Oracle® Fusion Cloud EPM

Administering Data Integration

E98147-99

Oracle Fusion Cloud EPM Administering Data Integration,

E98147-99

Copyright © 2017, 2026, Oracle and/or its affiliates.

Primary Author: EPM Information Development Team

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

Documentation Accessibility

Documentation Feedback

1 Creating and Running an EPM Center of Excellence

2 Data Integration

Prerequisites 1
Using Cloud EPM Business Processes in Data Integration 3
- Customers Using Planning 4
- Loading Data to a FreeForm Application 4
- Customers Using Financial Consolidation and Close 6
- Customers Using Tax Reporting 8
- Customers Using Account Reconciliation 10
- Customers Using Strategic Workforce Planning 12
- Customers Using Enterprise Profitability and Cost Management 12
- Customers Using Profitability and Cost Management 13

3 Launching Data Integration

Viewing the Data Integration Home Page 1
Working with the Data Integration Workflow 3

4 Integration Tasks

System Settings 1
Setting Security Options 6
- Assigning Role Access Security to Report Groups 6
- Role Level Security 7
- Defining Location Security 8

Setting User Level Profiles 10

Copyright © 2017, 2026, Oracle and/or its affiliates.

Viewing Locations 12
Bulk Managing POV Lock Statuses 12
Copying an Integration 14

5 Registering Applications

Launching Applications Options 2
Navigating Applications 2
Registering Cloud EPM Local Applications 5
Registering Cloud EPM Applications 6
Registering Data Source Based Applications 7
- Registering Oracle ERP Cloud General Ledger Applications 8
- Registering Oracle NetSuite Applications 9
  - Updating An Oracle NetSuite Application 12
- Registering Incremental File Applications 12
- Defining an On-Premises Data Source Application 13
- Registering Oracle Cloud ERP Applications 15
  - Updating Oracle Cloud ERP Applications 16
- Registering Oracle Cloud ERP Custom Applications 17
- Registering Peoplesoft GL Balance Applications 18
- Registering E-Business Suite Applications 19
  - Connecting to E-Business Suite and Peoplesoft Data Sources 20
- Registering Oracle HCM Cloud Applications 22
- Registering an On-Premises File 23
- Registering File Applications 24

Registering an Application for a Class of Dimensions or Dimension Type 25
Registering a Data Export File Application 27
- Registering a Data Export File as an EPM Data File Type Application 28

Application Details 29
- Defining Application Dimension Details 30
  - Adding Lookup Dimensions 32
- Defining Application Detail Options 32
  - Defining Application Details for Oracle ERP Cloud Data Sources 38
  - Refreshing Oracle ERP Cloud Source 39

Editing Options 39
Setting Default Options 42
Refreshing Members 43
Refreshing Metadata 44

6 Configuring Source Connections

Configuring an Oracle ERP Cloud Connection 1

Copyright © 2017, 2026, Oracle and/or its affiliates.

Configuring an Oracle HCM Cloud Connection 3
Configuring an Oracle NetSuite Connection 4
Configuring an Oracle Autonomous Database Connection 4

7 Managing Period Mappings

Global Mappings 3
Application Mappings 4
Source Mappings 6
Source Mappings for a File Source Type 6
Source Mappings for a Data Source Type 7
Source Mappings for a Cloud EPM Source Type 8
Source Mappings for an Oracle ERP Cloud Source Type 9
Source Mappings for an Oracle HCM Cloud Source Type 10
Source Mappings for an Oracle NetSuite Source Type 11
Period Mapping Options 12
Importing Period Mappings from Excel 13
Exporting Period Mappings to Excel 13
Downloading a Period Mapping Import Template 14
Deleting Period Mappings 15
Loading Multiple Periods for Cloud EPM or File-Based Source Systems 16

8 Managing Category Mappings

Global Mappings 1
Application Mappings 2
Selecting Members for the Target Category 3

9 Defining a Data Integration

Creating File-Based Integrations 3
Selecting Location Attributes 5
Specifying File Options 6
Previewing File Options 8
Editing a File Column Header 8
Using the File Browser 9
Using Quick Mode for File-Based Loads 10
Quick Mode - File-Based Load Considerations 10
Quick Mode for File-based Load Process Description 11
Defining Periods in a Quick Mode - File-Based Load 18
Purging Files 18
Creating Direct Integrations 18

Copyright © 2017, 2026, Oracle and/or its affiliates.

Mapping Dimensions 19
Creating the Dimension Maps 19
Using Target Expressions 23
Using Source Expressions 29
Mapping Members 32
Adding Member Mappings 33
Using Explicit Mappings 37
Using Between Mappings 38
Using In Mappings 39
Using Like Mappings 40
Using Special Characters in the Source Value Expression for Like Mappings 40
Using Special Characters in the Target Value Expression 43
Using Multi-Dimensional Mappings 43
Using Special Characters in Multi-Dimensional Mapping 45
Format Mask Mapping for Target Values 45
#FORMAT Mapping Type Components 46
#FORMAT Mapping Example 46
Replacing Segments 47
Replacing Segments with String Operations 47
Replace Segments with String Operations and Using a Prefix or Suffix 47
Selecting Members 47
Filtering Members 49
Import Member Mappings 49
Importing Excel Mappings 50
Downloading an Excel Template (Mapping Template) 52
Restoring Member Mappings 53
Exporting Member Mappings 54
Exporting the Current Dimension or All Dimensions Map 54
Exporting the Map to Excel 54
Setting Data Integration Options 56
Defining File-Based Options 56
Defining Direct Integration Options 58
Defining Target Options 60
Data Export Target Application Options 66
Defining Oracle General Ledger Target Options 69
Defining Target Options for Budgetary Control Applications 70
Defining Application Options for Financial Consolidation and Close 70
Defining Target Options for an Account Reconciliation Application 76
Defining Filters 77
Defining Planning Filters 77
Defining Oracle NetSuite Filters 78
Defining Budgetary Control Filters 79

Copyright © 2017, 2026, Oracle and/or its affiliates.

Defining Oracle HCM Cloud Filters 80
Defining Custom Filters to a MDX Query 81
Defining A Clear Region 81
Using Business Rules 84
Registering a Business Rule in Independent Mode 84
Business Rules Supported Events 86
Adding Run Time Prompts to Business Rules 86
Registering a Business Rule in Embedded Mode 88

10 Defining Data Maps

Understanding Data Maps 1
About Applications 2
About Reporting Cubes 2
Unsupported Features with Data Maps 2
Creating Data Maps 3
Refreshing Data Maps 4
Managing Data Maps 4
Viewing Data Maps 4
Changing Mapping Information for a Dimension 5
Considerations for Target Selections 6
Clearing and Restricting Target Data Selections for Mappings 6
Using Detailed Member-Level Mapping 9
Defining a Simple Member-Level Mappings 10
Defining a Roll-up Mapping 10
Defining a Multi-Dimension Mapping 11
Defining a Substitution Variable Mapping 12
Moving Unmapped Dimensions 14
Populating Detail Mappings 15
Excluding Source Members from a Data Map 16
Defining Data Map Options 18
Editing Details for a Data Map 20
Duplicating a Data Map 21
Exporting a Data Map to Excel 21
Importing a Data Map 22
Deleting a Data Map 24
Validation Rules for Selected Members 24
Moving Data 25
Moving Data to a Reporting Cube 25
Moving Data Using a Cross Application Data Map 26
Moving Data from One Cube to Another Cube Using Smart Push 27
About Smart Push 27

Copyright © 2017, 2026, Oracle and/or its affiliates.

Considerations When Using Smart Push 28
Configuring Smart Push for a Form 29
Using a Cross Application Smart Push 31
Configuring Merge Options 32
Moving Data After Configuring Smart Push 33
Data Push and Smart Push Performance 33
Running Data Maps and Smart Push Processes Simultaneously 33
Checking Real Time Progress Status for Data Maps and Smart Push 34
Viewing the Smart Push Status 35
Identifying Smart Push Intersections with Unrecognized Smart List Values 36
Viewing the Status of the Data Maps Job 36
Viewing the Strategic Modeling Push Status 37

11 Running an Integration

Viewing the Integration Job 4

12 Using the Pipeline

Pipeline Considerations 2
Pipeline Tasks 2
Copying a Pipeline 2
Pipeline Process Description 3
Pipeline Command Reference 12
Using the Table View 13
Pipeline Job Types 14
Using a Business Rule Job Type 15
Using a Business Ruleset Job Type 16
Using a Clear Cube Job Type 18
Using a Copy from Object Storage Job Type 21
Using a Copy from SFTP Job Type 23
Using a Copy to Object Storage Job Type 25
Using a Copy to SFTP Job Type 27
Using a Create Reconciliations Job Type 29
Using a Data Load Essbase Rules File Job Type 29
Using an EPM Platform Job Job Type for Planning Jobs 30
Using an EPM Platform Job Job Type for Financial Consolidation and Close and Tax Reporting Jobs 39
Using an EPM Platform Job Job Type for Enterprise Profitability and Cost Management 41
Using an Export Dimension by Name (EDMCS) Job Type 43
Using an Export Dimension Mapping by Name (EDMCS) Job Type 47
Using an Export Mapping Job Type 49

Copyright © 2017, 2026, Oracle and/or its affiliates.

Using an Export Metadata Job Type 50
Using an Extract Dimension Viewpoint (EDMCS) Job Type 51
Using an Extract Package (EDMCS) Job Type 52
Using a File Operations Job Type 54
Using a Generate Report for Account Reconciliation Job Type 59
Using an Import Attribute Values Job Type 60
Using an Import Balances Job Type 62
Using an Import Mapping Job Type 62
Using an Import Metadata Job Type 64
Using an Import Pre-Mapped Balances Job Type 65
Using an Import Pre-Mapped Transactions (TM) Job Type 66
Using Import Rates Job Type 67
Using an Integration Job Type 68
Using an Integration with Smart Split Job Type 70
Using Open Batch Job Types 81
Using an Open Batch - File Job Type 82
Using an Open Batch - Location Job Type 84
Using an Open Batch - Name Job Type 86
Using a Plan Type Map Job Type 87
Using a Run Auto Alert Job Type 88
Using a Run Auto Match Job Type 88
Using a Set Period Status Job Type 90
Using a Set Substitution Variable Job Type 90
Using a Wait Job Type 92
Editing Runtime Variables 92
Adding Custom Variable Parameters 96
Adding a Substitution Variable as an Input Parameter 96
Running the Pipeline 97
Sending Pipeline EMAIL Notifications 99
Allowing Non-Administrators to Execute Jobs in the Pipeline 100

13 Reviewing Process Details and Viewing Results in the Workbench

Viewing Process Details 1
Viewing an Error Messages Output File 3
Resetting an Integration or Pipeline Job Status 4
Using the Workbench 5
Using the Workbench Workflow 6
Importing Source Data 7
Validating Source Data 7
Exporting Data to Target Applications 8
Checking the Data 9

Copyright © 2017, 2026, Oracle and/or its affiliates.

Showing Data in the Workbench 9
Exporting Data in the Workbench 9
Viewing Target Data 11
Deleting Imported Data in the Workbench 12
Viewing Mappings in the Workbench 13
Viewing the Source Document in the Workbench 14
Filtering Workbench Data 14
Defining a Custom View in the Workbench 16
Validating Source Data 18
Validation with No Mapping Errors 18
Validation Mapping Errors 19
Fixing Mapping Errors 19

14 Exporting Data

Using the Standard Workflow 1
Creating a Data Export File Integration 1
Quick Mode to Export Data 5
Quick Mode to Extract Data Process Description 6
Creating an EPM Data File 13

15 Drill Through

Using Drill Through to Source 2
Defining Drill Details 3
Defining Drill Regions 9
Drill Through to a Landing Page 10
Drill Through to a Custom URL Page 11
Adding a Custom View to the Drill Through Landing Page 12
Defining Direct Drill 20
Drill through to the Oracle ERP Cloud 20
Drill through to a BI Publisher Report in the Oracle ERP Cloud 21
Drilling Down to a BI Publisher Report in the Oracle ERP Cloud Process Description 21
Drill through to the SAP® General Ledger in Cloud EPM 27
Drill through to Source using the EPM Integration Agent 29
Creating a Drill Down to a Source System Without a Landing Page 31
Drill through to Oracle Autonomous Database Source Without Using the EPM Integration Agent 33
Drilling Through to Legacy On-premises ERP Systems 35

Copyright © 2017, 2026, Oracle and/or its affiliates.

16 Synchronizing Data

Data Synchronization Process Description 1
Quick Mode Data Synchronization 3
Quick Mode Data Synchronization Process Description 4

17 Integrating Data

Integrating Oracle General Ledger Balances from the Oracle ERP Cloud 1
Integration Process Description 2
Configuring a Source Connection 3
Registering an Oracle General Ledger Application 4
Creating an Integration with the Oracle ERP Cloud (GL Balances) 5
Defining Location Attributes 6
Using Category Mappings 7
Mapping Dimensions 7
Mapping Members 9
Defining Oracle General Ledger Filters 12
Processing Oracle General Ledger Adjustment Periods 12
Running an Integration 14
Writing Back to the Oracle ERP Cloud 18
Writing Back Actuals to the Oracle General Ledger 18
Writing Back Budgets to the Oracle General Ledger 21
Integrating Metadata from the Oracle ERP Cloud 24
Loading Metadata Process Description 24
Integrating Oracle ERP Cloud Data 26
Process Description for Integrating Oracle ERP Cloud Data Using Prepackaged Queries 27
Configuring an Oracle ERP Cloud Connection 28
Registering Oracle ERP Cloud Applications 29
Applying Application Filters to an Oracle ERP Cloud Data Source 30
Selecting Period Report Parameters from the Oracle ERP Cloud 33
Process Description for Integrating Oracle Cloud ERP Data Using a Custom Query 33
Registering a BI Publisher Report as an Oracle Enterprise Scheduler (ESS) Job 42
Process Description for Registering ESS Jobs 42
Defining the Data Extract Process 43
Creating the Data Model for the BI Publisher Report Extract 45
Creating the BI Publisher Report Extract 52
Creating an Oracle Enterprise Scheduler (ESS) Job 58
Creating the Integration Definition Between Cloud EPM and the BI Publisher Report Extract 62
Loading Oracle ERP Cloud Exchange (FX) Rates to the Cloud EPM 66
Process Description for Loading Foreign Exchanges (FX) Rates Description 66
Security Role Requirements for Oracle ERP Cloud Integrations 71

Copyright © 2017, 2026, Oracle and/or its affiliates.

Integration User Privileges 71
Integration User Predefined Roles 72
Integration User Custom Roles 72
Allowlist 72
Integrating EPM Planning Projects and Oracle Fusion Cloud Project Management (Project Management) 72
About Integrating EPM Planning Projects and Project Management 73
Process Description for Integrating EPM Planning Projects and Project Management 73
Security Role Requirements for Cloud EPM to Project Management Integrations 76
BI Publisher Security 76
Projects and Budgets File-Based Data Import Process Security 76
Registering the Project Management Source 77
Defining the Integration 78
Registering the EPM Planning Projects Application 78
Classifying Project Dimensions in the EPM Planning Projects Application 79
Register the Reporting Type Application 80
Registering the Dimensions Application 81
Initializing the Integration 82
Integration Definition Reference 84
Post Initialization Mappings 84
Mapping Custom Dimensions 86
Defining Period Mappings 86
Defining Application Mappings 87
Applying Source Mappings 89
Integrating Budgetary Control 91
Loading Budgetary Control Budget Consumption Balances to Planning 91
Loading Budgetary Control Budget Consumption Balances to Planning Process Description 91
Writing Back Cloud EPM Budget Balances to the Budgetary Control 100
Writing Back Cloud EPM Budget Balances to the Budgetary Control Process Description 100
Viewing the Cloud EPM Budgets Loaded to Budgetary Control 107
Integrating Budget Revisions with Budgetary Control 110
Process Description 110
Registering the Planning Application with OEP_FS as an Input Cube 112
Registering the Oracle ERP Cloud Source System 112
Registering Budgetary Control as a Data Source and Importing Applications 113
Registering the Planning Budget Revision 114
Registering a Budget Adjustment Request 115
Integrating the Planning Budget Revision and Budget Adjustment Request 116
Mapping the Planning Budget Revision and the Budget Adjustment Request Dimensions 118
Mapping Custom Budget Revision Attributes 119

Copyright © 2017, 2026, Oracle and/or its affiliates.

Defining Budget Adjustment Options 119
Integrating Oracle NetSuite 120
Process Description for Integrating Oracle NetSuite 121
Creating an Oracle NetSuite Data Source 123
Mapping Columns in the Saved Search to Dimensions 123
Adding a Target Dimension for Oracle NetSuite Data Sources 124
Defining Oracle NetSuite Application Options 124
Applying Oracle NetSuite Application Filters 125
Adding Additional Filters to the Drill URL in the Import Format 126
Managing Periods in Oracle NetSuite 126
Adding a Target Dimension for Oracle NetSuite Data Sources 127
Adding Additional Filters to the Drill URL in the Import Format 128
Drilling Through to Oracle NetSuite 129
Defining Drill Through Parameters to Oracle NetSuite 129
Saved Search Requirements in the Drill Through 130
Adding the Drill Through URL 130
Adding Additional Filters to the Drill URL in the Import Format 131
Integrating with the Oracle HCM Cloud 132
Process Description for Integrating Data from Oracle HCM Cloud 132
Downloading Oracle HCM Cloud Extracts 136
Importing Oracle HCM Cloud Extract Definitions to Oracle HCM Cloud 137
Importing the Oracle HCM Cloud Extract Definition 137
Importing the BI Publisher eText Templates 140
Validating and Submitting the Oracle HCM Cloud Extract Definition 142
Configuring an Oracle HCM Cloud Connection 143
Registering an Oracle HCM Cloud Application 144
Updating Existing Oracle HCM Cloud Extracts 146
Integrating Oracle HCM Cloud Metadata 147
Loading Oracle HCM Cloud Metadata 148
Oracle HCM Cloud Extract Definition Field Reference 151
Account Merit Extract Definition Fields 152
Assignment Extract Definition Fields 152
Component Extract Definition Fields 153
Employee Extract Definition Fields 153
Entity Extract Definition Fields 154
Job Extract Definition Fields 154
Location Extract Definition Fields 154
Position Extract Definition Fields 155
Integrating SAP 155
Integrating with JD Edwards 156
Integrating Account Reconciliation Data 156
Loading Reconciliation Compliance Balances 156

Copyright © 2017, 2026, Oracle and/or its affiliates.

Loading Reconciliation Compliance Balances Process Description 156
Creating a Reconciliation Compliance Balances Integration 157
Mapping Dimensions for a Reconciliation Compliance Balances Application 158
Mapping Members for Reconciliation Compliance Balances 160
Running the Reconciliation Compliance Balances Integration 161
Integrating BAI, SWIFT MT940, and CAMT.053 Format Bank File Balances 162
Adding a Transaction Matching Target Application 169
Aggregating Transaction Matching Data 172
Integrating BAI, SWIFT MT940, and CAMT.053 Format Bank File Transactions 174
Verifying BAI Bank Statement Balances 184
Verifying BAI Bank Statement Balances Considerations 185
Enabling the Verify Statement Balance Option 186
Loading Reconciliation Compliance Transactions 187
Loading Reconciliation Compliance Transactions Process Description 187
Registering a Reconciliation Compliance Transactions Application 188
Creating a Reconciliation Compliance Transactions Integration 190
Mapping Reconciliation Compliance Transactions Attributes to Dimensions 191
Mapping a Data Member for Reconciliation Compliance Transactions 193
Defining Application Details for a Reconciliation Compliance Transactions Application 194
Defining a Period for Reconciliation Compliance Transactions 195
Running the Reconciliation Compliance Transactions Integration 195
Loading Exported Journal Entries 197
Integrating Metadata 201
Metadata Load File Considerations 202
Loading Metadata Process Description 203
Registering a Target Application for the Class of Dimension or Dimension Type 205
Loading Data Using an Incremental File Adapter 205
Setting up the Incremental File Adapter 206
Bulk Loading Workforce New Hire Data 210
Bulk Loading Process Description 211
Defining Incremental File Adapter Options 216
Integrating With Oracle Autonomous Database Data 218
Process Description for Integrating Oracle Autonomous Database 218
Connecting to the Oracle Cloud Infrastructure 219
Configuring an Oracle Autonomous Database Source Connection 219
Defining an Oracle Autonomous Database Source Connection Without Using Object Storage 219
Defining an Oracle Autonomous Database Source Connection Using Object Storage 220
Creating Database Tables 221
Creating External Tables 221
Creating Standard Tables 222

Copyright © 2017, 2026, Oracle and/or its affiliates.

Creating Data Source Application 222
Creating Query-based Data Source Application 222
Creating Table-based Data Source Application 223
Editing Application Details 223
Editing Integration-Level Filters Associated With Oracle Autonomous DB Data Source 224
Setting Up an Integration 225
Running Integration to Load Data from Oracle Autonomous Database to Cloud EPM Without an Agent 226

18 EPM Integration Agent

Installing and Configuring the EPM Integration Agent 2
Configuring and Starting the EPM Integration Agent Process Description 2
Upgrading the EPM Integration Agent 3
Setting JAVA_HOME 4
Java Runtime Environment and EPM Automate 5
Downloading the EPMAgent ZIP 5
Creating Application Folders 5
Secure Sockets Layer (SSL) Certificate 6
Configuring the Microsoft JDBC Driver for SQL Server in the EPM Integration Agent 6
Configuring a SSL Connection to an Oracle Database 7
Configuring a Custom JDBC Driver for SQL Data Sources in EPM Integration Agent 8
Configuring the EPM Integration Agent Startup Parameters 9
Encrypting the Cloud EPM User Password 15
Using OAuth 2.0 Authorization Protocol with OCI (Gen2) Environments 17
Process Description for Using OAuth 2.0 Authorization 17
Setting Up Authentication with OAuth 2 18
Enabling the OAUTH Option in the EPM Integration Agent 22
Running the createoauthtoken.bat or createoauthtoken.sh File to Generate Tokens 24
Running the EPM Integration Agent as a Windows Service 25
Starting the EPM Integration Agent 26
Running the EPM Integration Agent in Diagnostic Mode 27
Configuring Synchronous Mode 28
Cloud EPM to On-premises Connectivity Using the EPM Integration Agent 30
Using the EPM Integration Agent 31
Understanding EPM Integration Agents 32
EPM Integration Agent Process Description 32
Extracting Data from On-premises Data Sources 33
Editing Application Options 37
Creating an SQL Query 38
Using Prepackaged On-premises Adapters 43
E-Business Suite General Ledger Preseeded Query Reference 46

Copyright © 2017, 2026, Oracle and/or its affiliates.

Peoplesoft General Ledger Preseeded Query Reference 48
Importing Data from a File in the EPM Integration Agent Inbox 49
Using an Agent File Adapter 51
Performing a Quick Mode Load using the EPM Integration Agent 54
Quick Mode Process Description 56
Writing Back with the EPM Integration Agent 67
Process Description for EPM Integration Agent Write-Back 67
Registering the Data Export to On-Premises Application 69
Setting the Application Detail Dimensions for the Write-Back 70
Defining Application Details for A Data Export Application 71
Practical Techniques for Writing Back to the EPM Integration Agent 74
EPM Integration Agent Write-Back Event Scripts 83
Understanding Agent Clusters 90
Adding a Cluster 91
Assigning Integration Jobs 92
EPM Integration Agent Scripting 93
Events 93
EPM Integration Agent API Methods 94
Integrating Oracle Autonomous Database Data Using the EPM Integration Agent 96
EPM Integration Agent Context Functions 126
EPM Integration Agent Script Example 127

19 Reports Execution

Audit Reports 2
Account Chase - Freeform (Target Account,Period,Category) 2
Account Chase - Wildcard (TargAcct, Per, Cat) 2
Intersection Drill Down (Per,Cat) 3
Map Monitor for Location 3
Map Monitor for User 4
Base Trial Balance Reports 5
TB Current Location, by Target Acct (Cat, Per) 5
TB Current Location, By Target Entity-Acct (Cat, Per) 6
TB Current Location with Integration (Cat, Per) 6
TB Current Location, with Targets (Cat, Per) 6
TB All Columns (Per,Cat,Loc) 7
TB All Source Columns (Per,Cat) 7
TB All Target Columns (Per,Cat) 8
TB Converted Current Loc by Target Entity/Account 8
TB Current Loc, All Dimensions-Targets, by Target Entity-Account (Cat, Per) 9
TB Current Loc, All Dimensions-Targets (Cat, Per) 9
Process Monitor Reports 10

Copyright © 2017, 2026, Oracle and/or its affiliates.

