimension code must meet the following requirements: Is unique for each dimension. Begins with UD-. Contains 4 to 30 characters. Contains only uppercase letters, numbers, and hyphens. For example: UD-CUST-DIM

4. Click **Continue**.
Step 2: Review

The Review page summarizes the configuration to be created. The application automatically identifies the dimension attributes and, when available, the related hierarchy. You do not select these items manually.

Field	Description
Dimension	The custom entity selected in Step 1.
Dimension Hierarchy	The related hierarchy entity automatically identified from the Data Catalog. Displays N.A. when no hierarchy entity is available.
Dimension Code	The code entered in Step 1.
Attributes	Number of additional, nontechnical attributes identified for the dimension.
Translation	Number of translation-related attributes identified, typically Name and Description.
Hierarchies	Number of hierarchy entities identified for the dimension. Displays 0 when no hierarchy is available.

5. Review the configuration details.
6. Click **Submit**. To change the selection or dimension code, return to Step 1. To discard the configuration, click **Cancel**.

Copyright © 2000, 2026, Oracle and/or its affiliates.

> **Note**
>
> The following attributes are added by default in each custom dimension.
>
> - Surrogate Key
> - Record Start Date
> - Record End Date
> - Record Creation Date
> - Latest Record Indicator
> - Seeded Record Flag
> - Dimension Value Closed Flag
> - As Of Date
> - Numeric ID
> - Code
> - Name
> - Description
> - Leaf Only Flag

Mapping Configuration

Seed the following DFCSCOMPMETA tables with mappings for each dimension.

Table 12-1 Dimension Loader Mapping Tables

Table	Description
DF_AMH_LOAD_MASTER	Stores the source and target entities and other master information for Dimension Loader.
DF_AMH_LOAD_FILTER	Stores filter conditions for mappings.
DF_AMH_LOAD_MAPPING	Stores base and translation mappings.
DF_AMH_LOAD_ATTR_MAPPING	Stores mappings for functional attributes.
DF_AMH_LOAD_DEFAULT_MAPPING	Stores default mappings, including mappings required for dimension hierarchies.

The dimension hierarchy entity must contain the following physical columns:

- V_HIERARCHY_CODE
- N_PARENT_ID
- N_CHILD_ID
- N_DISPLAY_ORDER_NUM
- V_DATA_SOURCE_CODE

12.2 Execution through PMF

Pre-requisite: Before executing the PMF, you must configure the necessary settings to ensure that the Dimension Loader Attributes are populated with accurate conversion values.

Copyright © 2000, 2026, Oracle and/or its affiliates.

1. a. i. Enter the following:
   - **Object ID** (e.g., DIM_AMH_Load_20250131_1)
   - **Application Params**:

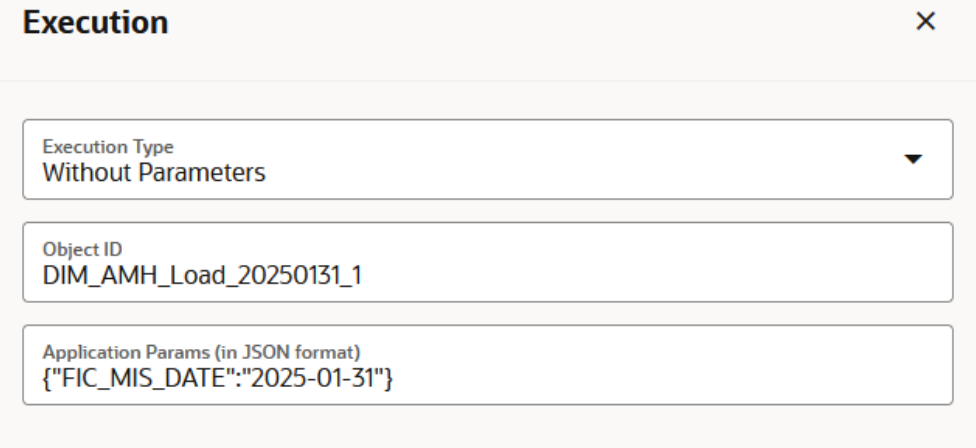
     ```
     json

     {"FIC_MIS_DATE":"2025-01-31"}
     ```

   Run the component to begin data load for the selected.