Process Monitor (Cat, Per) 10
Process Monitor All Categories (Per) 10
Process Status Period Range (Cat, Start Per, End Per) 11
Check Reports 11
Check Report 11
Check Report By Validation Entity Seq. 12
Check Report Period Range (Cat, Start Per, End Per) 12
Check Report With Warnings 13
Listing Reports 13
Import Formats by Location 13
Location Listing 14
Location Analysis 14
Dimension Map (Dimension) 14
Dimension Map For POV (Dimension, Cat, Per) 15
Variance Reports 15
Account Chase Variance (TargAcc,Cat1,Per1,Cat2,Per2) 15
Trial Balance Variance (Cat1,Per1,Cat2,Per2) 16

20 System Maintenance Tasks

Partition Data Load Staging Tables 1
Delete Integration 2
Export Setup Table Data 2
List Table Rowcount 3
Maintain Application Folder 4
Maintain Data Table by Application 4
Maintain Data Table by Location 5
Purge All Imported Data 6
Maintain Process Tables 7
Maintain Setup Data 8
Reallocate Tablespace 8
Reset Environment 8
Upgrade Custom Applications 9
Migrate Batches to Pipeline 9
Executing a Snapshot Export Job 10
Executing a Snapshot Import Job 11

21 Predictive Cash Forecasting and Data Integration

Copyright © 2017, 2026, Oracle and/or its affiliates.

A Performance Tuning in Data Integration

Job Log Analysis A-1
- Step 1: Stage and Load File to the TDATASEG_T table A-2
- Step 2: Clean up Prior Data Loads A-2
- Step 3: Process Mapping Rules A-3
- Step 4: Clean up Prior Integrations in TDATASEG A-3
- Step 5: Copy Mapping Results from TDATASEG_T to TDATASEG A-4
- Step 6: Validate Mapping Results A-4
- Step 7: Generate File and Load to Target A-5
- Job Log Additional Considerations A-5

Tuning Considerations A-5
- SQL Mapping A-5
- Expressions A-7
- Simple Workflow Mode A-7
- Quick Mode A-8
- Partitioning Tables A-8
- Additional Considerations A-9

B Building Reports Against the Data Integration Schema

AIF_BALANCE_RULES Table Definition Reference B-2
TDATASEG Table Definition Reference B-3
TLOGPROCESS Table Definition Reference B-9
TLOGPROCESSSTATES Table Definition Reference B-13
TPOVCATEGORY Table Definition Reference B-13
TPOVPARTITION Table Definition Reference B-13
TPOVPERIOD Table Definition Reference B-14

C Troubleshooting Data Integration Issues

D Data Integration Best Practices

Defining Data Flows D-1

E Scheduling Jobs

Scheduling a Data Rule, Integration, or Batch Definition to Run in the Job Scheduler E-1
Scheduling a Data Rule or Integration Definition to Run Later in the Job Scheduler E-5
Scheduling Hourly Jobs E-6
Scheduling Maintenance Scripts in the Job Scheduler E-9
- Partition Data Load Staging Tables E-9

Copyright © 2017, 2026, Oracle and/or its affiliates.

Deleting an Integration E-10
Export Setup Table Data E-11
List Table Rowcount E-12
Maintain Application Folder E-12
Maintain Process Tables E-13
Maintain Data Table by Application E-14
Maintain Data Table by Location E-14
Maintain Setup Data E-16
Purge All Imported Data E-16
Reallocate Tablespace E-17
Reset Environment E-18
Upgrade Custom Applications E-19
Executing a Snapshot Export Job E-19
Executing a Snapshot Import Job E-20
Migrate Batches to Pipeline E-21
Migrating Schedules to EPM Platform Job Scheduler E-22
Migrating Schedules to the EPM Platform Job Scheduler Important Considerations E-22
Migrating Schedules Best Practices E-23
Migrating Schedules to the EPM Platform Job Scheduler Process Description E-23

F Frequently Asked Questions (FAQ) about Data Integration

Copyright © 2017, 2026, Oracle and/or its affiliates.

Documentation Accessibility

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.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Documentation Feedback

To provide feedback on this documentation, click the feedback button at the bottom of the page in any Oracle Help Center topic. You can also send email to epmdoc_ww@oracle.com.

Copyright © 2017, 2026, Oracle and/or its affiliates.

1

Creating and Running an EPM Center of Excellence

A best practice for EPM is to create a CoE (Center of Excellence).

An **EPM CoE** is a unified effort to ensure adoption and best practices. It drives transformation in business processes related to performance management and the use of technology-enabled solutions.

Cloud adoption can empower your organization to improve business agility and promote innovative solutions. An EPM CoE oversees your cloud initiative, and it can help protect and maintain your investment and promote effective use.

The EPM CoE team:

- Ensures cloud adoption, helping your organization get the most out of your Oracle Fusion Cloud EPM investment
- Serves as a steering committee for best practices
- Leads EPM-related change management initiatives and drives transformation

All customers can benefit from an EPM CoE, including customers who have already implemented EPM.

How Do I Get Started?

Click to get best practices, guidance, and strategies for your own EPM CoE: Introduction to EPM Center of Excellence.

Learn More

- Watch the Cloud Customer Connect webinar: Creating and Running a Center of Excellence (CoE) for Cloud EPM
- Watch the videos: Overview: EPM Center of Excellence and Creating a Center of Excellence.
- See the business benefits and value proposition of an EPM CoE in *Creating and Running an EPM Center of Excellence*.

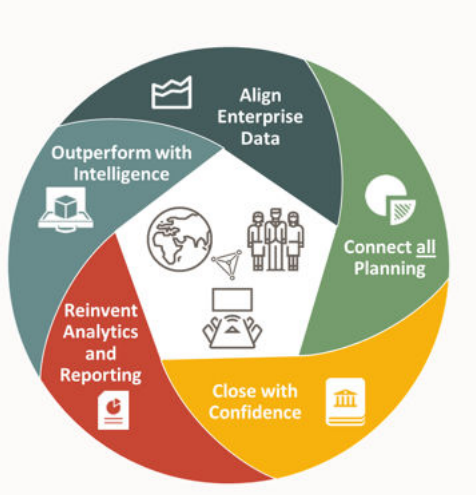
Copyright © 2017, 2026, Oracle and/or its affiliates.

Copyright © 2017, 2026, Oracle and/or its affiliates.

2

Data Integration

Data Integration is the mechanism by which integration processes are performed in the Oracle Fusion Cloud Enterprise Performance Management. Designed for busy administrators and users, you can define file-based and direct integration sources, create mapping rules to translate source data into the required target format, and execute and manage the periodic data loading process. Common integration tasks are done using an easy-to-navigate interface that supports and conforms to how you work.



> **Note**
>
> Data Integration will eventually replace Data Management once there is full parity between the features. Data Integration continues to be enhanced to move the legacy features from Data Management into Data Integration.

Watch this overview video to learn more about using Data Integration.

Overview: Using Data Integration in Planning and Budgeting Cloud.

Prerequisites

The prerequisites to use Data Integration are described below.

Data Integration is available for the following roles in the Planning, Tax Reporting, Financial Consolidation and Close, Account Reconciliation, Profitability and Cost Management, and Enterprise Profitability and Cost Management business processes:

- Service Administrator
- Data Integration - Manage granular role
  The Data Integration - Manage granular role can be assigned to any user, regardless of their application role, allowing broader access to all functional activities within Data

Copyright © 2017, 2026, Oracle and/or its affiliates.

Integration. Users with this role are able to: create and manage integrations between source and target systems, execute and monitor pipeline activities, perform data and metadata extraction and transformation from on-premises sources using the EPM Integration Agent. By default the Data Integration - Manage granular role is enabled.

- Power User—If a service administrator assigns the Data Integration – Create, Drill Through and Run roles to a Power User, the Power User can create an integration, view the details of an integration, edit or change the integration definitions, and run an integration between the source and target. The Power User can also drill through to the source system of the data. For more information, see Managing Granular Roles with Access Control in *Administering Access Control*
- User—If a service administrator assigns the Data Integration – Run and Drill Through roles to a user, the user can run an integration between the source and target, view the details of an integration but not edit or change the integration definitions unless they have been assigned a Service Admin or Power User Role. The user can also drill through to the source system of the data. For more information, see Managing Granular Roles with Access Control in *Administering Access Control*

Oracle Enterprise Data Management Cloud users do not have access to Data Integration.

> **Note**
>
> Sales Planning is available as a target application in Data Integration.

In Account Reconciliation, access to Data Integration is limited to users who have been assigned a Service Administrator application role only.

Data Management is still fully supported and is available as a menu selection on the Navigator menu. Between Data Integration and Data Management, integration-related activities are bidirectional and synchronized. For example, when you map dimensions in Data Integration, any saved mappings are populated automatically in Data Management.

For those customers using the EPM Integration Agent, see EPM Integration Agent.

The following are special considerations when using Data Integration.

Feature	Data Integration Special Consideration/Noteworthy Issues
Role	Data Integration is available for the following business processes: • Planning and Planning modules • Tax Reporting • Financial Consolidation and Close • FreeForm • Account Reconciliation • Enterprise Profitability and Cost Management Service administrators can also set up custom roles, which include: • The Data Integration - Manage granular role can be assigned to any user, regardless of their application role, allowing broader access to all functional activities • Create Integration—define an integration and associated setup and run the integration • Run Integration—Execute an integration and view mappings Profitability and Cost Management, Oracle Hyperion Public Sector Planning and Budgeting, and Oracle Enterprise Data Management Cloud do not have access to Data Integration.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Feature	Data Integration Special Consideration/Noteworthy Issues
Supported Business Services	Supported business services are: • Planning Modules • Planning • FreeForm • Financial Consolidation and Close • Tax Reporting • Strategic Workforce Planning • Profitability and Cost Management • Enterprise Profitability and Cost Management
Connections	Use the Configure Connections option in Data Integration to register, manage, and delete direct connection information to the following sources: • Oracle ERP Cloud • Oracle Human Capital Management Cloud • Netsuite
Applications	Applications shares the existing functionality of the Target Applications option in Data Management and enhances your experience with a more modern look and feel and responsive design. It is named "Applications" because it is used as the primary gateway to integrate data between different source systems and target applications.
Import Format	Fixed length file, including the Fixed Width – Numeric Data and Fixed Width – All Data types, are not supported.
Map Members page	• **Column Index** (column name) has been changed to **Processing Order**. • Mapping scripts are unavailable.
Options	Column Headers for multi-period loads are unavailable.
Run Integrations	Scheduling is available using the EPM Platform Job Scheduler .

> **Note**
>
> For business process to business process integrations, both business processes must be at the same release level.

When working with Data Integration, note the following terminology changes:

Data Management Name	Data Integration Name
Target Application	Applications
Import Format	Map Dimensions
Data Load Mapping	Map Members
Data Load Rule/Data Load Rule Name	Integration

Using Cloud EPM Business Processes in Data Integration

Data Integration is available as an integration option for the following business processes in the Oracle Fusion Cloud Enterprise Performance Management:

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Planning Modules
- Planning
- FreeForm
- Financial Consolidation and Close
- Tax Reporting
- Account Reconciliation
- Strategic Workforce Planning
- Profitability and Cost Management
- Enterprise Profitability and Cost Management

When transforming and loading data between Oracle Fusion Cloud EPM deployments (business processes), you may use Data Integration to process a source data file that is not in the required format for native loading in your application, or when Data Integration features are desired beyond the native data loading functionality.

In the following sections, notable differences are explained when customers are moving to the cloud version from the on-premises version of their products using Data Integration.

Customers Using Planning

For those Planning customers using Data Integration, here are key considerations:

- Partial Data Loads—When loading data, all valid data is loaded. If some of the data does not pass any cell level validation rules, then that data is not loaded, but all other valid data is loaded. A partial load is reflected as a failed integration, but all valid data is loaded. If users are defined as an administrator, then cell level validations are ignored, and the data is loaded.
- If you use an "all data type load" type load, you can create a job in Planning by selecting **Yes** for **All Data Type Load Using Jobs** in System Settings. For more information, see System Settings.
- When a customer selects **All data types with security** as a target option, the system loads numeric data, text, Smartlist, and Date data types. If the Planning administrator loads data, Planning data security *is not* enforced. If a Planning non-administrator user loads data, then Planning data security is enforced. A Planning non-administrator user can load only 500,000 cells of data.

Loading Data to a FreeForm Application

The FreeForm application type is a business process that recreates a native Oracle Essbase application within the Planning environment to support required analysis. FreeForm enables any dimensions to be associated with the cube. The Planning-specific Currency, Entity, Scenario, and Version dimensions, along with their member hierarchies are not required with FreeForm business processes. You use FreeForm to do your own modeling and build your own cubes while preserving the ability to leverage platform functionality. Service Administrators build a FreeForm application using an Essbase outline (OTL) file, or by manually defining the dimensionality. For more information, see Creating a FreeForm App in *Administering FreeForm* .

FreeForm applications are meant for customers who want to create an Essbase application for anything or to migrate an on-premises application to the Oracle Fusion Cloud Enterprise Performance Management because the FreeForm application in Planning does not have any dimension requirements.

Copyright © 2017, 2026, Oracle and/or its affiliates.

You can load these FreeForm applications in Data Integration, but there are a number of considerations:

> **Note**
>
> The following load requirements and considerations below also apply to **exporting** FreeForm applications since a user could create a FreeForm application and load it with the native load process, and not have it classified correctly in Data Integration, and then have issues exporting data.

1. The FreeForm application requires a minimum of three dimensions: Account, Period, and Scenario. The application definition in Data Management and Data Integration must have three dimensions with the dimension type of Account, Period, and Scenario.
2. You must set up a period mapping so that the system knows where to load the data. For example, you could set up period mapping with a period of Jan-20, which is the period member created in a FreeForm application. When you set up a period mapping in Data Management and Data Integration, you enter a period created in the FreeForm application and a year entry so that it passes the user interface validations for the period mapping. This is the case where you don't have to define a year dimension in your FreeForm application, only a period.
3. You must specify a Scenario dimension, but in Planning, it can be anything. The only requirement is that the dimension is classified as a Scenario dimension in Data Management and Data Integration. You then need to set up Category mapping so that the process succeeds.
   - Assign a dimension classification of Account for one of the dimensions.
   - If you want to use the drill through functionality, then a "Scenario" dimension is required. Assign a dimension classification of Scenario for one of the dimensions. Note when a dimension is classified as Scenario, the Category mapping is used to assign a target value, so data can be loaded to only one value. Select a dimension that meets this requirement and define a Category mapping.
   - If you want to use Check feature, then an "Entity" dimension is required. Assign a dimension classification of Entity for one of the dimensions.

   Applications of type ASO are not auto registered when they are created. Use the Data Management Target Application page and select the Application type of Essbase to manually register the application.
4. To load metadata to FreeForm applications with only ASO cubes, register the Dimension application as follows:
   a. Add a new EPM local application with cubes "All Input Cubes."

      Because there is no BSO application, this application does not display any dimensions.

   b. Add a new Dimension application.

      For more information, see Registering Applications.

      You can load metadata to the ASO cubes by defining an integration with the appropriate source using the dimension application as source.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Videos

Your Goal	Learn How
Get an introduction to FreeForm.	Introduction to FreeForm Apps in Cloud EPM
Create FreeForm apps from on-premises Essbase outline (OTL) files and snapshots.	Creating FreeForm applications from on-premises Essbase outline files and snapshots
Create multi-cube FreeForm apps with an Essbase outline	Creating Multi-Cube FreeForm Apps from Essbase Outlines
Create FreeForm apps with a Hybrid BSO cube	Creating FreeForm Applications with a Hybrid BSO Cube
Create FreeForm apps with an ASO or reporting cube	Creating FreeForm Applications with a Reporting/ASO Cube
Create multi-cube FreeForm apps with a BSO or ASO cube	Creating Multi-Cube FreeForm Apps from BSO and ASO Cubes
Learn how to create FreeForm apps using an Excel template. Using the Excel template, you learn how to define application properties, manage cubes, create dimensions and members, attributes, access permissions, and load data into the FreeForm application.	Creating Multi-Cube FreeForm Applications with an Excel Template - Part 1
Learn how to update FreeForm apps using an Excel template and verifying changes in the web interface. After creating your FreeForm application using an Excel template, learn how you can use the same template to make modifications to dimensions, members, substitution variables, and security. Then, in the Planning web interface, assign Application Management options, review the application properties and definition, and the data you imported.	Creating Multi-Cube FreeForm Applications with an Excel Template - Part 2

Customers Using Financial Consolidation and Close

For those Oracle Hyperion Financial Management customers moving to Financial Consolidation and Close, note the key differences between the two products:

- Financial Consolidation and Close shows a positive amount as a debit and a negative amount as a credit.
- A "plan type" is not a Financial Consolidation and Close concept.

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Financial Consolidation and Close users can load Oracle General Ledger data into their own applications.
- Users can export data from Financial Consolidation and Close to the Fusion GL as actuals.
- Data synchronization can push data from either Planning or Financial Consolidation and Close to a Financial Consolidation and Close target application.
- Financial Consolidation and Close can be used as a source system in the import format. In this way, you can use Financial Consolidation and Close as a source system, and then use a different cloud service (such as Planning Modules, Account Reconciliation, Planning, Profitability and Cost Management) as a target, and move data from Financial Consolidation and Close to these other cloud services.

  Additionally, you can pull data from Financial Consolidation and Close and push the data to a file for use in another application.
- For a consolidation dimension, you can load different override amounts and rates for different source members by location. This enables you to report on details used to perform the various stages of the consolidation process.
- You can protect override rates data that has been manually entered to the "managed data" data source when a load is done from Data Integration to the same entity by entering the data directly in the pre-defined override rates form in Financial Consolidation and Close. However, if a user loads general data to an entity in Financial Consolidation and Close using Data Integration and the same user needs access to enter the override rates in the form, then the load from Data Integration clears the override rates that have been entered.
- In addition to the system predefined dimensions, you can create up to four additional Custom dimensions based on your application needs. Custom dimensions are associated with the Account dimension and provide additional detail for accounts. If Extended Dimensionality is enabled for the application, you can create up to four Custom dimensions. If the application is enabled with the multi-GAAP reporting option, you can create three Custom dimensions.
- Data Integration supports a Financial Consolidation and Close "Period" as a column in a data file. If you have data for multiple periods in a single file, then you can include the Year and Period on each row of the data. In Map Dimensions, you select the source period rows of Year and Period, so the system knows that these columns are in the file, and then map them to the appropriate dimension in the target system. See Loading Multiple Periods for Cloud EPM or File-Based Source Systems.
- Data Integration supports an explicit load method for loading journals to Financial Consolidation and Close. Journals are loaded by defining an integration with the type "Journals." Both Excel and text-based journal loads are supported. See Loading Journals to Financial Consolidation and Close in *Administering Data Management for Oracle Enterprise Performance Management Cloud*.
- Drill through functionality is not supported for exchange rates data.
- The import modes available to Financial Consolidation and Close are "append" and "replace."
- For non-DSO applications (non Dense Sparse Optimization enabled application), when you import data from Financial Consolidation and Close and use an Explicit mapping set, do not use attribute columns ATTR2 and ATTR3 for any dimension mappings. Data Integration uses these columns to determine the correct period key for the row.
- Financial Consolidation and Close customers can extract dynamic calculated values by selecting the **All Data** option in Direct Integration Options (for more information, see Defining Direct Integration Options). It is a Data Integration prerequisite that the CONTROL TO-DATE VIEW STORAGE setting in Financial Consolidation and Close is enabled, or the Financial Consolidation and Close application is DSO (Dense Sparse Optimization enabled

Copyright © 2017, 2026, Oracle and/or its affiliates.

application) based, to extract dynamic calculated values. For more information, see Using the Control To Date View Option.

- The export modes available to Financial Consolidation and Close target application are:
  - **Merge**—Merge adds existing cells and create new cells if they do not exist. It does not add to the data.
  - **Replace** —The system first deletes all values based on the Scenario, Year, Period, Entity, and data source before it submits the load.
  - **Accumulate**—Accumulate the data in the application with the data in the load file.
  - **Dry Run**—Scan a data load file for invalid records without loading data it to the target application. For each unique point of view in the data file, the value from the load file is added to the value in the application. Dry Run is not applicable for Quick Mode loads.
- To load data to actual currency rather than entity currency when the currency is fixed, set the currency in the Functional Currency field in the Location option. You can also add a Currency row in the import format and map it.
- The Change Sign option is not supported for Quick Mode loads.
- Partial Data Loads—When loading data, all valid data is loaded. For example, if some of the data does not pass any cell level validation rules, then that data is not be loaded, but all other valid data will be loaded. A partial load is reflected as a failed integration, but all valid data is loaded. If a user is defined as an administrator, then cell level validations are ignored, and the data is loaded.
- When running an integration across instances in push mode, the credentials from the connection details are used to determine the load user, and not the user that submitted the integration for processing. If you set up the integration in the reverse manner in "pull" mode, then the user executing the integration drives the security when loading to the consolidation application.
- When loading as an administrator, the data load bypasses security including validation rules, and all data is loaded.
- With validation rules are turned on and the Enable Data Security for Admin Users option is enabled when an administrative user loads data or for a non-administrator load, the load blocks data from loading to the cells where validations apply but loads the rest of the data. The behavior when security is turned on is to perform a partial load, but then show the integration rule as failed.
- A cross instance data load uses the user-defined in the connection, and also uses that user to determine the mode to load the data, NOT the user that is executing the rule.
- In a data load file if you have movements that need to be loaded to both the FCCS_ClosingBalance_Input member and a specific movement member in the same .Data Integration, you can load the two movements members using Logic Accounts. First, create a Logic Group name, and then provide a name and description with criteria type such as "Like" and a value such as the "ABC" account with an operator N/A. Then assign the logic group name to a location under the logic account. In this integration, map the ABC account to any selected target account. Ten map the ABC account to any other account other than the prior target account.

Customers Using Tax Reporting

For those Oracle Hyperion Tax Provision customers moving to Tax Reporting, note the key differences:

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Balance data or exchange rates may be loaded to the Tax Reporting application using a file. (Data and exchange rates cannot be loaded by way of the same file.) In addition, balance data from the Oracle Financials Cloud may also be directly integrated to the Tax Reporting application. At this time, exchange rate loading from the Oracle ERP Cloud is not supported.
- Data is loaded to Tax Reporting at the summary account level. Line-item detail is not supported in Tax Reporting.
- Journals are not supported in Tax Reporting at this time. In Data Integration, only "data" load types are supported for Tax Reporting applications.
- Drill through *from* a Tax Reporting web form or Oracle Smart View for Office (dynamically linked to Tax Reporting) *to* Data Integration is supported.
- Drill through *to* a Tax Reporting web form *from* Data Integration is only available when Tax Reporting includes a URL that can be called from Data Integration.
- Drill through functionality is not supported for exchange rates data.
- Data loaded from Data Integration to Tax Reporting is summarized based on the dimensionality in Data Integration, and this summarized data is loaded to Tax Reporting. Any computations or consolidation logic is only performed within Tax Reporting.
- Tax Reporting supports "YTD" data only, and consequently no data is modified when it has been loaded.
- Tax Reporting customers can extract dynamic calculated values by selecting the **All Data** option in Direct Integration Options (for more information, seeDefining Direct Integration Options). It is a Data Integration prerequisite that the CONTROL TO-DATE VIEW STORAGE setting in Tax Reporting is enabled, or the Tax Reporting application is DSO (Dense Sparse Optimization enabled application) based, to extract dynamic calculated values. For more information, see Using the Control To Date View Option.
- The Change Sign option is not supported for Quick Mode loads.
- The export modes available to Tax Reporting target application are:
  - **Merge**—Overwrites existing data and creates new data if it does not exist. Merge does not add to data.
  - **Replace** —The system first deletes all values based on the Scenario, Year, Period, Entity, and data source before it submits the load.
  - **Accumulate**—Accumulate the data in the application with the data in the load file.
  - **Dry Run**—Scan a data load file for invalid records without loading data it to the target application. For each unique point of view in the data file, the value from the load file is added to the value in the application. Dry Run is not applicable for Quick Mode loads.

  > **Note**
  >
  > In Replace mode, before the first record for a specific Scenario/Year/Period/Entity/Mapped Data Source is encountered, the entire combination of data for that Scenario, Year, Period, Entity, and Mapped Data Source is cleared, whether entered manually or previously loaded. Note that when you have a year of data in the Planning application, but are only loading a single month, this option clears the entire year before performing the load.

- If you need to consolidate all entities as part of the data load process, in Data Integration, use the Check Entity Group option (see Creating Check Rule Groups in *Administering Data Management*).

Copyright © 2017, 2026, Oracle and/or its affiliates.

- The "ownership of data" feature in Tax Reporting is not supported in this release.
- The `rundatarule` command of the EPM Automate utility, which executes a Data Integration based on the start period and end period, can be executed for an Tax Reporting application.
- Data Integration can be used to move data between business processes. This means you can move data between Tax Reporting applications, or Tax Reporting data to and from other Oracle Fusion Cloud Enterprise Performance Management business processes.
- To load data to actual currency rather than entity currency when the currency is fixed, set the currency in the Functional Currency field when creating the integration. See Creating Direct Integrations.
- You can add a Currency row in the import format and map it. See Mapping Dimensions.
- After completing a data load cycle in Tax Reporting, data can be written out to a text file created in a custom application for use in an external application, or to an (on-premises) location. When the custom application is defined, you can export the file and download it using EPM Automate.
- For additional features available to Tax Reporting users, see the contents of this guide.

Customers Using Account Reconciliation

The process to implement an integration to Account Reconciliation, is to first test the import and validation steps in Data Integration. Once you're satisfied with the results, launch the actual integration for loading balances from the Data Load option within the Account Reconciliation. Transaction Matching or Reconciliation Compliance transactions are pushed from Data Integration as per the standard Oracle Fusion Cloud Enterprise Performance Management process.

Location security in not available in the Account Reconciliation user interface. Location security is available only in Data Integration user interface.

Lock/Unlock POV from the Application page in Data Integration is only available for administrators. In the Data Management user interface, you can lock an individual location from the POV page as non-administrator.

When you are an on-premise Accounts Reconciliation Manager customer moving to Account Reconciliation, note the following in Data Integration:

- Using Account Reconciliation as a source for other than Cloud EPM applications is not currently supported with Account Reconciliation.
- You cannot load to an Account Reconciliation environment remotely. If you want integrate application data from another business process with Account Reconciliation, then you need to register the application from the other business process in Account Reconciliation and not the other way around.
- Drill-through is supported from Account Reconciliation to the drill-through landing page in Data Integration. Account Reconciliation can also drill through to the data source in the on-premise Financial Data Quality Management, Enterprise Edition if the data has been first imported to Data Integration using the hybrid integration feature. To drill through to the landing page, the Account Reconciliation profile segments must be mapped to the LOOKUP dimension in the Target Dimension Class on the Target Application page.
- Integrating data from Oracle NetSuite is supported in Account Reconciliation.
- Integrating data from the Oracle Financials Cloud is supported in Account Reconciliation.

Copyright © 2017, 2026, Oracle and/or its affiliates.

- When mapping a "Source Type" dimension in the Data Load Mapping option for Account Reconciliation balances, use the **Target Value** as either the **source system** or **sub-system** (subledger). These options are a categorization that define the sources of the data. For example, if you select **source system**, this option does not necessarily indicate that the source data is from the same source system but does indicate that the data is not from a subledger, such as Accounts Receivable (AR), Accounts Payable (AP), and so on.
- Any Lookup dimension you add for mapping purposes should be of classification of "LOOKUP." Do not add any dimension of classification "Generic."
- When mapping to a Reconciliation Account ID within Account Reconciliation, there are considerations for blank target segments. If there are blank segments in between two populated segments in the target reconciliation, Account Reconciliation treats each blank/null segment values in the middle of a Reconciliation Account ID as three blank spaces. Account Reconciliation also trims off the trailing NULL segments after the last populated segment.

  For example, a Grouped Reconciliation with the following Reconciliation ID: "001-null-null-1925 XXX" (the "null" in Account Reconciliation would be non-existent (void of characters) when viewing the Reconciliation/Profile.) The design of Account Reconciliation replaces the "null" at the database tier with three spaces for each blank/null segment in between the populated segments. The Target Mapped Profile in Data Management needs the following: "ACCOUNT ID 001- - -1925 XXX" to align with Account Reconciliation.
- Account Reconciliation customers, who need to load bank statement files (which use a BAI or Bank Administration Institute file format or a SWIFT MT940 file format) to the Transaction Matching module in Account Reconciliation, can use Data Integration as the integration mechanism. Data Integration supports a pre-built adapter for loading:
  - BAI Format Bank File Transactions
  - BAI Format Bank File Balances
  - SWIFT MT940 Format Bank File Transactions
  - SWIFT MT940 Format Bank File Balances
  - Camt.053 Format Bank File Transactions
  - Camt.053 Format Bank File Balances

  > **Note**
  >
  > In addition, any other file format that Data Integration supports, can also be used to import, map, and load to the Transaction Matching module.
- As an integration mechanism, Data Integration enables Account Reconciliation customers to load Reconciliation Compliance Transactions into Reconciliations. You can pull transactions directly from the Oracle ERP Cloud to the Cloud EPM including:
  - Balance Explanations
  - Source System Adjustments
  - Subsystem Adjustments
  - Variance Explanations
- In Transaction Matching, you can export adjustments or transactions as dual sided journal entries that can then be imported into your ERP system. Data Integration is used as the mechanism to load the exported journal entries.

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Account Reconciliation customers can use up to 64 fields per transaction to load data using Data Integration to Transaction Matching.
- For Transaction Matching, you **do not** need to recreate the application in Data Integration when you take the following actions:
  - ID of a match type
  - ID of a data source
  - ID of a data source attribute
  - Name of a journal column
- For additional features available to Account Reconciliation users, see the contents of this guide.

Customers Using Strategic Workforce Planning

You can load Human Resources data from the Oracle Fusion Human Capital Management to use in the Strategic Workforce Planning business process of Oracle Fusion Cloud Enterprise Performance Management.

Strategic Workforce Planning customers who need to translate long-term corporate strategy into execution plans, can align and pull strategic data from Human Capital Management. Data Management is the integration mechanism. It provides an out of the box solution that enables customers to apply predefined mappings from Human Capital Management data model to target dimensions in Strategic Workforce Planning. Customers can also customize and extend these integrations, for example, by applying other mappings as needed to meet their business requirements.

Customers Using Enterprise Profitability and Cost Management

You can integrate data to and from the Enterprise Profitability and Cost Management to use in the Oracle Fusion Cloud Enterprise Performance Management. Enterprise Profitability and Cost Management is an analytic software tool that manages the cost and revenue allocations that are necessary to compute profitability for a business segment, such as a product, customer, region, or branch. Enterprise Profitability and Cost Management enables you to use cost decomposition, consumption-based costing and scenario-playing to measure profitability for effective planning and decision support.

For Enterprise Profitability and Cost Management customers using Data Integration, note the following:

- An Enterprise Profitability and Cost Management application can be registered as both a source and a target.
- Using Data Integration, Enterprise Profitability and Cost Management integrates with:
  - business processes within the Cloud EPM
  - Oracle ERP Cloud General Ledger and Sub-Ledger balances

  Enterprise Profitability and Cost Management does not integrate with EPM Planning, Projects, and Oracle Fusion Cloud Project Management (Project Management)
- When setting up an integration in an Enterprise Profitability and Cost Management instance, register it as a local application. If pulling from a separate instance, register the application as a Cloud EPM application.
- An Enterprise Profitability and Cost Management application is registered automatically when the application is created using the Create Application Wizard.

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Enterprise Profitability and Cost Management customers need to configure a source connection to other instances in the Cloud EPM and the Oracle ERP Cloud.
- Drill Through is available.
- When exporting values from Enterprise Profitability and Cost Management, you can round values to 2 decimal places, use the Round method.
- Data Maps and Smart Push are available. Data Maps supports an Aggregate storage (ASO) cube as a source cube.
- The EPM Integration Agent is available.
- You can only call Calculation Manager based business rules by way of the Pipeline. Enterprise Profitability and Cost Management based rules and rulesets is not supported.
- You can only call Calculation Manager based business rules by way of the Pipeline. Enterprise Profitability and Cost Management based rules and rulesets are not supported.
- There are only four platform job types supported in the Pipeline for Enterprise Profitability and Cost Management. For more information, see Using an EPM Platform Job Job Type for Enterprise Profitability and Cost Management.
- If you want to run something in Enterprise Profitability and Cost Management from Financial Consolidation and Close, use a groovy based business rule in Financial Consolidation and Close to call the Enterprise Profitability and Cost Management rest api. For more information, see Profitability and Cost Management REST APIs in *REST APIs* .

Customers Using Profitability and Cost Management

Profitability and Cost Management is an analytic software tool that manages the cost and revenue allocations that are necessary to compute profitability for a business segment, such as a product, customer, region, or branch. For this reason, Profitability and Cost Management is designed to integrate with key Oracle Fusion Cloud Enterprise Performance Management business processes. You integrate data to and from Profitability and Cost Management to use in the Cloud EPM using Data Integration.

As the integration mechanism, Data Integration, enables Profitability and Cost Management customers to load data to and from the Enterprise Resource Planning (ERP) such as the Oracle ERP Cloud, a consolidation process, or a planning and forecasting process.

For Profitability and Cost Management customers using Data Integration, note the following:

- An Profitability and Cost Management application can be registered as both a source and a target.
- Using Data Integration, Profitability and Cost Management integrates with:
  - business processes within the Cloud EPM
  - Oracle ERP Cloud General Ledger and Sub-Ledger balances

  Profitability and Cost Management does not integrate with EPM Planning, Projects, and Oracle Fusion Cloud Project Management (Project Management)
- When setting up an integration in an Profitability and Cost Management instance, register it as a local application. If pulling from a separate instance, register the application as an Cloud EPM application.

  Profitability and Cost Management supports only a single application per environment (single ASO database).
- An Profitability and Cost Management application is registered automatically when the application is created using the Create Application Wizard.

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Profitability and Cost Management customers need to configure a source connection to other instances in the Cloud EPM and the Oracle ERP Cloud.
- Drill Through is available.
- When exporting values from Profitability and Cost Management, you can round values to 2 decimal places, use the Round method.
- Data Maps and Smart Push are available. Data Maps supports an Aggregate storage (ASO) cube as a source cube.
- The EPM Integration Agent is available for Profitability and Cost Management customers.

Copyright © 2017, 2026, Oracle and/or its affiliates.

3

Launching Data Integration

To launch Data Integration:

1. From the Home page, click (**Application** cluster).
2. Click (**Data Exchange** icon), and then select the **Data Integration** tab.

   Optionally, you can launch Data Integration by clicking , and then from **Application**, click **Data Exchange**.

   To return to the Home page when you've navigated away from it, click (**Home**) icon.

Viewing the Data Integration Home Page

Service administrators see all integrations on the Data Integration home page. When location security is enabled, Power users see only integrations assigned to the locations to which they have been granted. Otherwise, Power user can view all integrations.

Features include:

Table 3-1 Data Integration Home Page Features

Feature	Description
Search	Search the list of integrations based on selected text in any of the fields. For example, if you type File, the system returns all integrations when the name File appears in the Name, Location, Source, or Target fields.
⋯ before the **Actions** menu > **Create Integration**	Add a new integration.
⋯ before the **Actions** menu > **Create Pipeline**	Add a new pipeline.
⋯ before the **Actions** menu > **User Settings**	Update or clear user-level profiles that apply to a specific user.
Refresh	Refresh the home page.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 3-1 (Cont.) Data Integration Home Page Features

Feature	Description
Actions menu	Provides quick access to key Data Integration functions: • **Process Details**—View the status and logs of a submitted integration or Pipeline. Additionally, you can reset the status of an executed job on this page. See Viewing Process Details • **File Browser**—Launch the File Browser from which you can navigate to the `inbox`, `outbox`, and select files. See Using the File Browser. • **Setup**—Common setup tasks including: – **Applications**—Gateway to integrate data between source systems and target applications, and apply business rules. – **Locations**—View all locations used in your integrations and Pipelines. – **Period Mapping**—Define and manage various kinds of calendars (for example, monthly, weekly, or daily) based on your business and statutory requirements. – **Category Mapping**—Define and manage category mappings, which enables you to categorize and map source system data to a target Scenario dimension member. – **Query**—Create and save a SQL query definition. • **Configure**—Commons configuration tasks: – **System Settings**—Use system settings to set, update, or clear system level options that apply to the entire system. See System Settings. – **Security Settings**—Set up location security and role security for report group options. For more information. See Setting Security Options. – **Agent**—Extract data and metadata from your on-premises data source and then load the data directly to the Oracle Fusion Cloud EPM. See EPM Integration Agent. – **Download Agent**. See Downloading the EPMAgent ZIP. • **Other**: – **Report Execution** - View prebuilt reports that capture business-critical operations and revenue-generating activities within your organization. These reports provide key information on how data is integrated from the Enterprise Resource Planning (ERP) source system into the Oracle Fusion Cloud Enterprise Performance Management target application. – **System Maintenance Tasks** - Run system processes to maintain and clean up all runtime artifacts, such as the Process tables, Staging tables or `inbox`/`outbox` folders.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 3-1 (Cont.) Data Integration Home Page Features

Feature	Description
Data Integration detail	Show the detail of each integration including: • —Integration was executed successfully. • —Integration execution failed. • —New Integration • —Integration was processed with warnings. **Name** **Location** **Source** **Target** **Last Executed** — Go to a workflow process including: • General • Map Dimensions • Map Members • Options • Copy • Process Details • Workbench • Delete Integration When you select an integration to be deleted, a Delete Confirmation page displays with the following message: "Deleting the integration `<integration name>` will delete all related artifacts like dimension mapping, member mapping and any data loaded using the integration. Do you want to proceed?" **Run Integration/Pipeline**

Working with the Data Integration Workflow

Shows Data Integration workflow tasks.

The Data Integration workflow bar consists of the following tasks:

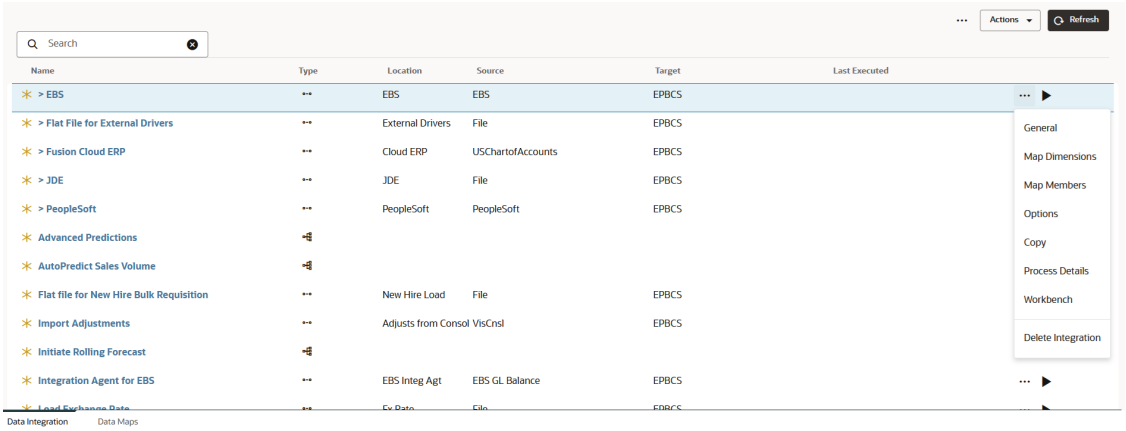
You can navigate between tasks on the workflow bar simply by clicking a task.

Task	Description	More Information
General	Add or edit an integration for file-based and direct integration sources.	Defining a Data Integration

Copyright © 2017, 2026, Oracle and/or its affiliates.

Task	Description	More Information
Map Dimensions	Map the columns in the data source to dimensions in the target application.	Mapping Dimensions
Map Members	Map dimensions to identify how source field values translate to target dimension members.	Mapping Members
Options	Define options for importing and exporting data. Also, define any source filters and business rules.	Setting Data Integration Options

You can also access all tasks on the Data Integration home page by clicking the **Actions** drop-down (•••) to the right of an integration or pipeline and selecting the action. The following shows all the actions associated with an integration job.



To run an integration to extract data from the source and load it to target based on general and filter criteria, select the integration, and then click

.

To run a Pipeline and execute the series of jobs in the stages of the Pipeline, perform the integration process, send emails, and attach logs, select the Pipeline, and then click

.

For more information, see Running an Integration.

Copyright © 2017, 2026, Oracle and/or its affiliates.

4

Integration Tasks

Integration tasks include:

- System Settings
- Setting Security Options
  - Assigning Role Access Security to Report Groups
  - Role Level Security
  - Defining Location Security
- Setting User Level Profiles
- Viewing Locations
- Bulk Managing POV Lock Statuses
- Copying an Integration

System Settings

Use system settings to set, update, or clear system level options that apply to the entire system.

To define system settings:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Configure** actions drop-down, select **System Settings**.
2. From the **System Settings** page, select the option to add or change and then click **Save**.

Available system-level options:

Table 4-1 System-Level Settings

Option	Value
Create Location Folder	Instructs the system to create a location folder in the `inbox` when a location is created. Available values are **Yes** or **No**. Set this option once and do not change it. This setting is optional but recommended.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 4-1 (Cont.) System-Level Settings

Option	Value
Archive Mode	Specifies whether archived files are copied or moved to the archive location. Enter **Copy**, **Move**, or **None**. If you select **Copy**, then the file is left in the `inbox`. If you select **Move**, then the file is copied to the archive folder and deleted from the `inbox`. The folder named `data` is the archive folder. When the file is moved to the archive location, it is renamed as follows: `<Process ID><Year><Month><Day>.<Original Extension>` For example, if the source file name is `BigFile.csv`, and it was loaded for period Mar-07 with a period key of 03/01/2023, and if the process id was 983, then the resulting file name is `98320230301.csv`. Files that you create or upload to an environment are deleted after 60 days. Data Integration process log files are retained for seven days only. Oracle recommends that you regularly download the files you want to keep to a local machine.
Excluded File Upload Wild Cards	Specify file extensions that cannot be uploaded. For example, you might specify ***.jsp** or **,*.py**. Enter ***.*** to disallow all file uploads.
Batch Size	Specify the number of rows read at a time from the file to memory. This parameter is mainly used for performance. When data is loaded, this setting determines how many records are stored in the cache. For example, when 1000 is specified; the system stores 1,000 records in cache. Similarly, when 5000 is specified, the system stores 5,000 records in cache and commit. Determine this setting by Server Memory and adjust as needed.
File Character Set	Select the method for mapping bit combinations to characters for creating, storing, and displaying text from the drop-down. Each encoding has a name; for example, UTF-8. Within an encoding, each character maps to a specific bit combination; for example, in UTF-8, uppercase A maps to HEX41. Specify the method for mapping bit combinations to characters for creating, storing, and displaying text. Each encoding has a name; for example, UTF-8. Within an encoding, each character maps to a specific bit combination; for example, in UTF-8, uppercase A maps to HEX41. Encoding refers to mapping *bit* combinations to characters for creating, storing, and displaying text. Convert the encoding to UNICODE if your source file is not in one of the supported formats.
Default POV Location	Specify the default POV location from the drop-down.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 4-1 (Cont.) System-Level Settings

Option	Value
Default POV Period	Specify the default POV Period from the drop-down. These preferences take precedence when no equivalent settings have been defined in Application Settings or User Settings (for more information, see Setting Application-Level Profiles and Setting User Level Profiles in *Administering Data Management for Oracle Enterprise Performance Management Cloud*.)
Default POV Category	Specify the default POV Category from the drop-down. These preferences take precedence when no equivalent settings have been defined in Application Settings or User Settings.
Global POV Mode	When this value is set to **Yes**, other POVs (Application Level and User Level POV settings) are ignored.
Default Check Report	Specify the type of Check Report to use as the def default check report from the drop-down. The following are pre-seeded check reports, but you can create a new one and specify it here: • Check Report—displays the results of the validation rules for the current location (pass or fail status). • Check Report Period Range (Cat, Start per, End per)—Displays the results of the validation rules for a category and selected periods. • Check Report by Val. Entity Seq.—Displays the results of the validation rules for the current location (pass or fail status); sorted by the sequence defined in the validation entity group. • Check Report with Warnings—Displays the results of the validation rules for the current location. Warnings are recorded in validation rules and shown if warning criteria are met. This report does not show rules that passed the validation.
Batch Timeout in Minutes	When a batch job is run in sync mode (immediate processing), specify the maximum time the job can run. In sync mode, Data Integration waits for the job to complete before returning control.
Log Level	Specify the level of detail displayed in the logs. A log level of **1** shows the least detail. A log level of **5** shows the most detail. Logs are displayed in Process Details by selecting the **Log** link.
Check Report Precision	Specify the total number of decimal digits for rounding numbers, where the most important digit is the left-most nonzero digit, and the least important digit is the right-most known digit.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 4-1 (Cont.) System-Level Settings

Option	Value
Display Data Export Option "Override All Data"	Specify **Yes** to display the **Override All Data** option in the Export Mode drop-down located on the Execute Rule screen. When you select to override all data, the following message is displayed "Warning: Override All Data option will clear data for the entire application. This is not limited to the current Point of View. Do really want to perform this action."
Enable Map Audit	Set to **Yes** to create audit records for the Map Monitor reports (Map Monitor for Location, and Map Monitor for User). The default value for this setting is **No**.
Access to Open Source Document	When drilling down to the Data Integration landing page, this setting determines access to the Open Source Document link (which opens the entire file that was used to load data). Available values: • Administrator—Access to Open Source Document link is restricted to the administrator user. • All Users—Access to the Open Source Document link is available to all users. All Users is the default setting.
Map Export Delimiter	Specify the column delimiter value when exporting member mappings. Available delimiters are: • ! (exclamation mark) • , (comma) • ; (semi-colon) • | (pipe)
Map Export Excel File Format	Select the Excel file format to use when exporting member mappings from the drop-down: Available values: • Excel 97-2003 Workbook (*.xls) • Excel Macro-Enabled Workbook (*.xlsm)
Map LCM Format	Select the export option for data load mappings to Migration (Lifecycle Management) from the drop-down Available value: • Individual Data Load Mapping—maps are exported individually for each location and imported individually for each location. If you have a large number of maps for each location, use this method since it allows loading of maps in smaller sets by location. This method is also useful to migrate maps for certain locations selectively. With this method, existing maps are deleted and replaced by new maps in the snapshot. • Combine Data Load Mapping for All Locations—maps are exported in a single artifact for all locations. With this method, maps from the snapshot are merged to the existing maps in the target system. This is the default setting.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 4-1 (Cont.) System-Level Settings

Option	Value
LCM Mode	Select the LCM Mode to use when exporting Lifecycle Management snapshots from the drop-down. Available options: • **Individual Artifact – Setup Only**—In this mode, you select the individual artifacts to migrate such as locations and mappings. When you import the snapshot in the target system merges data with the existing data in the target system. This mode does not include Workbench data. This is default mode used by the LCM process. Snapshots of individual artifacts are exported as XML files contained in a ZIP file. Snapshot ZIP files are available from the Snapshots tab in Migration. • **Single Snapshot – Setup Only**—In this mode, when you import the snapshot in the target system, all the existing setup data is deleted and the data from the snapshot is imported. For a list of setup artifacts, see Snapshot Export Setup Artifacts in *Administering Data Management for Oracle Enterprise Performance Management Cloud*. Setup artifacts are exported to CSV files in table format contained in a ZIP file. Snapshot ZIP files are available from the Snapshots tab in Migration. Snapshots run in Single Snapshot – Setup Only mode are executed faster than snapshots run in Individual Artifact – Setup Only mode and may prevent database timeout issues. • **Single Snapshot – Setup and Data**—In this mode, the data in the target system is completely deleted and data from the snapshot is imported. The import process can take longer depending on the volume of the Workbench data and may affect the performance of the LCM backup depending on the size of the data in the staging tables. For a list of setup and data artifacts, see Snapshot Export Setup Artifacts in *Administering Data Management for Oracle Enterprise Performance Management Cloud* and Snapshot Export Data Artifacts in *Administering Data Management for Oracle Enterprise Performance Management Cloud*. Before using this mode, create a baseline individual artifact snapshot. The system exports setup and all data (including Workbench) incrementally to CSV files in table format contained in a ZIP file. Snapshot ZIP files are available from the Snapshots tab in Migration.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 4-1 (Cont.) System-Level Settings

Option	Value
Record Limit for Database Batch Processing	The load process uses SQL batch processing to insert and delete rows when the number of records exceeds a 3 million number or records limit. You can adjust the record limit to a slightly higher limit in this setting if greater than the 3 million. **Note** Increasing the Record Limit for Database Batch Processing value to greater than 5 million records may cause a system resource issue so it is recommended not to set the value very high.
Object Storage Namespace	Unique, system-generated top-level container for all buckets and objects in the specific OCI tenancy.
Object Storage Bucket	Container for storing objects in a compartment within a namespace.
All Data Type Load Using Jobs	Specify **Yes** to create a Planning job when an integration is an "All Data Type Load" type load.
Enable Pipeline Execution for Non-Admin Users	Specify **Yes** to enable non-administrator users to view Pipelines on the Data Integration home page.
Use EPM Shared Connection Pool	

Setting Security Options

Set up location security and role security for report group options.