   ```
   FIC_MIS_DATE
   ```

Figure 12-1 Dimension Transformation - Execution



2. **ADS Connectors**
   - Extract data from Preparation Tables
   - Apply additional formatting if needed
   - Transfer data securely to PBSM Data Schema
- Reprocess failed loads
- Investigate errors
- Monitor success rates and traceability
- File Format and Specification:
  - Data File Specification is **not required** for Dimension Transformation.
  - File **format and names are static** and predefined within the framework.
- Once the ingestion process is completed, start the PMF Execution process. For more information, see PMF Execution Process.

Process Modeller

The Process Modeller is used to create and modify types of Pipelines, test the Process Flow, and execute the Run.

The Process Modeller page displays the existing Business Process Pipelines and Run Pipelines with the details such as Process ID, Process Name, Process Description, Version, Instance, Application, and Last Modified.

You can perform the following tasks from the Process Modeller page:

Copyright © 2000, 2026, Oracle and/or its affiliates.

- Click **Create** to create a new Pipeline.
- Click the Process Name link to launch and edit the Process Flow.
- Click the **Delete** icon to delete a Pipeline.
- Click the **More** icon to view the following menu:
  - Click **View** to see the process flow.
  - Click **Copy** to create a new Pipeline with the same process flow.
  - Click **Process Flow Monitor** to monitor the Pipeline.
  - Click **Execute Run** to execute a Run Pipeline.
- Use the Search grid to search for a specific Pipeline by providing a keyword from the Process ID, Process Name, or Process Description and then click **Search**. Click **Reset** to reset the Search fields.
- You can sort the Pipelines based on Process ID, Process Name, or Application. Click **Columns** and select the required attribute to sort.
- Click the **Process Monitor** icon to launch the Process Monitor page.

Process Monitor

The Process Monitor is used to supervise the current stage of the process for different instances. After integration with an application, you can invoke the workflow. After invoking, the workflow goes through all the stages defined. The Process Monitor displays all the completed stages, current stage, and future stages. Your user group must be mapped to the function role WFMACC (Workflow Monitor Access) to access the Process Monitor page.

This page displays all the workflows, which are invoked from the application with details such as Entity Name, Process Code, Process Description, Execution Start Time, Last Execution Time, Last Updated By, and Status.

You can perform the following tasks on the Process Monitor page:

- Click the **three dots** to view the following submenu :
  - View: To view the process for dimension loader components.
  - Abort: To abort an ongoing Run Pipeline process.
  - Resume: To resume a Run Pipeline process.
  - Re Run: To execute a Run Pipeline process again irrespective of the previous execution status.
- Use the Search grid to search for a specific Pipeline by providing a keyword from the Job ID, Process Name, or Process Description, and then click **Search**. Click **Reset** to reset the Search fields.
- You can sort the Pipelines based on Job ID, Process Name, or Application. Click the Sort by list and select the required attribute to sort.
- You can use the **Search** bar to filter pipelines based on the pipeline type. For example, to view only the Run Pipelines, remove Process from the Filter Pipeline field.
- Click the **Process Modeller** icon to open the Process Modeller page.

To view the details of the selected component, follow the below steps.

1. Select the required process ID to view **DF Components**.
2. Click on the three dots from the DF Components option and click **Properties**. The Activity details pane is displayed.

Copyright © 2000, 2026, Oracle and/or its affiliates.

3. Review the details and view the execution logs.

Execution Logs

To view the Execution Logs for Dimension Loader, follow the below steps.

- Navigate to the Activity Details pane, scroll down and click on **Execution Logs**. The **Logviewer Service** screen is displayed.

> **Note**
>
> The Logviewer Service is used for viewing, monitoring, and analyzing logs generated during a batch job or process execution.

- **Download Button:** Export logs in CSV or other supported formats.
- **Last Refreshed:** Displays the timestamp of the last log refresh.
- **Auto Refresh Toggle:** Enables or disables automatic refresh of the log feed.
- **Batch Run ID:** Unique identifier for the job execution.
- **Process ID:** Identifies the specific subprocess within the batch job.
- **Log Filter Section**:
  - Tabs to filter logs by **Severity Level**: ALL, DEBUG, INFO, WARN, ERROR, FATAL
  - Search bar to quickly locate logs using keywords.
  - **Time Elapsed:** Duration since the job started.
  - **Total Logs:** Count of log entries recorded.
  - **Timezone:** Local time zone of the log timestamps.
  - **Log Frequency Chart:** Visual graph of log volume over time.