Service administrators can enable and disable Location Security. In addition, you can assign role access to various report groups.

Assigning Role Access Security to Report Groups

Shows how to assign role level security to report groups.

Data Integration security enables service administrators and power users to customize user access to report groups using the concept of roles. Roles are permissions that grant user access to reports. In Data Integration, default roles are assigned to functions that aggregate and tailor specific requirements. After the reports are assigned to a role, the corresponding role is mapped to users when provisioning users in Access Control. The process of granting roles to users is described in Managing User Group Memberships in *Administering Access Control* .

To add role level security to report groups:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Configure** actions drop-down, select **Security Setting**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

2. On the **Security Setting** page, select the **Report Group** function and the role to assign to the report group.

   Available roles:

   - **Data Integration - Administrator**—Performs all functional activities in Data Integration. Users with this role can create, perform, and run:
     - Integrations between source and target systems
     - Pipeline activities
     - Extraction and transformation of data and metadata from on-premises sources using the EPM Integration Agent

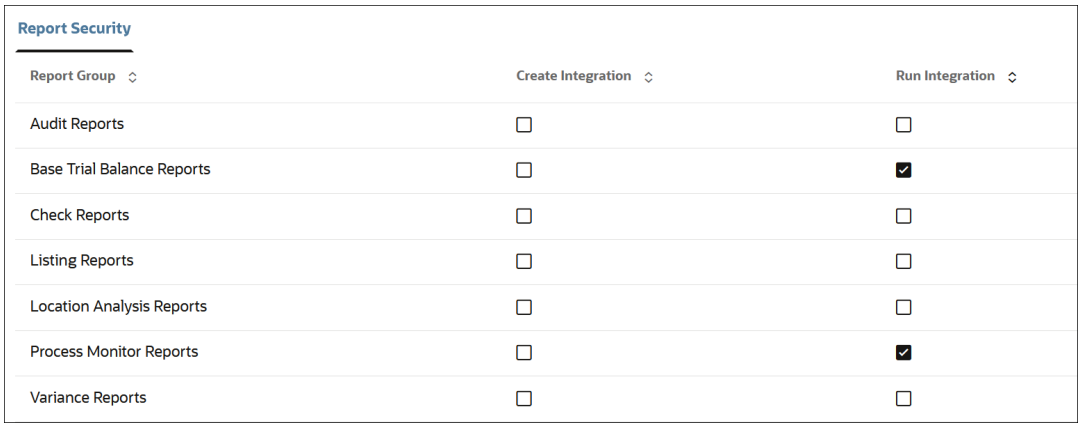
     Included in this application role is the Service Administrator
   - **Data Integration - Create**—Uses Data Integration to create mappings to integrate data between source and target systems. This role can define data rules with various run time options.

     Included in this application role is the Power User.
   - **Data Integration - Dill Through**—Drills through to the source system of the data.

     Included in this application role are the Power User and User.
   - **Data Integration - Run Integration**—Users with this role use Data Integration to run an integration between the source and target.

     If this is the only role assigned to the user, and if they are not a Power User or Service Administrator, they can also view the integration details, but are not able to make any changes.

     Included in this application role is the Power User.



3. Click **Save**.

Role Level Security

Shows how to add role level security.

User access to user interface functions in Data Integration can be customized using role-level security. Roles are permissions that grant user access to functionsData Integration ships with two key roles: "Integration - Create" and "Integration – Run." These roles provide a fixed set of functions that can be performed by a user. Service administrators can add or remove functions available to a role as needed. For example, a service administrator can add the Edit Mapping

Copyright © 2017, 2026, Oracle and/or its affiliates.

function to the "Integration – Run." role or remove Period Mapping and Category Mapping from the "Integration - Create" role. After the functions are assigned to a role, the corresponding role is mapped to users when provisioning users in Application Management. The process of granting roles to users is described in the Managing Granular Roles with Access Control in *Administering Access Control* .

To add role level security:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Configure** actions drop-down, select **Security Setting**.
2. On the **Security Setting** page,
3. From the **Function Security**, select the function (role category) to which to assign access.
4. In **Select**, select the role to assign to the function.

   A list of roles is described below.

Table 4-2 Role and Descriptions

Role	Description
Data Integration - Administrator	Performs all functional activities in Data Integration. Users with this role can create, perform, and run: • Integrations between source and target systems • Pipeline activities • Extraction and transformation of data and metadata from on-premises sources using the EPM Integration Agent Included in this application role is the Service Administrator
Data Integration - Create	Uses Data Integration to create mappings to integrate data between source and target systems. This role can define data rules with various run time options. Included in this application role is the Power User.
Data Integration - Dill Through	Drills through to the source system of the data. Included in this application role are the Power User and User.
Data Integration - Run Integration	Users with this role use Data Integration to run an integration between the source and target. If this is the only role assigned to the user, and if they are not a Power User or Service Administrator, they can also view the integration details, but are not able to make any changes. Included in this application role is the Power User.

5. Click **Save**.
6. **Optional**: To restore role settings, click **Restore Defaults**.

Defining Location Security

User access to locations is determined by location security by enabling you to select the user groups to create for each location.

Location Security is a framework in which service administrators can secure user access to features by location. Data Integration configures and enforces Location Security in Security Settings. You define the user groups to create for each location. When a location is created or updated, then you can create as many groups as defined in the system settings for the location. Additionally, a **Maintain User Groups** option enables you to create user groups in mass for all the existing locations.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Several dependent processes must occur before Location Security is fully implemented:

1. When a location is created, user groups are created automatically in Access Control.

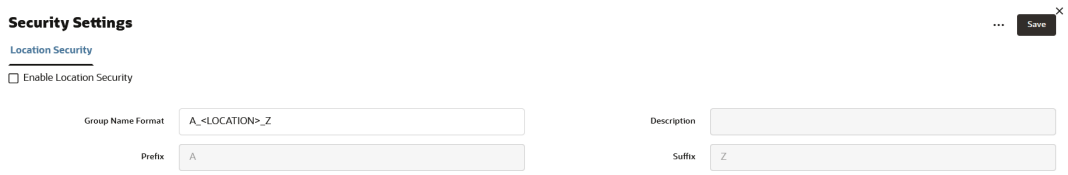
   The user group contains the name of the location and additional prefix and suffix information based on the user preference.

   In addition, roles are provisioned for User Groups by a service administrator. For more information, see: Managing User Group Memberships in *Administering Access Control* .
2. When the user logs in, Data Integration determines the groups assigned to the user.
3. Based on the name of the group, Data Integration determines the accessible locations.
4. The POV region filters the locations based on the user access.

> **Note**
>
> If the web services are used, then location security is still maintained and enforced.

To enable location security:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Configure** actions drop-down, select **Security Settings**.
2. In the **Location Security** section, select **Enable Location Security**.



3. In **Group Name Format**, specify the group name format for the location.

   When the group is saved, the Group name is in the form of Prefix_Location_Suffix. For example, *DM_<LOCATION>_ROLE*.

   The prefix and suffix help identify groups in Access Control.

   > **Note**
   >
   > Do not create a location name that matches with a name in a Security group in Access Control or vice versa. If you have a Security Group with same name as the location, users under the specific group will have access to the integrations associated with that location.
4. In **Description**, enter a description of the user group.

   For example, enter: `Group for Creating and Running Integration`.
5. In the **Prefix** field, enter the prefix to identify the user group name, for example, **DI**.

   When the group is saved, the prefix is prepended to the group name.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> Underscore is not supported in the prefix or suffix for group names.

6. In the **Suffix** field, select the name of the function or rule that the user is provisioned to access.

> **Note**
>
> Underscore is not supported in the prefix or suffix for group names.

For example, you could specify that the location security for the group is for a :

- Run integration role
- Create integration role

When the group is saved, the suffix is appended to the user group name.

7. Click **Save**.
8. To create user groups in mass for the location, click ••• and select **Maintain User Group**.

To disable location security:

1. From the **Actions** menu, then from the **Configure** actions drop-down, select **Security Settings**
2. From the **Location Security** section, deselect **Enable Location Security**.
3. When security by location is disabled, this message is displayed: `Security by Location is disabled. Would you like to enable the feature?`
4. Click **Save**.

Setting User Level Profiles

User-level settings enable you to update or clear user-level profiles that apply to a specific user.

> **Note**
>
> When the Global mode is defined, then user level-profiles for the POV are not applicable.

To set a user-level profile:

1. On the Data Integration homepage, click ••• and select **User Settings**.
2. On the **User Settings** page, select the options to add or modify.
3. Click **Save**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 4-3 User Level Settings

Option	Description
File Character Set	Specify the method for mapping *bit* combinations to characters for creating, storing, and displaying text. Each encoding has a name; for example, UTF-8. Within an encoding, each character maps to a specific *bit* combination; for example, in UTF-8, uppercase A maps to HEX41. Encoding refers to mapping *bit* combinations to characters for creating, storing, and displaying text. You should convert the encoding to UNICODE if your source file is not in one of the supported formats.
Default POV Location	Specify the default POV location.
Default POV Period	Specify the default POV Period.
Default POV Category	Specify the default POV Category.
Default Check Report	Specify the type of Check Report to use as the default check report at the user level. The following are pre-seeded check reports, but you can create a new one and specify it here: • Check Report—Displays the results of the validation rules for the current location (pass or fail status). • Check Report Period Range (Cat, Start per, End per)—Displays the results of the validation rules for a category and selected periods. • Check Report by Val. Entity Seq.—Displays the results of the validation rules for the current location (pass or fail status); sorted by the sequence defined in the validation entity group. • Check Report with Warnings—Displays the results of the validation rules for the current location. Warnings are recorded in validation rules and shown if warning criteria are met. This does not show rules that passed the validation.
Log Level	Specify the level of detail displayed in the logs. A log level of **1** shows the least detail. A log level of **5** shows the most detail. Logs are displayed in Process Details by selecting the **Log** link.
Map Export Delimiter	Sets the column delimiter value when exporting member mappings. Available delimiters are: • ! (exclamation mark) • , (comma) • ; (semi-colon) • | (pipe)
Map Export Excel File Format	Select the Excel file format to use when exporting member mappings: • Excel 97-2003 Workbook (*.xls) • Excel Macro-Enabled Workbook (*.xlsm)
Drill Export File Format	Specify the file format to use when exporting the drill file. Available options: • Comma Separated Values (*.csv) • Excel Workbook (*.xlsx)
Object Storage Namespace	Unique, system-generated top-level container for all buckets and objects in the specific OCI tenancy.
Object Storage Bucket	Container for storing objects in a compartment within a namespace.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> If you specify the user settings, then the user settings takes preference over system settings.

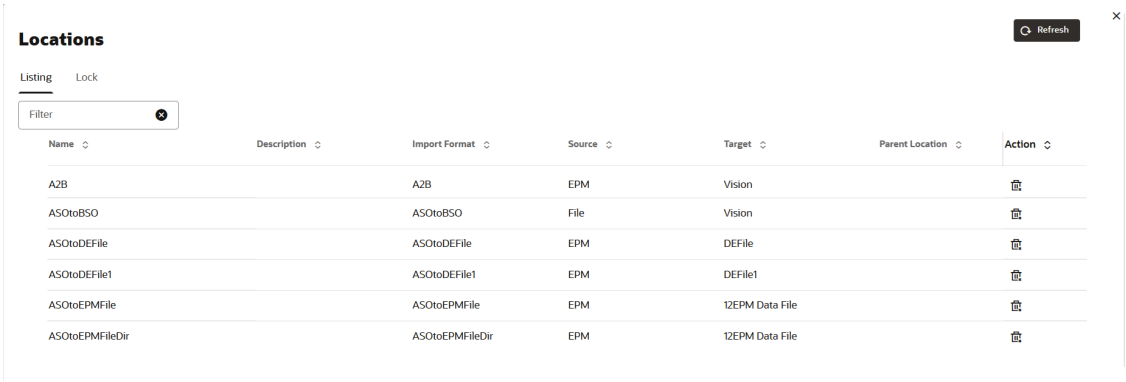
Viewing Locations

Shows how to view all locations used in your integrations and Pipelines.

Use the Locations page to view all locations used in your integrations and Pipelines. You can filter the locations displayed and optionally delete any used locations.

A location is the level at which an integration and Pipeline is executed in Data Integration. You define locations to specify where to load the data. Additionally, Locations enable you to use the same import format for more than one target application where the dimensionality of the target applications is the same.

To view all locations, from the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Locations**.



To filter the display of location by selected criteria, in **Filter**, specify a string, text, or characters to filter on.

To delete a location, select the location and click . The system prompts you to confirm the deletion. When a location is deleted, the location is removed from all other Data Integration pages where the location is included. You cannot delete location used in any integration.

Bulk Managing POV Lock Statuses

Shows how to select a subset of POVs and then lock or unlock them.

You can select a subset of POVs (a POV displays a particular version of an application for a selected snapshot, such as year, period, and scenario) and then lock or unlock them as needed.

You can also filter a list of locked/unlocked POVs, and then perform either a selected action or multiple actions at the same on the filtered set of POVs.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Locking POV Locations

Locking a POV prevents data from being loaded to a selected POV by locking all locations related to the current period and category for an entire target application. When a location has been locked, you cannot import, validate, export, or re-run the validation.

To lock POV locations:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Locations**.
2. From the **Locations** page, select the **Lock** tab.
3. From **Application** drop-down, select the application associated with the POV to lock.
4. **Optional:** From **Period** drop-down, select the period to lock.
5. **Optional:** From the **Category** drop-down, select the category to lock.

   For example, select **Actual** or **Budget**.
6. Click to select the application to lock.

> **Note**
>
> In order for the POV to show in the detail section, data must be loaded to it.

7. Click to lock the POV.

   A Confirmation message shows the application name that has been locked successfully.

UnLocking POV Locations

1. From the **Actions** menu, then from the **Setup** actions drop-down, select **Locations**.
2. From the **Locations** page, select the **Lock** tab.
3. From **Application** drop-down, select the application associated with the POV to unlock.

> **Note**
>
> In order for the POV to show in the detail section, data must be loaded to it.

4. **Optional:** From **Period** drop-down, select the period to unlock.
5. **Optional:** From the **Category** drop-down, select the category to unlock.

   For example, select **Actual** or **Budget**.
6. Click to select the application to unlock.
7. Click to unlock the POV.

   A Confirmation message shows the application name that has been unlocked successfully.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Copying an Integration

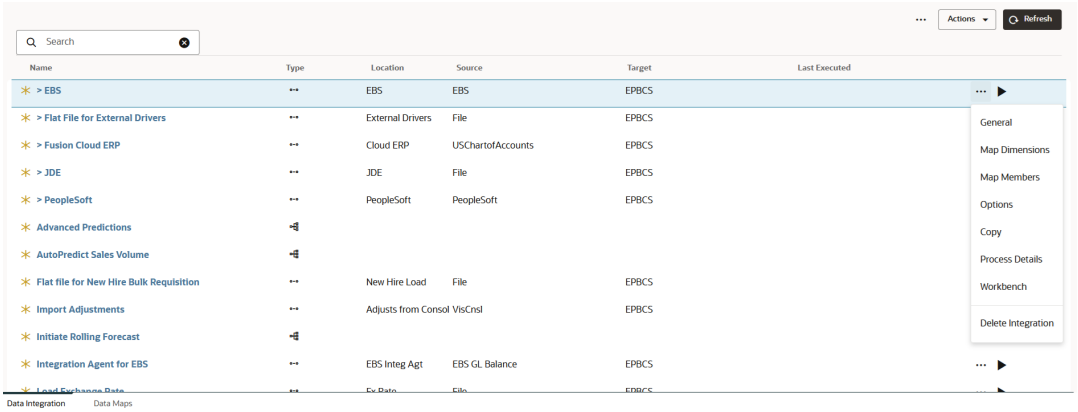
Shows how to copy an integration using a different integration name.

The Copy Integration feature enables you to create an integration identical to another integration but with a different name. You can reuse the existing location or create a new location, reuse the import format or create a new import format, and optionally select to use a local or remote connection.

> **Note**
>
> The Copy integration features is only available for administrators.

To copy an integration:

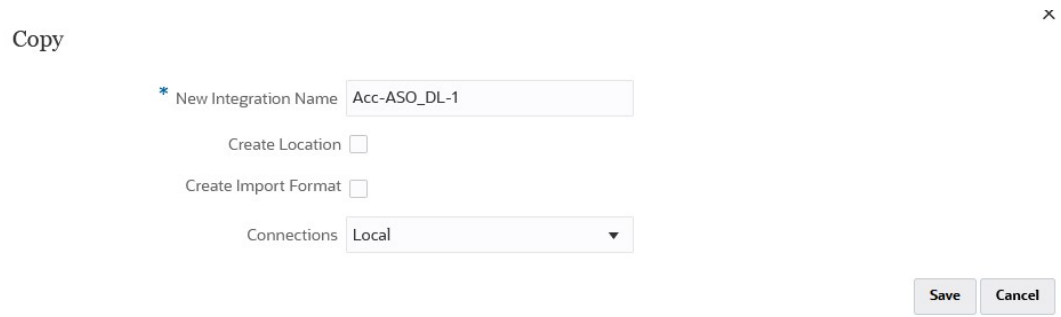
1. From the Data Integration home page, click **...** next to the integration to copy and then from the **Actions** drop-down, select **Copy**.



2. From the **Copy** page, then in **New Integration Name**, specify the new integration name.

   The system defaults the new integration name to the source integration name and adds the suffix "-1." For example, if the name of the source integration is "Monthly Process", then the seeded new name is "Monthly Process-1."

   If the new integration name already exists, the system generates an error when you attempt to save the new integration. In this case, change the name of the new integration to a unique name.



Copyright © 2017, 2026, Oracle and/or its affiliates.

3. **Optional**: Enable **Create Location** to create a new location for the copied integration and copy the integration mappings from the source location.

   By default, the system copies the new integration name for the location name. For example, if the new integration name is **Acc-ASO_DL-1**, then the location name for the new integration is also **Acc-ASO_DL-1**.

   If you do not select to create a new location, the existing location name is used in the copied integration definition.

4. **Optional**: Enable **Create Import Format** to create a new import format for the copied integration and copy the integration mappings from the source import format.

   By default, the system copies the new integration name for the import format name. For example, if the new integration name is **Acc-ASO_DL-1**, then the import format name for the new integration is also **Acc-ASO_DL-1**.

   A copied import format name cannot exceed 20 characters.

   If you do not select to create a new import format, the existing import format name is used in the copied integration definition.

   If a new import format is enabled, then a new location is enabled automatically.

5. From **Connections**, select the connection to use when copying the integration.

   Select **Local** to copy the integration in the same instance. You can also select the connection of another instance to copy the integration to the other instance. The connections must be defined in Oracle Fusion Cloud EPM Connections. For more information, see About Connecting Cloud EPM Environments in *Administering Planning* and Creating, Editing, and Deleting Connections to Other Cloud EPM Environments in *Administering Planning*.

   Select **Remote** to copy an integration to a remote environment. When you select a remote connection, the remote environment must be identical to the current environment including the: source and target applications, user-defined dimension assignments in the target application, logic group, and check rules.

6. Click **Save**.

   The following example shows the source and copied integrations on the Data Integration home page

Copyright © 2017, 2026, Oracle and/or its affiliates.

5

Registering Applications

Shows how to register source and target applications in Data Integration.

Use the Applications option to register source and target applications when defining integrations. You can register local Oracle Fusion Cloud Enterprise Performance Management applications, business to business process integrations (cloud to cloud applications, such as Planning to Financial Reporting) data export to file applications, and generic data source entities into your existing EPM portfolio.

The process to register applications provides data visibility, integrity, and verification systems. After you select the Applications option, an Applications page displays allowing you to register new applications, or you can select an existing application and update application options, delete it, or refresh application definitions.

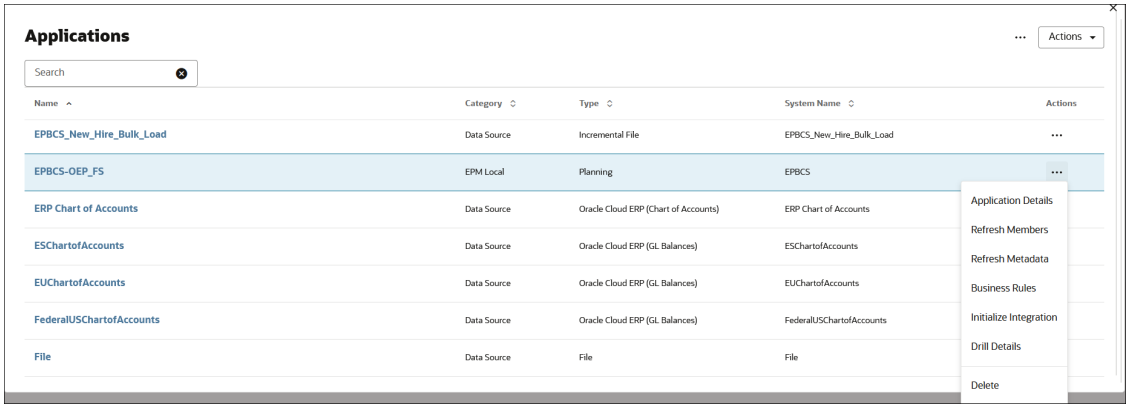
Applications registered in Data Integration are made immediately available for selection when you choose a source and target on the General page. They are also made available in Data Management.

> **Note**
>
> The Applications option is available in the Actions drop-down on the Data Integration home page for administrators only.

You can use the Actions drop-down next to each row of applications to select different application related tasks associated with the application type including:

- viewing dimensions
- change applications options
- adding, updating, or deleting values for data source applications.
- updating (reinitialize) Oracle NetSuite and Oracle ERP Cloud applications.

The following example shows the application options available for an EPBCS application from the **Actions** drop-down:



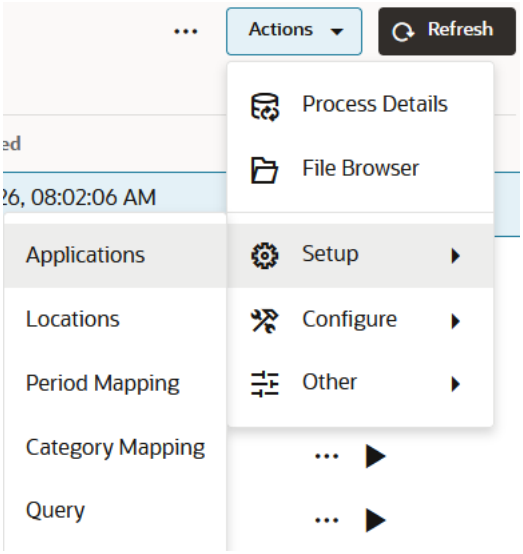
Copyright © 2017, 2026, Oracle and/or its affiliates.

Launching Applications Options

Shows how to launch the Applications component in Data Integration.

> **Note**
>
> Administrators only can register and modify applications.

To launch Application options, from the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.



Navigating Applications

Shows how to navigate the features on the Applications page.

Service administrators can view all registered applications on the Applications page by:

- Name
- Category
- Type
- System Name

You can reorder each display item above alphabetically by clicking to the right of the display item.

Features include:

Feature	Description
••• before the **Actions** menu > **Create Application**	Create a new application.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Feature	Description
Name	In the Oracle Fusion Cloud Enterprise Performance Management, when you register all input cube applications, this field shows the service application name, such as Planning or Financial Consolidation and Close, with any unique prefix. For Cloud EPM, when you register individual cubes, this field shows the application name, cube name with any prefix. For an Oracle NetSuite application, this field show the user-friendly search name of the application. For all other applications, this field shows the name of the application with any prefix.
Category	Shows the class or division to which an application belongs. Available application categories: • EPM Local, which includes: – Input Cubes (BSO) – Reporting Cubes (ASO/Essbase) • Cloud EPM, which includes: – Input Cubes – Reporting Cubes • Data Source, which include: – Data Source – Oracle Financials Cloud, Oracle Human Capital Management Cloud, Oracle NetSuite and incremental files. – SQL Query data sources • Data Export, which includes an application type of Data Export to File • Dimension (for loading metadata)

Copyright © 2017, 2026, Oracle and/or its affiliates.

Feature	Description
Type	Shows the type of the registered application: • EPM Local and Cloud EPM application • Oracle ERP Cloud • Oracle HCM Cloud • Oracle Netsuite • Oracle ERP Cloud (Receivables Transactions) • On Premise Database • Peoplesoft GL Balance • Oracle ERP Cloud (Trial Balance Average) • Incremental File • EBS GL Balance • Oracle ERP Cloud (Chart of Accounts) • Oracle ERP Cloud (Custom) • Oracle ERP Cloud (Payables Transactions) • Oracle ERP Cloud (Trial Balance) • EPM Dimension • Data Export to File • Data Export to On Premise Database • Oracle Projects (Active Projects) • Oracle Projects (Actual) • Oracle Projects (Budgets) • Oracle Projects (Integration Status) • Oracle Projects (Organization) • Oracle Projects (Templates) • Planning Budget Revision • Planning Projects Data
System Name	In the Cloud EPM, when you register all input cube applications, this field shows the service application name, such as Planning or Financial Consolidation and Close, without any prefix. For EPM, when you register individual cubes, this field shows the application name of the Planning, Financial Consolidation and Close, etc. application. For an Oracle NetSuite application, this field show the internal name of the application. For all other applications, this field shows the name of the application without any prefix.
Actions	An Actions drop-down next to each row of applications enables you to select different application related tasks associated with the application category and type. For EPM local, Cloud EPM, and Dimension applications, the Application Detail option is available. For Data Source and Data Export applications, the Application Detail option, Edit option, and Set Default option are available. You can also delete an application. A Delete Confirmation message is shown with the following: "There may be rules associated with this target application. Deleting this application will delete all its rules. Do you still want to continue?" When you delete a target application, the application is marked for deletion and is unavailable for any metadata or integration processes, including the import format and location creation options. All the existing integrations involving the application are removed.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Registering Cloud EPM Local Applications

Shows how to register an Oracle Fusion Cloud EPM local application in Data Integration.

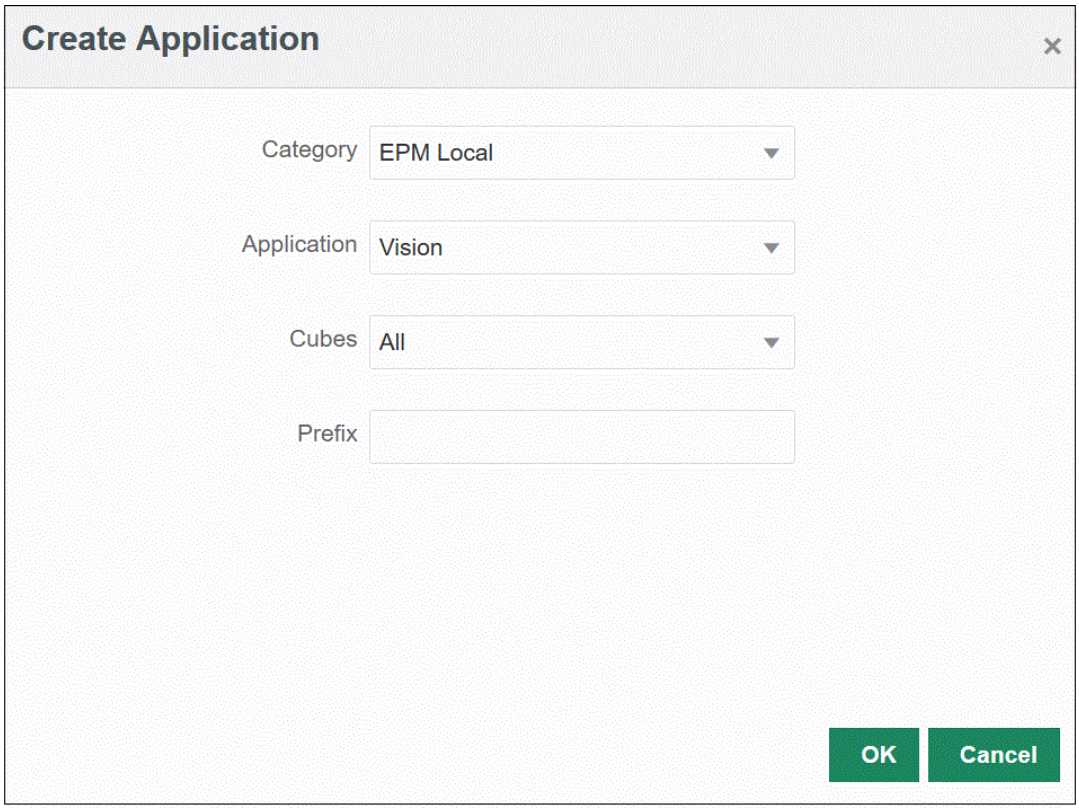
When integrating Oracle General Ledger data from your primary service application, such as Planning, use the EPM Local category to register the application. An EPM local application is for one instance and only one application can exist per instance.

EPM local applications enable you synchronize between data or Reporting Cubes or move data between different Input Cubes.

For example, Financial Consolidation and Close customers can add Planning data, or a Planning customer can add more Planning applications. In addition, this integration enables you to write back from a cloud to an on-premises application or other external reporting applications.

To create an Cloud EPM local application:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, click ••• before the **Actions** menu and then click **Create Application**.
3. On the **Create Application** page, and then **Category**, select **EPM Local**.



4. From **Application**, select the application for the primary service application.
5. From **Cubes**, select the source plan type associated with the application.

Copyright © 2017, 2026, Oracle and/or its affiliates.

The source cube defines the flow of data between the models or plan types of an application.

6. **Optional**: In **Prefix**, specify a prefix to make the application name unique.

   The prefix is concatenated with the application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.

Registering Cloud EPM Applications

Shows how to register an Oracle Fusion Cloud EPM application in Data Integration.

You create and register a Cloud EPM application when you use a business process instance with a remote service, such as Planning, to integrate data with another business process instance, such as Financial Consolidation and Close. In this case, connection information must be selected between the two business process service instances.

This feature enables EPM customers to adapt cloud deployments into their existing EPM portfolio including:

- Planning Modules
- Planning
- FreeForm
- Financial Consolidation and Close
- Tax Reporting
- Account Reconciliation
- Strategic Workforce Planning
- Profitability and Cost Management
- Enterprise Profitability and Cost Management

To create a Cloud EPM application:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, click ••• before the **Actions** menu and then click **Create Application**.
3. On the **Create Application** page, and then **Category**, select **Cloud EPM**.
4. In **URL**, specify the service URL that you use to log on to your service.
5. In **User Name**, specify the user name for the Cloud Service application.

   The user name is the name of the administrator user defined in the Applications page to define the connection. For more information, see Connecting Environments in Cloud EPM in *Administering Planning* .
6. In **Password**, specify the password for the Cloud Service application.
7. In **Domain**, specify the domain name associated with the Cloud Service Application.

   An identity domain controls the accounts of users who need access to service instances. It also controls the features that authorized users can access. A service instance belongs to an identity domain.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> Administrators can update the domain name that is presented to the user, but Data Integration requires the original domain name that was provided when the customer signed up for the service. Alias domain names cannot be used when setting up Oracle Fusion Cloud Enterprise Performance Management connections from Data Integration.

8. From **Application**, select the type of application.
9. Click **OK**.

Registering Data Source Based Applications

Shows how to register a data source application in Data Integration. A data source is an application category which applies to generic a framework of applications that can be used as source data to the Oracle Fusion Cloud Enterprise Performance Management.

Data source is an application category applies to generic framework of applications that can be used as source data to the Cloud EPM, such as Oracle NetSuite or Oracle E-Business Suite (EBS), etc.

Data source-based applications available in Data Integration include:

- Oracle NetSuite—Load data from the following Oracle NetSuite applications:
  - NetSuite EPM Connector SuiteApp
  - NSPB Sync SuiteApp
  - Planning and Budgeting Sync SuiteApp
  - Account Reconciliation Sync SuiteApp
  - Financial Consolidation and Close Sync SuiteApp
- Oracle Cloud ERP—General Ledger. Load Actuals from General Ledger and write-back Budgets and Actual Adjustment Journals
- Oracle Cloud ERP—Budgetary Control. Load commitments, obligations, and expenditures and write-back Budgets
- Oracle Cloud ERP—Sub-Ledgers. Load sub-ledger transaction data.
- Oracle NetSuite—Load actuals and metadata from Oracle NetSuite.
- Oracle Human Capital Management Cloud—Load Employee attribute, salary, and other related Job information.
- E-Business Suite— Load Actuals from General Ledger using the EPM Integration Agent.
- Peoplesoft GL Balance—Load Actuals from General Ledger using the EPM Integration Agent.
- Incremental File—Enables you to compare a source data file with a prior version of the source data file and identify new or changed records and then load only that data set.
- On Premises Database
- On Premises File
- Oracle Autonomous Database
- Oracle Cloud ERP (Chart of Accounts)

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Oracle Cloud ERP (Custom)
- Oracle Cloud ERP (Payables Transactions)
- Oracle Cloud ERP (Receivables Transactions)
- Oracle Cloud ERP (Trial Balances - Averages)
- Oracle Cloud ERP (Trial Balances)
- Oracle Projects (Active Projects)
- Oracle Projects (Actual)
- Oracle Projects (Budgets)
- Oracle Projects (Integration Status)
- Oracle Projects (Organization)
- Oracle Projects (Templates)
- Planning Budget Revision
- Planning Project Integration Status
- Planning Projects Status

Registering Oracle ERP Cloud General Ledger Applications

When integrating Oracle General Ledger data from the Oracle ERP Cloud with your Oracle Fusion Cloud Enterprise Performance Management application, you use Applications to specify the Oracle ERP Cloud General Ledger applications as a data source application type and then import the applications so that they can be used as sources in the Cloud EPM target applications. When you import the applications, the system brings over the source Oracle ERP Cloud General Ledger data into the Cloud EPM system as Oracle Essbase cubes. Each Essbase application represents a Ledger definition from the source Oracle ERP Cloud General Ledger.

> **Note**
>
> Data Integration does not allow you to register a Oracle ERP Cloud application which has the same name as a Cloud EPM application. To avoid any issues, rename the Cloud EPM.

To define an Oracle ERP Cloud application as data sources:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, click ••• before the **Actions** menu and then click **Create Application**.
3. From **Category**, select **Data Source**.
4. From **Type**, select **Oracle Cloud ERP**.
5. From **Connections**, select available Oracle ERP Cloud connection.
6. In **Application Filter**, specify any filters for the load.

   Application filter is used to filter the name of application to import from the Oracle ERP Cloud. For example, you can filter on a specific application name such as Vision USA or wild card Vision%.

Copyright © 2017, 2026, Oracle and/or its affiliates.

7. Select **Budgetory Control** to import Budgetary Control applications into Data Integration by discovering the Budgetary Control balance Essbase cubes from Oracle ERP Cloud.
8. Click **Import Applications** to register the application and initialize the Oracle ERP Cloud application.

> **Note**
>
> In some cases, customers may have multiple instances of their Fusion ERP, for example, one for test, and one for production. When setting up an integration to these sources, if the ledgers are named the same, and are identical in both sources, only a single connection can be used.
>
> For example, when customers set up an integration to the test instance with ledgers named Ledger1 and Ledger2, and now need to pull data from the production instance with Ledger1 and Ledger2, they can just change the connection information for the connection and the integration will use those connection details with the selected integration. Adding a new connection will not be referenced when running the integration – only the original connection, but with the updated connection information.

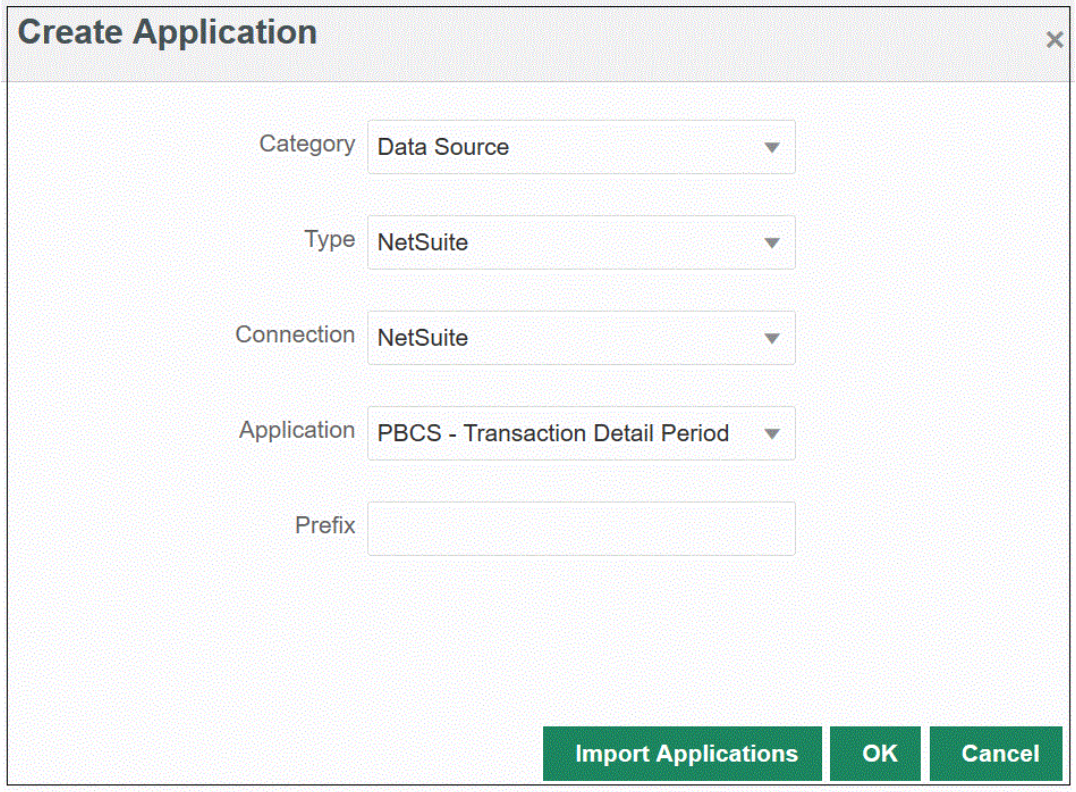
Registering Oracle NetSuite Applications

You register Oracle NetSuite as an application type of "data source" and then import Oracle NetSuite saved searches. During the import, the system loads the Oracle NetSuite saved searched results. After the import, an application record is created for each Oracle NetSuite saved search results under the user name registered in the Oracle NetSuite source system.

To define the Oracle NetSuite connection and import the Oracle NetSuite applications:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, click ••• before the **Actions** menu and then click **Create Application**.
3. From **Category**, select **Data Source**.
4. From **Type**, select **Netsuite**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



5. From **Connections**, select **Netsuite**.

   The connection step provides the connection to Oracle NetSuite and imports the Oracle NetSuite applications. Later if you want to add an application with a prefix, you can select from drop down and add it.

6. From **Application**, select the name of the NSPB Sync SuiteApp saved Search.

   Metadata saved searches includes "Metadata" in the saved search name, and Data saved searches includes "Data" in the saved search name.

7. **Optional**: In **Prefix**, specify a prefix to make the application name unique.

   The prefix is concatenated with the application name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.

8. Click **Import Applications** to register the application and initialize the Oracle NetSuite applications.

   Oracle NetSuite Searches results are registered with an application type of "data source." During the initialization, the system creates the data source automatically based on the available Saved Searches results under the user name registered in the Oracle NetSuite source system.

   NSPB Sync SuiteApp Saved Searches may include:

Saved Search Name	ID	Type
PBCS All Posting Transactions Last Period		Summary
PBCS All Posting Transactions Last Period		Detail

Copyright © 2017, 2026, Oracle and/or its affiliates.

Saved Search Name	ID	Type
PBCS All Posting Transactions Last Period—Trial Balance		Trial Balance
PBCS - Transaction Summary	customsearch_nspbcs_all_transactions_sum	Transaction
PBCS - Transaction Detail	customsearch_nspbcs_all_transactions_det	Transaction
PBCS - Trial Balance	customsearch_nspbcs_trial_balance	Transaction
PBCS sync (Data) Income Statement Transactions [.csv]	customsearch_pbcs_sync_is_csv	Transaction
PBCS sync (Data) Balance Sheet Transactions [.csv]	customsearch_pbcs_sync_bs_csv	Transaction
PBCS sync (Data) Beg Balance [.csv]	customsearch_pbcs_sync_begbal_csv	Transaction
PBCS sync (Data) Income Statement Transactions	customsearch_nspbcs_sync_is	Transaction
PBCS sync (Data) Balance Sheet Transactions	customsearch_nspbcs_sync_bs	Transaction
PBCS sync (Data) Beg Balance	customsearch_nspbcs_sync_begbal	Transaction
PBCS sync (Data) Consolidated Exchange Rates	customsearch_nspbcs_sync_fx	Consolidated Exchange Rates
PBCS sync (Metadata) Income Statement Accounts	customsearch_nspbcs_sync_acct_inc_stmt	Account
PBCS sync (Metadata) Balance Sheet Accounts	customsearch_nspbcs_sync_acct_bs	Account
PBCS sync (Metadata) Class	customsearch_nspbcs_sync_class	Class
PBCS sync (Metadata) Customer	customsearch_nspbcs_sync_cust	Customer
PBCS sync (Metadata) Department	customsearch_nspbcs_sync_dept	Department
PBCS sync (Metadata) Item	customsearch_nspbcs_sync_item	Item
PBCS sync (Metadata) Location	customsearch_nspbcs_sync_loc	Location
PBCS sync (Metadata) Project	customsearch_nspbcs_sync_prj	Project
PBCS sync (Metadata) Customer Project	customsearch_nspbcs_sync_cprj	Project
PBCS sync (Metadata) Subsidiary	customsearch_nspbcs_sync_sub	Subsidiary
PBCS sync (Metadata) Vendor	customsearch_nspbcs_sync_vend	Vendor

9. Click **OK**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Updating An Oracle NetSuite Application

Use the Update Application option to reinitialize the Oracle NetSuite application. In this way, you can extract the most up-to-date accounting reports from your NSPB Sync SuiteApp or EPM Connector SuiteApp Saved Searches and use them in your integration with the Oracle Fusion Cloud Enterprise Performance Management business process.

To update an Oracle NetSuite application:

1. From the **Data Integration** home page, then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, then from the **Actions** menu, select **Update Applications**.
3. On the **Update Applications** page, then **Type**, select Oracle NetSuite.
4. From **aif.application connection** drop-down, select the name of the connection associated with the Oracle NetSuite application(s) to update.
5. In **Prefix**, specify a prefix to make the application name unique.

   The prefix is concatenated with the application name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.
6. Click **Update Applications** to reinitialize the application.

Registering Incremental File Applications

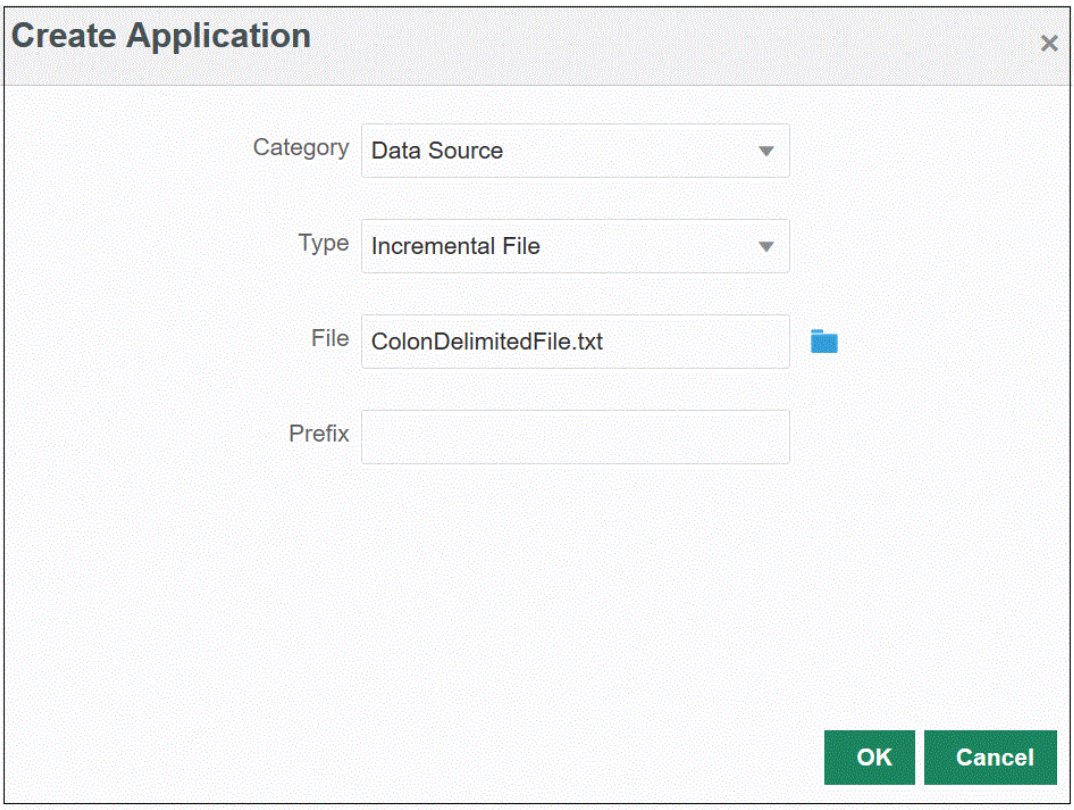
The Incremental File Adapter feature enables you to compare a source data file with a prior version of the source data file and identify new or changed records and then load only that data set.