12.3 For Key Dimensions

The following is a list of sample files that you can use to build the Dimension Data.

The name of the Data Files must be same as mentioned below and the file extension must be .DAT. The value of column HIERARCHY_CODE must not contain space in files STG_DIMENSIONS_HIER_INTF, STG_HIERARCHIES_INTF and STG_HIERARCHY_LEVELS_INTF. HIERARHY_CODE must be in capital letters.

> **Note**
>
> Any column description that contains a ",", then it should be enclosed within "" (double quotes). For example, Account, Type should be "Account, Type".

To load the Dimensions:

1. Define a new Batch and save it.
2. Add the following Tasks to the above Batch:

Copyright © 2000, 2026, Oracle and/or its affiliates.

Table 12-2 Dimension Loading Process Tasks

Task Code	Task Name	Component	Parameters	Parent Task
1	Stage Dimension Loader	Stage Dimension Loader	All parameters are automatically generated.	
2	Stage Hierarchy Loader	Stage Hierarchy Loader	All parameters are automatically generated.	1
3	Stage DRM Loader	Stage DRM Loader	• **Dimension Name**: Select the relevant one or more dimensions. • **Sync Stage and Dimension**: – **Yes**: The record(s) that is/are already present will be overwritten by the incoming dimension loader. – **No**: The new records will be merged to the existing records. • **Force Member Delete**: This is used only when the above flag is Yes. – **Yes**: This allows you to delete a members even if is referred in hierarchies. – **No**: No records will be deleted.	2

Copyright © 2000, 2026, Oracle and/or its affiliates.

> **Note**
>
> The above Tasks must be executed in the same order.
> The **File to Stage** task must precede the **Stage to Processing** task in a batch.
>
> The Stage DRM Loader allows you to select a Dimension.

3. Execute the Batch.

> **Note**
>
> This method of loading dimensions will be deprecated in future releases. Recommended approach is in the following section.

Dimension Loader with ZIP File Support

To to upload the Dimension data using a zip file:

1. Create a Folder.
2. Follow the naming convention for the DAT files as follows and save them:
   - DIMENSIONS_ATTRIBUTES
   - DIMENSIONS_BASE
   - DIMENSIONS_HIERARCHIES
   - DIMENSIONS_TRANSLATION
   - HIERARCHIES
   - HIERARCHY_LEVELS
3. Place all the DAT files inside the folder.
4. Zip the folder.
5. Create a Batch.
6. Create a Task with the Component name as **Dimension and Hierarchy Loader**.
7. Enter the Zipped Folder Name along with the extension. For example, DIMENSIONS.ZIP.
8. Execute the Batch.

12.4 For Simple Dimensions

The following is a list of sample files that you can use to build the Dimension Data.

The name of the Data Files must be same as mentioned below and the File Extension must be .DAT.

> **Note**
>
> Any column description that contains a ",", then it should be enclosed within "" (double quotes). For example, Account, Type should be "Account, Type".

To load the Dimensions, perform the following steps:

Copyright © 2000, 2026, Oracle and/or its affiliates.

1. Define a new Batch and save it.
2. Add the following Tasks to the above Batch:

Table 12-3 Dimension Loading Process Tasks

Task Code	Task Name	Component	Parameters	Parent Task
1	Stage Dimension Loader	Stage Dimension Loader	All parameters are automatically generated.	
2	Stage DRM Loader	Stage DRM Loader	• **Dimension Name**: Select the relevant one or more dimensions. • **Sync Stage and Dimension**: – **Yes**: The record(s) that is/are already present will be overwritten by the incoming dimension loader. – **No**: The new records will be merged to the existing records. • **Force Member Delete**: This is used only when the above flag is Yes. – **Yes**: This allows you to delete a members even if is referred in hierarchies. – **No**: No records will be deleted.	1

Copyright © 2000, 2026, Oracle and/or its affiliates.

> **Note**
>
> The above Tasks must be executed in the same order.
> The **File to Stage** task must precede the **Stage to Processing** task in a batch.
>
> The Stage DRM Loader allows you to select a Dimension.