To use the Incremental File Adapter feature, you register an initial source data file as an incremental file adapter data source. The initial source data file is used as the template. Actual integrations are run from the file designated in the run integration process where a file comparison is run between the initial source data file and a subsequent file. You can load once, twice or many times thereafter. The last run file b[Registering Incremental File Applications]ecomes the basis against which the subsequent load is evaluated. The data source loads only the differences, which leads to a faster load during the file import. The remaining data import processes stay the same as in a standard data load for a file.

To create an incremental file data source:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, click ••• before the **Actions** menu and then click **Create Application**.
3. On the **Create Application** page, and then **Category**, select **Data Source**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



4. From **Type**, select **Incremental File**.
5. From **File**, specify the initial source data file to use as the template.

   Click to search for a file from the **File Browser** page.

   When selecting a file, note the following:

   - The source data file must be a delimited data file.
   - Data files used must contain a one-line header, which describes the delimited columns.
   - Both numeric and non-numeric data can be loaded.
6. **Optional**: In **Prefix**, specify a prefix to make the application name unique.

   The prefix is concatenated with the application name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.
7. Click **OK**.

Defining an On-Premises Data Source Application

You create an on-premises data source application when you need to extract data from your on-premises data sources and then load the data directly to the Oracle Fusion Cloud Enterprise Performance Management using the EPM Integration Agent. The EPM Integration Agent executes a query against an on-premises relational database and then loads the data to the Cloud EPM.

To create and register an on-premise data source application created for an extraction, you must create a file that contains only sample data with one header record. The file cannot not

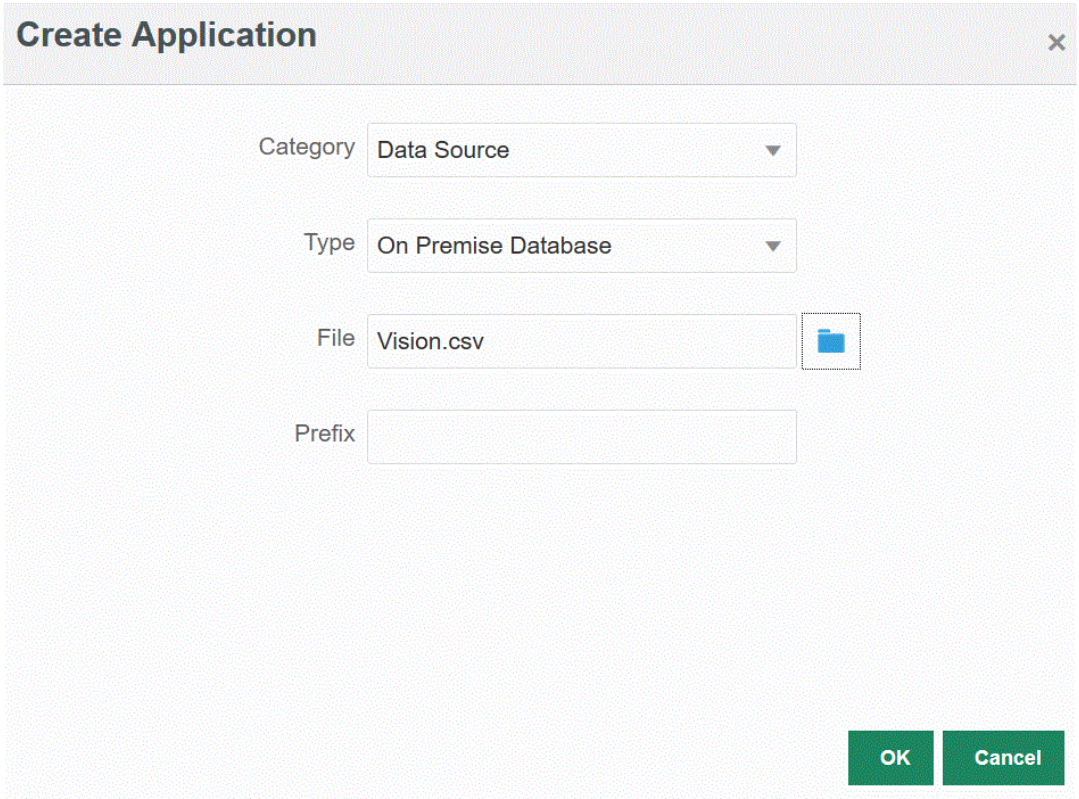
Copyright © 2017, 2026, Oracle and/or its affiliates.

contain the application name or SQL query etc. The name of the file must be the name of the application.

> **Note**
>
> For more information, see Extracting Data from On-premises Data Sources.

To create an on-premises data source application:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, click ••• before the **Actions** menu and then click **Create Application**.
3. On the **Create Application** page, and then **Category**, select **Data Source**.



4. From **Type**, select **On Premise Database**.
5. From **File**, specify the file that contains only sample data with one header record to be used for the on-premises application.

   The header row must exactly match the dimension name in the target dimension. You can use an alias for a column name in the SQL query to the dimension name.

   Click to search for a file from the **File Browser** page.
6. **Optional**: In **Prefix**, specify a prefix to make the application name unique.

Copyright © 2017, 2026, Oracle and/or its affiliates.

The prefix is concatenated with the application name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.

7. Click **OK**.

Registering Oracle Cloud ERP Applications

You can extract non-General ledger data from Oracle Cloud ERP data sources and load it to the Oracle Fusion Cloud Enterprise Performance Management. These types of data sources let you define an Oracle Business Intelligence Publisher data extract that is referenced by Data Integration to pull data directly from Oracle Cloud ERP sources like payables, receivables, fixed assets, and supply chain.

Data Integration provides prepackaged queries to the Oracle Cloud ERP sources including:

- Oracle Cloud ERP (Payables Transactions)
- Oracle Cloud ERP (Receivables Transactions)
- Oracle Cloud ERP (Trial Balance - Average)
- Oracle Cloud ERP (Trial Balance)

In addition, you can create and run a custom query against a BI Publisher data extract. In this case, use an Oracle Cloud ERP (Custom) data source adapter in Data Integration to import the source CSV file from BI Publisher, and then set the report parameters to use.

> **Note**
>
> An Oracle Cloud ERP integration requires that you have the privileges or user role and data access to all ERP ledgers.
>
> You must register source system for the Oracle ERP Cloud data sources using the system type Oracle Cloud ERP and specify your user credentials. This step includes specifying the connection details and testing the connection. For more information, see Configuring an Oracle ERP Cloud Connection.

> **Note**
>
> When configuring connections for Oracle Fusion Cloud EPM and Oracle Cloud ERP , the URLs specified for connections must use the secure HTTP protocol, for example, "https://server."

To register an Oracle Cloud ERP application:

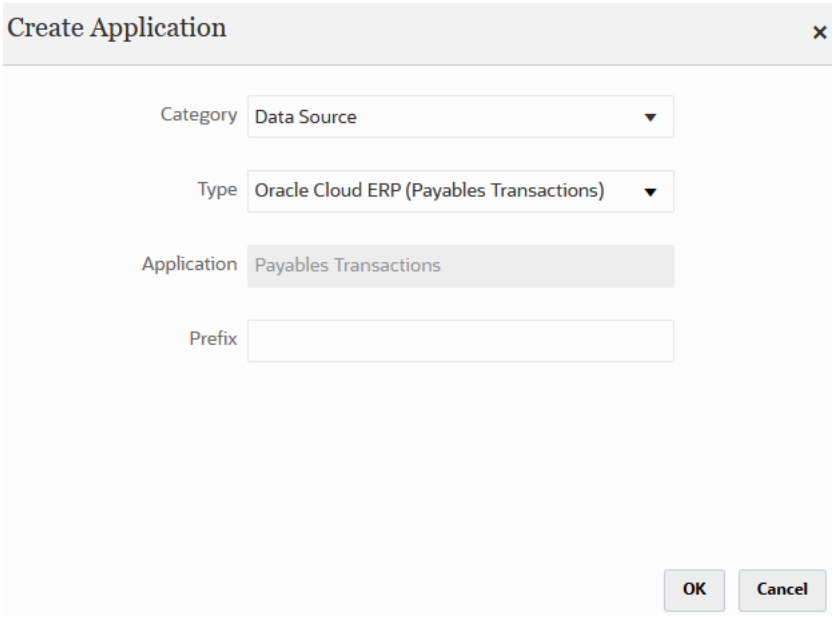
1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, click ••• before the **Actions** menu and then click **Create Application**.
3. From **Category**, select **Data Source**.
4. From **Type**, select the type of prepackaged Oracle ERP Cloud query.

   Available Oracle Cloud ERP types:

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Oracle Cloud ERP (Payables Transactions)
- Oracle Cloud ERP (Receivables Transactions)
- Oracle Cloud ERP (Trial Balance - Average)
- Oracle Cloud ERP (Trial Balance)

The name of the prepackaged application that displays in the **Application** depends on the type of prepackaged Oracle Cloud ERP query selected in this field.



5. **Optional**: In **Prefix**, specify a prefix to make the application name unique.

   The prefix is concatenated with the application name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.

6. Click **OK**.

Updating Oracle Cloud ERP Applications

Use the Update Application option to reinitialize Oracle Cloud ERP applications. When you reinitialize an application, the source system fetches all new additions such as ledgers, chart of accounts, and so on.

To update an application:

1. From the **Data Integration** home page, from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, select an Oracle Cloud ERP application to reinitialize.
3. From the **Actions** menu, select **Update Applications**.
4. On the **Update Applications** page, then **Type**, select Oracle Cloud ERP.
5. From the **aif.application connection** drop-down, select the name of the connection associated with the data source.
6. From **Application Filter**, specify any application filter for the application.

   For more information about application filters, see Applying Application Filters to an Oracle ERP Cloud Data Source.

Copyright © 2017, 2026, Oracle and/or its affiliates.

7. Enable **Budgetary Control** to register the application as a Budgetary Control data source.
8. Click **Update Applications** to reinitialize the application.

Registering Oracle Cloud ERP Custom Applications

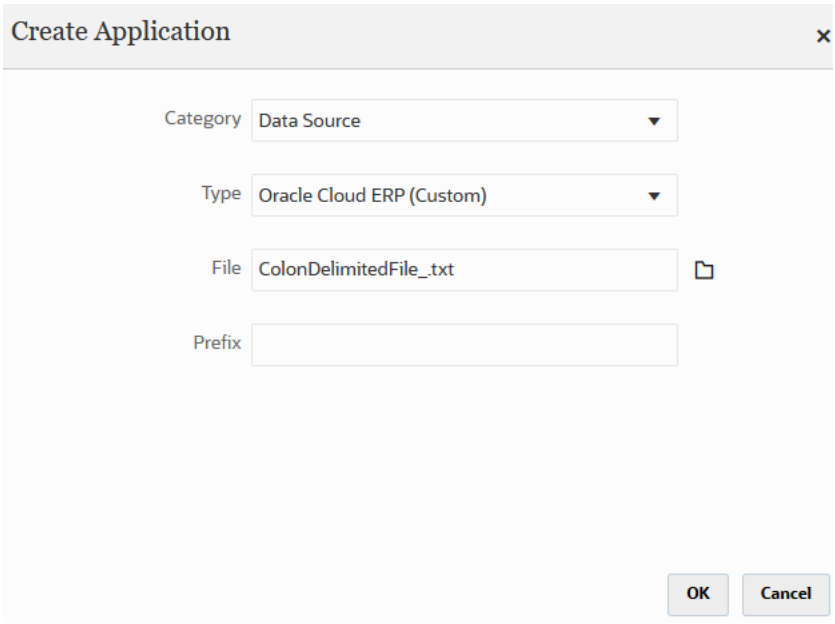
You can import data from the Oracle Cloud ERP using a custom query and load it into the Oracle Fusion Cloud Enterprise Performance Management. The custom query is run against any report from Oracle Business Intelligence Publisher that creates a CSV formatted data file. The system then loads the data to the Cloud EPM. In this case, Data Integration executes the report to extract the data and loads it to Cloud EPM. You use an Oracle Cloud ERP (Custom) data source adapter in Data Integration to import the source CSV file from Oracle Business Intelligence Publisher, and then set the report parameters to use.

> **Note**
>
> An Oracle Cloud ERP integration requires that you have the privileges or user role and data access to all ERP ledgers to integrate data from the Oracle Cloud ERP. For more information, see Security Role Requirements for Oracle ERP Cloud Integrations.
>
> You must register source system for the Oracle Cloud ERP data sources using the system type **Oracle Financials Cloud** and specify your user credentials. This step includes specifying the connection details and testing the connection. For more information, see Configuring an Oracle ERP Cloud Connection.
>
> For detailed information on integrating Oracle Cloud ERP data using a custom query, see Process Description for Integrating Oracle Cloud ERP Data Using a Custom Query.

To register a custom Oracle Cloud ERP application:

1. From the Data Integration home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, click ••• before the **Actions** menu and then click **Create Application**.
3. From **Category**, select **Data Source**.
4. From **Type**, select **Oracle Cloud ERP (Custom)**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



5. In **File**, click **File Browser** to specify the name of the CSV file associated with the Oracle Business Intelligence Publisher report from which to load data.
6. **Optional**: In **Prefix**, specify a prefix to make the application name unique.
   The prefix is concatenated with the application name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.
7. Click **OK**.

Registering Peoplesoft GL Balance Applications

You can use a preseeded query to import General Ledger balances from Peoplesoft sources and then create and register an application that can be loaded to Oracle Fusion Cloud Enterprise Performance Management target applications using the EPM Integration Agent. Peoplesoft General Ledger GL data sources include ledger summary and detail ledger information based on selected ChartField combinations.

In this way, you have direct connectivity from the Cloud EPM to on-premises Peoplesoft data.

For more information about registering Peoplesoft GL balances applications using the EPM Integration Agent, see Using Prepackaged On-premises Adapters.

To register Peoplesoft GL Balance applications:

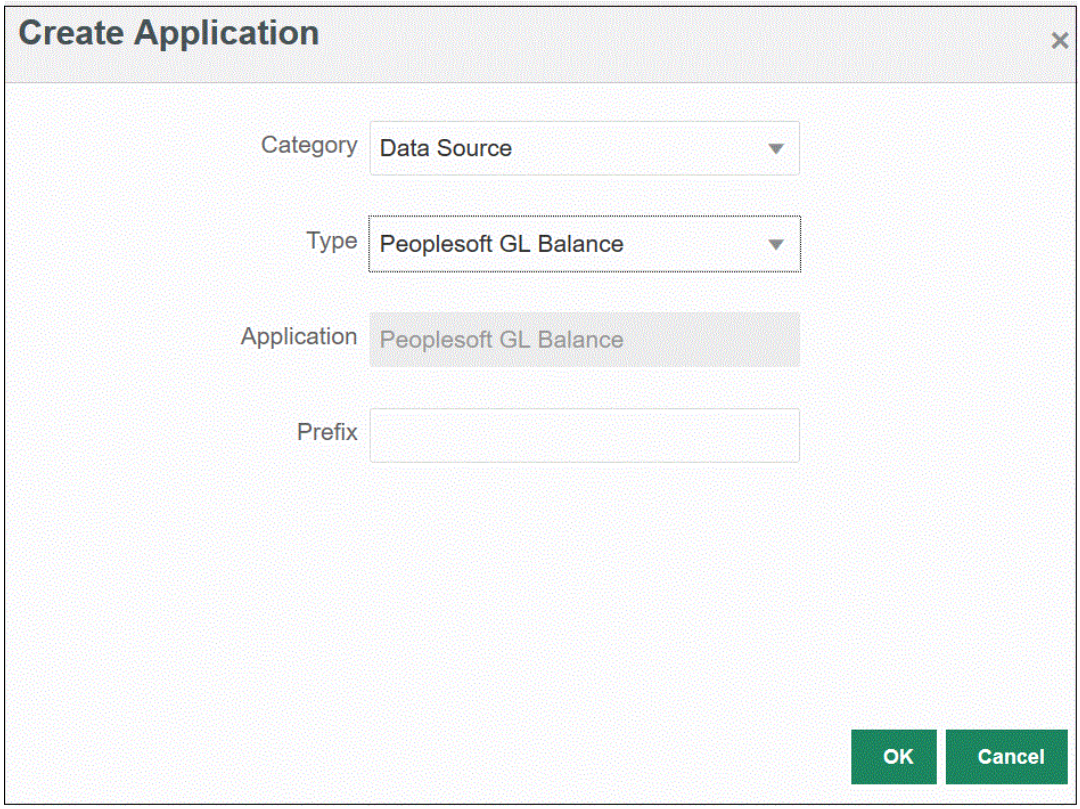
1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, click ••• before the **Actions** menu and then click **Create Application**.
3. From **Category**, select **Data Source**.
4. From **Type**, select **Peoplesoft GL Balance**.
5. The application shown in the **Application** is created and register after you click **OK** on this page.

   You specify connection information for Peoplesoft data source applications on the Options tab in Application Details.

Copyright © 2017, 2026, Oracle and/or its affiliates.

6. **Optional**: In **Prefix**, specify a prefix to make the application name unique.

   The prefix is concatenated with the application name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.



7. Click **OK**.

Registering E-Business Suite Applications

You can use a preseeded query to import General Ledger balances from Oracle E-Business Suite sources and then create and register an application that can be loaded to Oracle Fusion Cloud Enterprise Performance Management target applications using the EPM Integration Agent (see Installing and Configuring the EPM Integration Agent).

In this way, you have direct connectivity from the Cloud EPM to on-premises E-Business Suite data. For more information, see Using Prepackaged On-premises Adapters.

To register an E-Business Suite application:

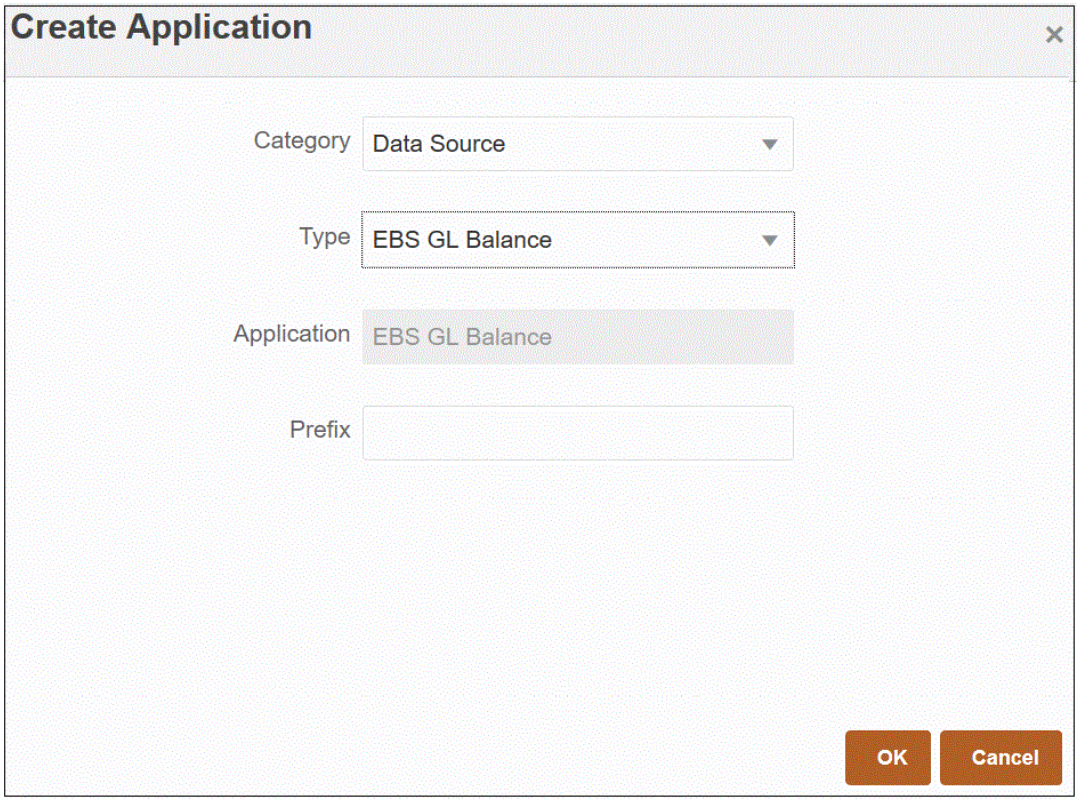
1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, click ••• before the **Actions** menu and then click **Create Application**.
3. From **Category**, select **Data Source**.
4. From **Type**, select **EBS GL Balance**.
5. The application shown in the **Application** is created and register after you click **OK**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

You specify connection information for E-Business Suite data source applications on the Options tab in Application Details.

6. **Optional**: In **Prefix**, specify a prefix to make the application name unique.

   The prefix is concatenated with the application name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.



7. Click **OK**.

Connecting to E-Business Suite and Peoplesoft Data Sources

Direct connectivity from the Oracle Fusion Cloud Enterprise Performance Management to on-premises data sources is available using prepackaged on-premises adapters. These adapters enable you to import data from:

- EBS GL Balances
- Peoplesoft GL Balances

To load data using a prepackaged on-premises adapter:

1. Register the Oracle E-Business Suite or Peoplesoft data source application.

   For information on registering an E-Business Suite data source application, see Registering E-Business Suite Applications.

   For information on registering a Peoplesoft data source application, see Registering Peoplesoft GL Balance Applications.

Copyright © 2017, 2026, Oracle and/or its affiliates.

2. From the **Application** page, click ... to the right of the E-Business Suite or Peoplesoft data source application, and then select **Application Details**.
3. Click the **Options** tab.
4. In **Delimiter**, select the type of delimiter used in the file.

   Available delimiter symbols:

   - Comma (,)
   - Exclamation (!)
   - Not Applicable (NA)
   - Semicolon (;)
   - Colon (:)
   - Vertical bar (|)
5. In **Credential Store**, specify the type of credential store used for the connection.

   Available types of credential stores:

   - Cloud
   - File

   For the **Cloud** credential store type, you store the *user name/password/connect* string in the application.

   For the **File** credential store type, create a file in the on-premises environment to store the JDBC URL, user name and password. The file name must be named *appname*`.cred` and stored in the `config` directory.

   The file must contain the following lines:

   ```
   driver=oracle.jdbc.driver.OracleDriver
   jdbcurl=jdbc:oracle:thin:@host:port/service
   username=apps
   password=w+Sz+WjKpL8[
   ```

   > **Note**
   >
   > The password used for both credential store types, must be encrypted.
   > When the type is a "Cloud" store, type the password in the user interface in the usual way. The system encrypts and stores the password.
   >
   > When the type is a "File" store, you must encrypt the password using the `encryptpassword` utility and store the password in the file. For more information about encrypting your password, see Encrypting the Cloud EPM User Password.
6. **Peoplesoft only**: In **JDBC Driver**, select the type of JDBC driver to use when connecting to the Database Management System (DBMS).

   Available types of JDBC drivers include the following:

   - Microsoft SQL Server
   - Oracle

Copyright © 2017, 2026, Oracle and/or its affiliates.

A JDBC driver is software that enables Java application to communicate with a database. The JDBC driver communicates the connection information to the database and sends the protocol used for transferring the query and result between the client and the database.

7. In **JDBC URL**, specify the JDBC driver URL connection string.

   The JDBC Driver URL connection string enables you to connect to a relational database using Java.

   For an Oracle Thin JDBC driver, the JDBC driver URL includes:

   `jdbc:oracle:thin:@host:port:sid`

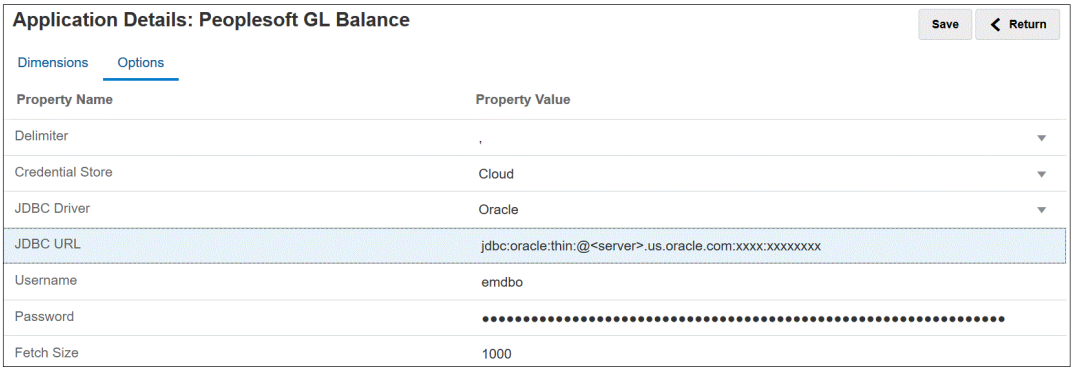
   `jdbc:oracle:thin:@host:port/service`

   For SQL Server, the JDBC driver URL includes:

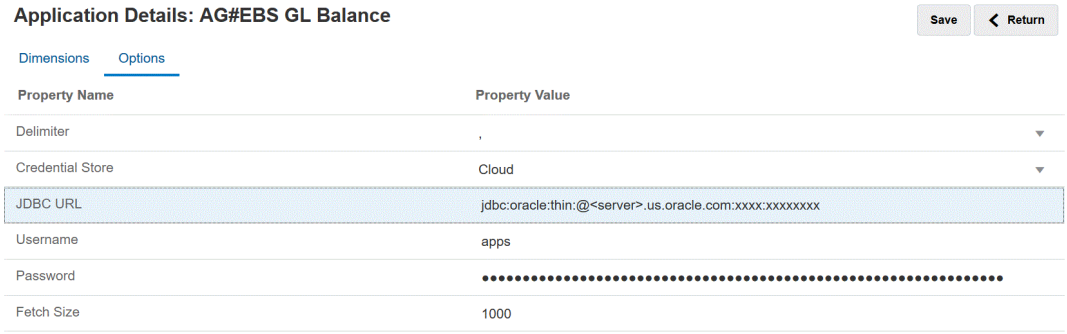
   `jdbc:sqlserver://server:port;DatabaseName=dbname`

8. In **Username**, specify the on-premises database user name.
9. In **Password**, specify the on-premises database password.
10. In **Fetch Size**, specify the number of rows fetched (that the driver processes) with each database round trip for a query.

    The following shows PeopleSoft connection options on the Options tab.



The following shows E-Business Suite connection options on the Options tab.



11. Click **Save**.

Registering Oracle HCM Cloud Applications

For information on registering an Oracle Human Capital Management Cloud application, see Integrating with the Oracle HCM Cloud.

Copyright © 2017, 2026, Oracle and/or its affiliates.

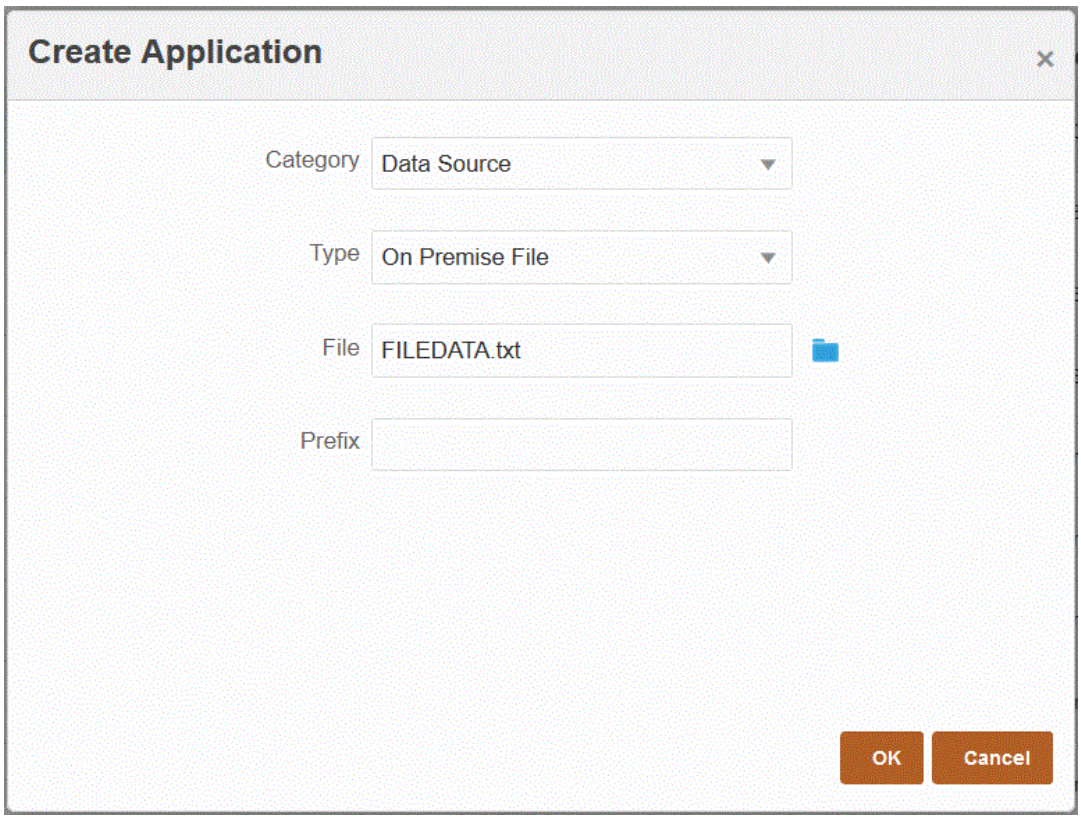
Registering an On-Premises File

You register an on-premises file application when using the Quick Mode to load files using the EPM Integration Agent. This type of adapter enables customers to load very large files from on-premises file data source to the Oracle Fusion Cloud EPM. It extends the Quick Mode method where the Cloud EPM database is bypassed for staging and processing, eliminating any performance bottlenecks and improving the performance and scalability of the load process.

An on-premises file adapter requires no connection parameters or query. However, a delimited file with a header used for the import format is required for the load.

For more information on using the on-premises file data source, see Using an Agent File Adapter.

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, click ••• before the **Actions** menu, and then click **Create Application**.
3. From **Category**, select **Data Source**.
4. From **Type**, select **On Premise File**.



5. In **File**, click **File Browser** and specify the source data file to use as the template. When selecting a file, note the following:
   - The source data file must be a delimited data file.

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Data files used must contain a one-line header, which describes the delimited columns.
- Both numeric and non-numeric data can be loaded.

6. **Optional**: In **Prefix**, specify a prefix to make the application name unique.

   The prefix is concatenated with the application name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.

7. Click **OK**.

Registering File Applications

You can register a file as a data source. A file data source enables a business user to easily import and register data from any file-based source, whether a fixed width file or a delimited, by way of a file format, and requires limited technical help, if any, when loading into a Oracle Fusion Cloud Enterprise Performance Management application. Additionally, you can specify different file sources to logically group files based on content.

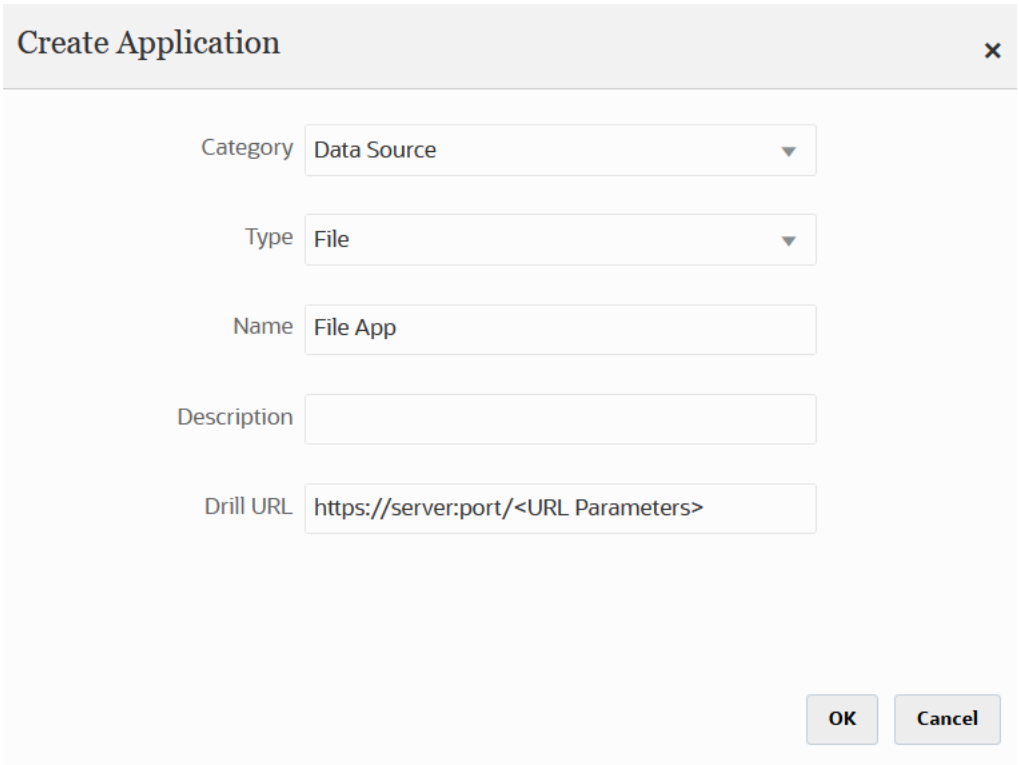
To register a file as a data source application:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, click ••• before the **Actions** menu, and then click **Create Application**.
3. On the **Create Application** page, and then from **Category**, select **Data Source**.
4. From **Type**, select **File**.
5. In **Name**, specify a user-defined name for the file application.
6. In **Description**, specify a description of the file application.
7. In **Drill URL**, specify the custom URL for the custom drill.

   The drill URL must include the server, port, and URL parameters, for example, `https://server:port/<URL Parameters>`.

8. Click **OK**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



You select the actual file to use for the file application when you create the integration from the File Browser. For more information, see Using the File Browser.

Registering an Application for a Class of Dimensions or Dimension Type

You can register applications of the type that include only a class of dimensions or a dimension type. These types of applications are used to load metadata and include the following applications, which the system creates automatically:

- Account
- Entity
- Custom
- Scenario
- Version
- Smartlist

For information on using an application for a class of dimensions or dimension type for a metadata load, see Integrating Metadata.

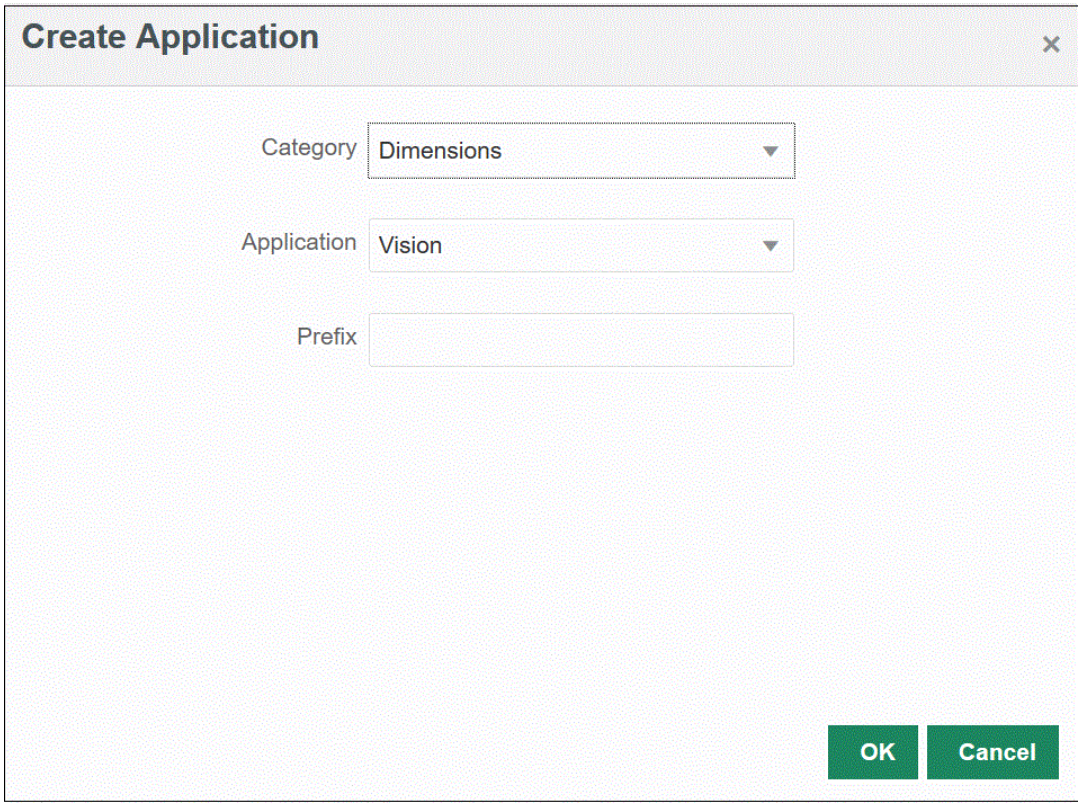
Copyright © 2017, 2026, Oracle and/or its affiliates.

Note

- Only "Generic" type dimensions are allowed for metadata/dimensions in the target applications. However, you can map fields on the source extract to Attribute columns on the Map Dimensions page. This means you can't apply mappings on the Attribute columns as might with a LOOKUP dimension, but this method is helpful when attributes don't need further mapping and are only used to derive mappings for other fields.
- If you need to import more than one custom dimension, then create an integration for each dimension and set the dimension name to the required dimension in the Integration Options (see Defining File-Based Options) and not in the Application options.
- After registering an application, avoid modifying the Dimension Classification/Data Table Column names for Oracle Fusion Cloud Enterprise Performance Management applications. In some cases such as Oracle ERP Cloud General Ledger applications, modifications to the dimension classifications names are restricted.

To register an application for a class of dimensions or dimension type:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, click ••• before the **Actions** menu, and then click **Create Application**.
3. On the **Create Application** page, and then **Category**, select **Dimensions**.

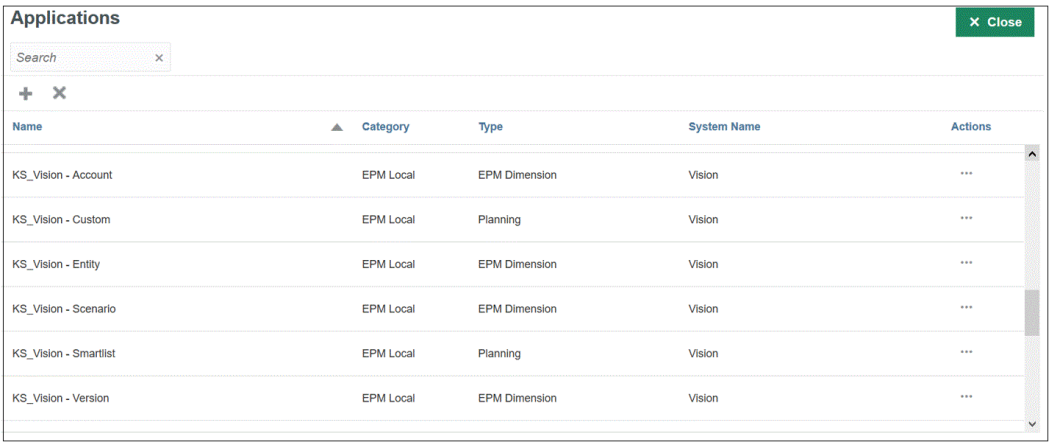


Copyright © 2017, 2026, Oracle and/or its affiliates.

4. From **Application**, select the application for the primary service application.
5. **Optional**: In **Prefix**, specify a prefix to make the application name unique.

   The prefix is concatenated with the application name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.
6. Click **OK**.

   In the following example the applications have been created automatically for the KS_Vision application: KS_Vision - Account, KS_Vision - Custom, KS_Vision - Entity, KS_Vision - Scenario, KS_Vision - Smartlist, and KS_Vision - Version.



Registering a Data Export File Application

Shows how to register a data export file application.

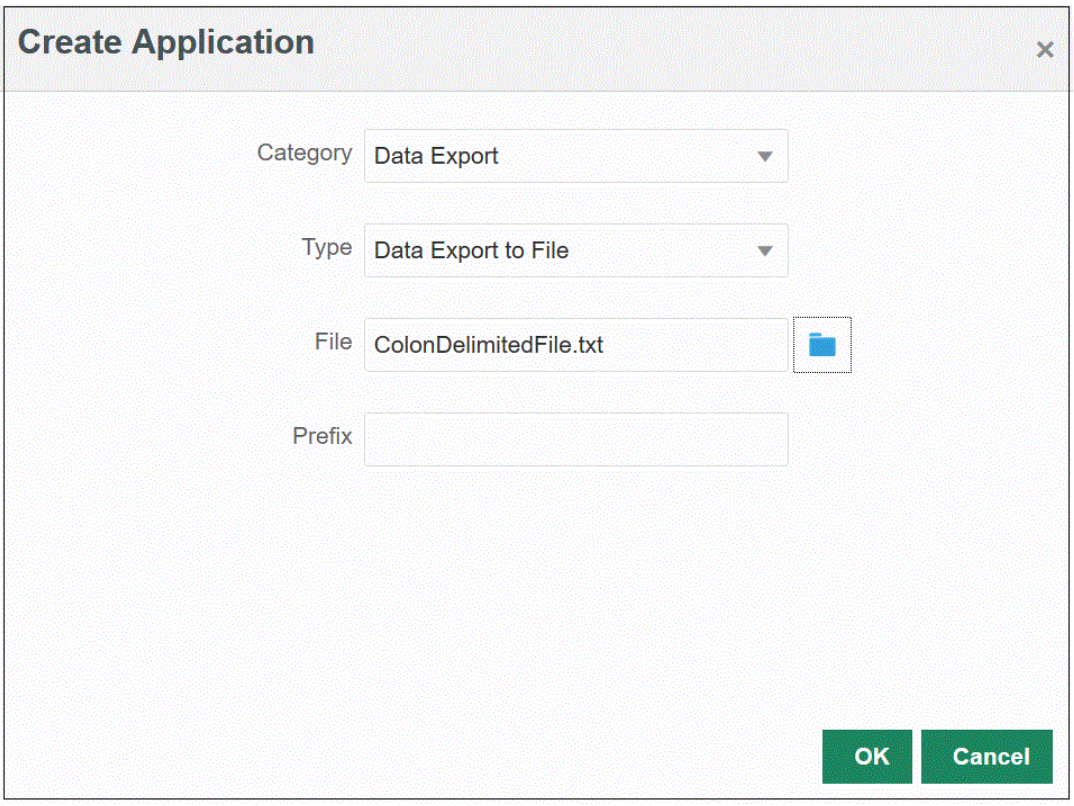
You create an application for a data export file when you need to export data from an Oracle Fusion Cloud Enterprise Performance Management.

> **Note**
>
> For information on the Data Export to On-Premises Database application type, see Registering the Data Export to On-Premises Application.

To register an application for a data export file:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, click ••• before the **Actions** menu, and then click **Create Application**.
3. On the **Create Application** page, and then **Category**, select **Data Export**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



4. From **Type**, select **Data Export to File**.
5. In **File**, click **File Browser** and specify the name of the source file from which to create the application.
6. **Optional**: In **Prefix**, specify a prefix to make the application name unique.

   The prefix is concatenated with the file name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.
7. Click **OK**.

> **Note**
>
> For more information about create a data export file integration, see Creating a Data Export File Integration.

Registering a Data Export File as an EPM Data File Type Application

You can register an EPM Data File type application when you need to export text, smart list and dates from your Oracle Fusion Cloud Enterprise Performance Management application to an EPM Data File using Quick Mode.

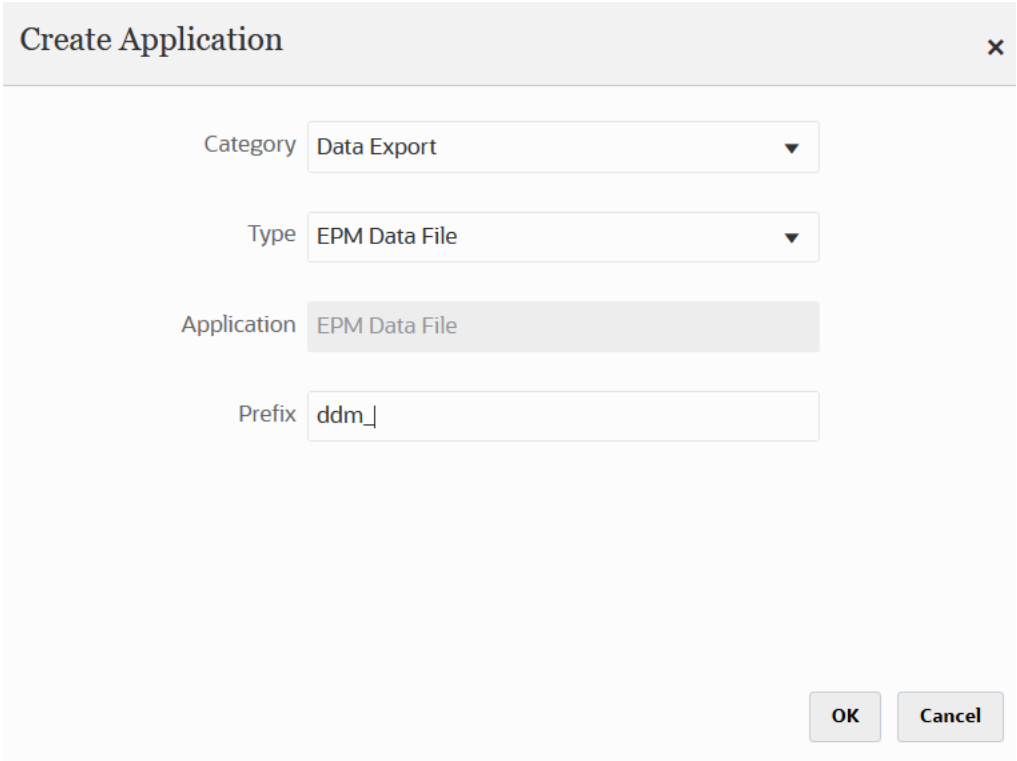
To register an Data File as an EPM Data File type application:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, click ⋯ before the **Actions** menu, and then click **Create Application**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

3. On the **Create Application** page, and then **Category**, select **Data Export**.
4. From **Type**, select **EPM Data File**.

   The **Application** field shows the name **EPM Data File**



5. **Optional**: In **Prefix**, specify a prefix to make the application name unique.

   The prefix is concatenated with the file name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.

Application Details

Use the Application Details feature to view, edit, or select the details associated with an application.

Use the Application Details page to view, edit, or select the details associated with an application.

Application details include:

- Dimensions—lists dimensions for a selected application. For selected application types, you can reclassify the dimensions and change the dimension class as necessary.