3. Execute the Batch.

> **Note**
>
> This method of loading dimensions will be deprecated in future releases. Recommended approach is in the following section.

Dimension Loader with ZIP File Support

To to upload the Dimension data using a zip file:

1. Create a Folder.
2. Follow the naming convention for the DAT files as follows and save them:
   - DIMENSIONS_BASE
   - DIMENSIONS_TRANSLATION
3. Place all the DAT files inside the folder.
4. Zip the folder.
5. Create a Batch.
6. Create a Task with the Component name as **Dimension and Hierarchy Loader**.
7. Enter the Zipped Folder Name along with the extension. For example, DIMENSIONS.ZIP.
8. Execute the Batch.

12.5 Clear and Back up Dimension Data

This process helps you to clear or delete the existing Dimension Data from the relevant Dimension tables using the Scheduler Services.

Before clearing the data from the Dimension tables, the service creates a back up of the table.

To clear the Dimension Data:

1. Navigate to **Operations and Processes**, select **Scheduler**, and then select **Define Batch**.
2. Create the Batch and save it.
3. Navigate to **Operations and Processes**, select **Scheduler**, and then select **Define Task**.
4. Select the created Batch and create a Task with **Clear Dimension Members** as Component.
5. Select the Dimension Name that you want to delete. You can select one or more Dimension Names.
6. From the LHS menu, navigate to **Operations and Processes**, select **Scheduler**, and then select **Execute Batch**.

Copyright © 2000, 2026, Oracle and/or its affiliates.

The service first creates a backup of the existing Dimension table and then deletes the Dimension Member entries for the selected Dimensions.

While deleting the data from the tables, there are no validations. The back up files are suffixed with As Of Date and Current Time Stamp.

The following table gives the sample actual and backed up table names:

Table 12-4 Sample Table Names

Dimension Type	Actual Table Name	Backup Table Name
Simple Dimension	FSI_ACCRUAL_BASIS_CD	FSI_ACCRUAL_BASIS_CD_<AS_OF_DATE>_<CURRENTTIMESTAMP>
	FSI_ACCRUAL_BASIS_MLS	FSI_ACCRUAL_BASIS_MLS_<AS_OF_DATE>_<CURRENTTIMESTAMP>
Key Dimension	DIM_COMMON_COA_ATTR	DIM_COMMON_COA_ATTR_<AS_OF_DATE>_<CURRENTTIMESTAMP>
	DIM_COMMON_COA_B	DIM_COMMON_COA_B_<AS_OF_DATE>_<CURRENTTIMESTAMP>
	DIM_COMMON_COA_HIER	DIM_COMMON_COA_HIER_<AS_OF_DATE>_<CURRENTTIMESTAMP>
	DIM_COMMON_COA_TL	DIM_COMMON_COA_TL_<AS_OF_DATE>_<CURRENTTIMESTAMP>

12.6 Data Preparation Guidelines

While creating the data files, ensure the following:

- Ensure the data files are in TXT, DAT, or CSV formats.
- Ensure that there are no duplicate records in a data file.
- Data file names are in the prescribed format.
- Use only comma (,) and/or pipeline (|) as delimiters.
- Only double quotes (“”) are used as Field Enclosures.
- The language code must be as per BCP 47 format. For example, **en-US**.
- In the file for hierarchies, there must be no empty space or special characters for HIERARCHY_CODE.
- Dimension member name must not contain **&** character.

Copyright © 2000, 2026, Oracle and/or its affiliates.

13

Data Pipelines

Refer to Oracle® Financial Services Data Foundation Cloud Service for Banking Data Operations Guide for more details.

Copyright © 2000, 2026, Oracle and/or its affiliates.

14

Key Terms and FAQs

Table 14-1 Key Terminologies

Term	Description
DFCS	**Data Foundation Cloud Service** - acts as the Publisher
PBSMCS	**Profitability and Balance Sheet Management Cloud Service**- acts as the Subscriber
ADS	**Application Data Services** that handle metadata exchange between tenants
PMF	**Process Modeller Flow** that enables data movement post-subscription
Wallet	Secure credentials used for establishing tenant subscriptions
BT	In OCI applications, the Business Term is applied to support and standardize DFCS integration workflows.

Copyright © 2000, 2026, Oracle and/or its affiliates.