  For more information, see Defining Application Dimension Details.
- Options—Enables you to select options that support how you load date including load methods, date formats, batch, purge options. For more information, see Defining Application Detail Options.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Defining Application Dimension Details

The Application Dimensions page lists dimensions for a selected application. A dimension is a structure that categorizes data in order to enable users to answer business questions. Each dimension usually contains a hierarchy of related members grouped within it. Commonly used dimensions are Customers, Products, and Time.

For example, in Planning, there are seven standard dimensions: Account, Entity, Scenario, Version, Period, Years, and Currency. In addition, general Planning applications can include up to and twenty-five custom dimensions.

Dimension details differ for each application type. For selected application types, you can reclassify the dimensions and change the dimension class as necessary.

To define dimension details:

1. From the **Application** page, click ••• to the right of the application, and then select **Application Details**.
2. Select the **Dimensions** tab.
3. Select **Create Drill Region** option to include a dimension in the Drill Region data slice definition. When the dimension is selected during the data load process, the system builds the data slice by including the unique list of values for each of the dimension.

   > **Note**
   >
   > The Drill Region simply defines the cells for which the drill icon is enabled in the Data Forms and SmartView. It is recommended to use a minimum set of dimensions to define the drill region. If a large number of dimensions are included in the drill region, then the size of the drill region becomes large and consumes system resources every time a form is rendered. For Planning applications, use dimensions with small number of members like Scenario, Year, Period, Version to define the drill region. For an Financial Consolidation and Close application, use only the Data Source for defining the drill region.

   If you want to define a more granular drill region with multiple dimensions, then use the Drill Region tab in Define Drill Details to edit the region definition. For example, you can use member functions like iDescendants to define the region instead of individual members.

   If you edit the drill region manually, set the **Drill Region** option to **No** in **Application Options**.

4. For each dimension name not defined in the application, click •••), and then click **Add Dimension**.

   The dimension class is a property that is defined by the dimension type. For example, if you have a Period dimension, the dimension class is also "Period." For Oracle Essbase applications, you must specify the appropriate dimension class for Account, Scenario, and Period. For Oracle Hyperion Public Sector Planning and Budgeting applications, you must specify the dimension class for Employee, Position, Job Code, Budget Item, and Element.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> It's recommended not to manually edit the application, specifically for EPM Local and Cloud EPM applications. Use refresh metadata only to add the missing dimenions. For Data Source and Data Target applications, you must reregister the apps.

5. The **Data Table Column Name** shows the table column name of the column in the staging table (TDATASEG) where the dimension value is stored.

   The Data Table Column Name is an informational field and cannot be changed.

6. In **Mapping Sequence**, specify the order in which the maps are processed.

   For example, when "Account" is set to **1**, "Product" is set to **2**, and "Entity" is set to **3**, then Data Integration first processes the mapping for Account dimension, followed by Product, and then by Entity.

7. **Data Export Applications only**: In **Column Sequence**, specify the order of each column in the data export file.

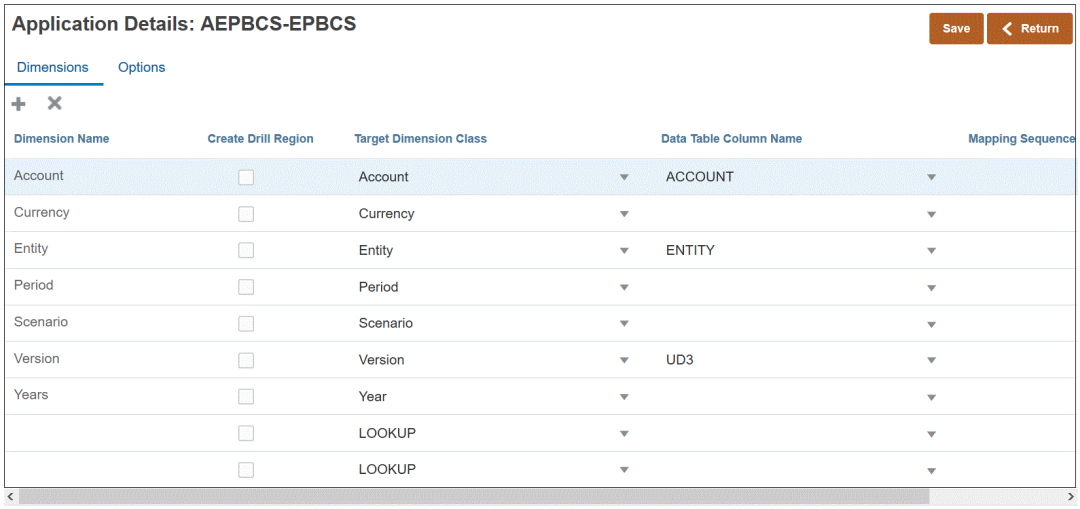
   For example, when "Account" is set to **1**, "Product" is set to **2**, and "Entity" is set to **3**, then Data Integration, shows "Account" first, "Product" is shown second, and "Entity" is shown third.

   By default, Data Integration assigns the "Account" dimension as the first column in the order.

8. To delete a dimension, in the Actions column, click the (**Delete**) icon.

9. Click **Save**.

   The application is ready for use with Data Integration.



> **Tip**
>
> To edit the dimension details, select the application, then edit the application or dimension details, as necessary.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Adding Lookup Dimensions

Lookup dimensions can be created and assigned with data columns for target applications and are used for mapping and reference.

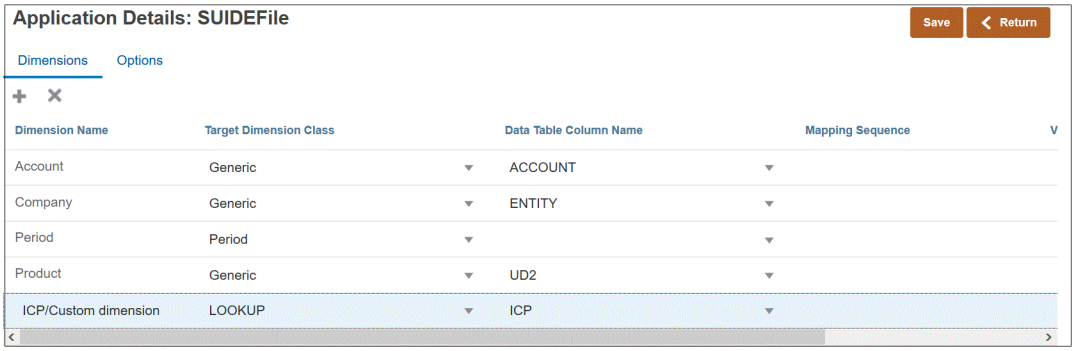
A lookup dimension can be added in Data Integration that does not have a corresponding dimension in the target application. These types of dimensions are useful when you need to add content that determines how to transform another column. They can be used with member mapping functionality to cross-reference multiple source segments and chartfields and assign a target value.

To add a lookup dimension:

1. From the **Application** page, click ••• to the right of the Oracle Workforce Planning application, and then select **Application Details**.
2. Select the **Dimensions** tab.
3. Click •••), and then click **Add Dimension**. Select the target dimension class name from the **Target Dimension Class** drop-down.

   A **LOOKUP** dimension entry is added to the list of dimensions.
4. In **Dimension Name**, enter the name of the lookup dimension and click **Save**.

   The lookup dimension is added to the dimension detail list with the target dimension class name of "LOOKUP." To use the lookup dimension as a source dimension, make sure you map it in the import format.



Defining Application Detail Options

When integrating source applications with target applications, you can select options that support how you load date including load methods, date formats, batch, and purge options.

> **Note**
>
> For information about Application Details for Oracle ERP Cloud Data Sources, see Defining Application Details for Oracle ERP Cloud Data Sources.

To define target options:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**

Copyright © 2017, 2026, Oracle and/or its affiliates.

2. From the **Application** page, click ••• to the right of the application, and then select **Application Details**.
3. Select the **Options** tab.
4. Select the option and select the needed action.

Option	Description
Load Method	Select the method for loading data to the target application. Available methods: • **Numeric**—Loads numeric data only. Planning data security is not enforced with this method. • **All data types with auto-increment of line item**—This method requires that you define the data and driver dimensions, and unique identifiers for the Planning application. You define data load and driver dimension in the Data Dimension for Auto-Increment Line Item field and Driver Dimension for Auto-Increment Line Item field below. Replace Mode is not supported for the load method "All data types with auto-increment of line item." • **All data types with security**—Loads Numeric, Text, Smartlist, Date data types. If the Planning administrator loads data, Planning data security *is not* enforced. If a Planning non-administrator user loads data, then Planning data security is enforced. A Planning non-administrator user can load only 500,000 cells of data.
Batch Size	Specify the number of rows read at a time from the file to memory. This parameter is mainly used for performance. When data is loaded, this setting determines how many records are stored in the cache. For example, when **1000** is specified; 1,000 records are stored in cache. Similarly, when **5000** is specified, 5,000 records are stored in cache and commit. Determine this setting by Server Memory and adjust as needed.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Option	Description
Drill Region	Tap the slider on to enable the drill region. When this option is enabled, a drillable region is created to use the drill through feature. When loading data, the drill region is loaded to Planning data. The drill region by scenarios is created. For any cube (Planning cubes or Planning databases), the name of the drill region is FDMEE_<name of the scenario member>. When creating the drill region, the system checks if a dimension is enabled for the drill. Members of enabled dimensions selected in data loads, are included in the drill region filter. If no dimensions are enabled, the Scenario, Version, Year, and Period dimensions are enabled by default. You can enable additional dimensions, and the subsequent data load considers members of newly enabled dimensions. If you disable any dimensions which were previously included in a drill region used for drill creation, members of such dimensions are not deleted during the subsequent data loads. If needed, you can remove obsolete members manually.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Option	Description
Enable Drill from Summary	Select **Yes** to drill down from summary members in a Planning data form or report and view the detail source data that make up the number. After enabling this option and loading the data with the Create Drill Region option set to **Yes**, the Drill icon is enabled at the summary level. Drill is limited to 1000 descendant members for a dimension. **Note** If you Enable Summary Drill, do not include the dimension you want to drill from the Parent Members in the drill region definition. If you absolutely need to include this dimension, then disable the auto drill region creation and then maintain the drill region manually using Calculation Manager user interface. Use Essbase member function like Descendants to enumerate the members you want to include in the drill region. Summary drill is available for local service instances only. It is not available between cross service instances or hybrid deployments. **Note** In order to support a summary drill, your drill query should not include an operator in the WHERE clause of the QL. For example your query should be: WHERE COMPANY ~ENTITY~. The system determine appropriate condition (IN, LIKE) based on the number of descendants.
Purge Data File	Tap the slider on to delete the data file from the application `outbox` directory when a file-based data load is successful. Tap the slider on to delete the file or tap the slider off to retain the file.
Date format for date data	Select the format used for loading date data. Use the date format based on the locale settings for your locale. For example, in the United States, enter the date using the format **MM/DD/YY** format.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Option	Description
Data Dimension for Auto-Increment Line Item	Select the data dimension that matches the data dimension you specified in Planning. This option is used for loading incremental data using a LINEITEM flag. See Loading Incremental Data using the LINEITEM Flag.
Driver Dimension for Auto-Increment Line Item	Select the driver dimension that matches the driver dimension you specified in Planning. This setting is used for loading incremental data using a LINEITEM flag. See Loading Incremental Data using the LINEITEM Flag.
Member name may contain comma	If the member name contains a comma, and you are loading data to one of the following services, set this option to **Yes**, and then load the data: • Planning Modules • Planning • Financial Consolidation and Close • Tax Reporting
Workflow Mode	Select the data workflow method. Available options: • Full—Data is processed in the TDATASEG_T table and then copied to the TDATASEG table. All four Workbench processes are supported (Import. Validate, Export, and Check) and data can be viewed in the Workbench. Drill-down is supported. The Full Workflow Mode is the default mode. • Full No Archive—Data is processed in the TDATASEG_T table and then copied to TDATASEG table. All four Workbench processes are supported (Import, Validate, Export, and Check). Data can be viewed in the Workbench but only after the import step has been completed. Data is deleted from TDATASEG at the end of the workflow process. Drill-down is not supported. • Simple— Data is processed in the TDATASEG_T table and then exported directly from the TDATASEG_T. table. All data loads include both the import and export steps. Data is not validated and any unmapped data result in load failure. Maps are not archived in TDATAMAPSEG. Data cannot be viewed in the Workbench. Drill-down is not supported.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Option	Description
Enable Data Security for Admin Users	Enables data validation when an administrative user loads data. In this case, all data validations in the data entry form are enforced while loading data. Due to the enhanced validations, the performance of data load will be slower. When the 'Enable Data Security for Admin Users' is set to **No** (default value), then data loads by the administrator are performed using the Outline Load Utility (OLU). In this case, performance is faster, but you are unable to get a detailed error report for any rows that are ignored for any reason. When this option is set to **Yes**, data is validated for administrator and non-administrator data loads in the same manner. Validations include: security checks, intersection validations, read-only cells, dynamic calc cells, etc. In addition, a detailed error list for any rows that are rejected or ignored is available and no additional Planning permissions are needed. However, performance may be slower even for administrators.
Display Validation Failure Reasons	Enables you to report rejected data cells and the rejection reason when you load data in a data validation report. Select **Yes** to report rejected data cells and the rejected reason. The limit for the number of rejections reported is 100. The data validation report is available for download from the Process Details page by clicking the Output link. In addition, a copy of the error file is stored in the `Outbox` folder. Select **No** to not report rejected data cells and the rejection reason.
Drill View from Smart View	Specify the custom view of columns from the Workbench when displaying customized attribute dimension member names in Oracle Smart View for Office drill-through reports. Custom views are created and defined in the Workbench option in Data Integration. When the custom view has been defined and then specified in the Drill View from Smart View field, in Smart View you can click the drill-through cell and select Open as New Sheet, and the drill-through report opens based on the view defined in the Workbench. If no views are defined on the Application Options page, the default view is used, meaning that attribute dimensions do not display customized member names in Smart View. For more information, see Defining a Custom View in the Workbench.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Option	Description
Replace for Non Admin Load Method	Select the Replace method when loading data to a Planning application for non-administrator users. Available methods: • None—Performs no replace. • Numeric Data Only—Performs replace when the load method is Numeric Data only. • All Load Methods—Perform replace for all load methods.
Summary Drill Behavior when more than 1000 descendants	Select how a summary drill functions with more than 1,000 descendants. Available options: • Ignore • Limit If the number of descendants in summary drill is greater than 1,000 and the **Summary Drill Behavior when more than 1000 descendants** option is **Ignore**, then the system ignores the dimension in the filter and returns the drilled data based on filters on the other dimensions. A maximum of three dimensions can be ignored. If the number of descendants in summary drill is greater than 1,000 and the **Summary Drill Behavior when more than 1000 descendants** option is **Limit**, then the system considers only the first 1,000 members for the dimension while returning the data rows. If you select to perform a summary drill on more than 1,000 descendants and include only top-level dimensions but the results are the same as the base-level dimension, then also select base-levels intersections for at least three dimensions.

5. Click **Save**.

Defining Application Details for Oracle ERP Cloud Data Sources

After registering an Oracle ERP Cloud data source application, you can manage selected details associated with the application.

To define Oracle ERP Cloud application detail:

1. From the **Data Integration** home page, and then **Actions**, select **Applications**.
2. From the **Application** page, click ••• to the right of an Oracle ERP Cloud data source application, and then select **Application Details**.
3. From the **Application Details** page, click the **Options** tab.
4. Select the setting to change for the application and click **Save**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Oracle ERP Cloud Application Details	Descriptions
Convert Period Tokens to Upper Case	Enables you to convert a report's period token, such as $START_PERIODKEY$ to uppercase when the resulting period name may be in mixed case, such as Jan or Feb. When this setting is enabled, the system converts the mixed case period to uppercase only, such as JAN or FEB. Select **Y** to enable the conversion of the period token. Select **N** to not convert the period token.
Notification Type	Select the type of notification to send when Oracle Business Intelligence Publisher reports are executed in the Oracle ERP Cloud. • Email Notification • Bell Notification • Email and Bell Notification
Notification Event	Select the event that triggers the notification to send when BI Publisher reports are executed in the Oracle ERP Cloud: • Always • On Success • On Failure
COA Application Name	Select the Chart of Accounts application name from the Oracle ERP Cloud to use for the integration.
Segment Name	Select the segment name from the segment group to use for the integration.
Hierarchy Top Name	Select the hierarchy top node used to filter the portion of the source viewpoint to use for the integration.

Refreshing Oracle ERP Cloud Source

When you have registered the Oracle ERP Cloud data source application and selected the new COA source, you can have the system automatically refresh the COA data directly from the source:

To refresh the COA data from the source automatically

1. From the **Data Integration** home page, and then **Actions**, select **Applications**.
2. From the **Application** page, click ⋯ to the right of the Oracle ERP Cloud data source application, and then select **Application Details**.
3. From the **Application Details** page, click the **Set Defaults** tab.
4. From the **Refresh from Source** drop-down, select the refresh option.

   Available options:

   - Yes—Refresh data directly from the source automatically.
   - No—Do not refresh data from the source automatically.
5. Select the setting to change for the application and click **Save**.

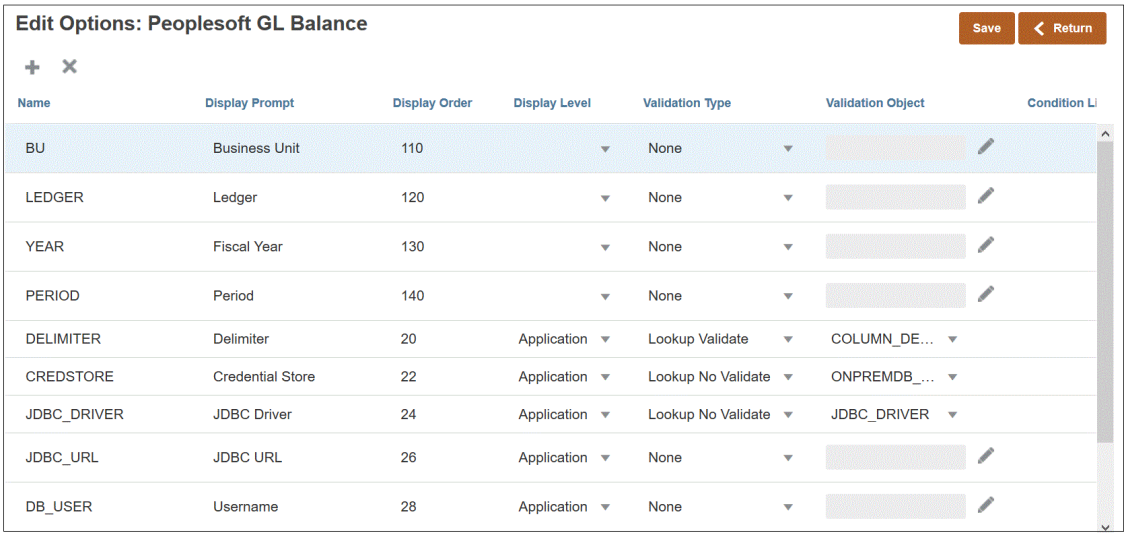
Editing Options

Use editing options to add and edit application filters that are associated with a data source or data export to file application.

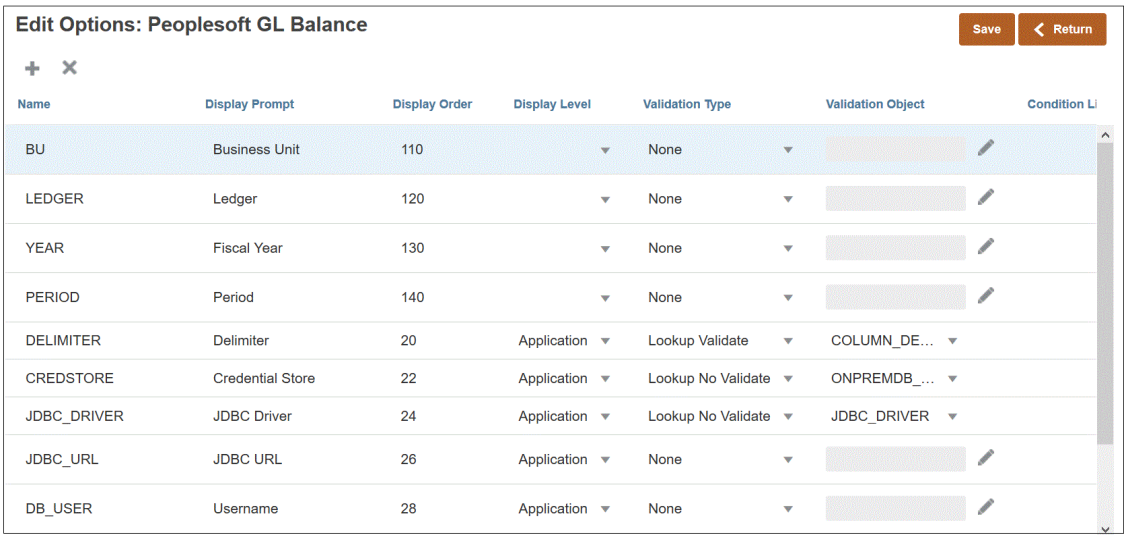
Copyright © 2017, 2026, Oracle and/or its affiliates.

You can add and edit application filters that are associated with a data source or data export to file application when the user selects integration options. You can specify a single filter condition or multiple filters conditions, and additionally specify the exact values that you want returned.

In the following example, filter input conditions have been defined for the "Business Unit," "Ledger," "Year," and "Period" on the Edit Options page.



Here are how the filters were defined for the data source application:



In some cases, application filters are defined explicitly for a data value. For example, if you are working with an Oracle Human Capital Management Cloud data source, it is recommended that you do not modify or change the filter definitions if you use the predefined integration with Oracle HCM Cloud.

To add a data source or data export to file application filter:

1. From the **Application** page, click ⋯ to the right of the Data Source or Data Export to File application, and then select **Edit Options**.
2. From **Edit Options**, click **Add** (✚)

   A blank entry line display.
3. In **Name**, specify the name of the display prompt in the user interface.

Copyright © 2017, 2026, Oracle and/or its affiliates.

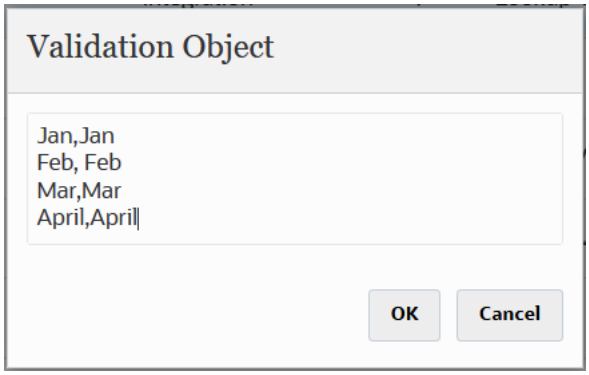
> **Note**
>
> When you register an Oracle E-Business Suite, Peoplesoft, or a custom import database source application, the out-of-box setup automatically adds a filter for the "Period" with the display name shown as "Period." In this case, the system throws the error: "EPMAT-1:EPMFDM-ERROR: 'periodName' parameter format is invalid." As a workaround, select the application and change the display name for "Period" to another name on the Edit Options page. Additionally, you can use the new name in an EPM Automate command to pass it as a filter.

4. In **Display Prompt**, specify the name of the display prompt for the filter on the Edit Integration page.
5. In **Display Order**, specify the display order of the filter on the Application Filters tab. If this field is blank, the custom filter cannot be displayed, and the default value is used as the filter value.

   For example, enter **99** to show the filter in the 99th position sequence or position in a list of filters. Display orders are listed from lowest to highest.
6. In **Property Level**, select where to display the filter. Available options are:
   - Application only
   - Application and Integration
   - Integration only
7. To provide a drop-down of values listed using the lookup type, in **Validation Type**, select one of the following:
   - None
   - Number
   - Lookup Validate
   - Lookup No Validate
   - Yes/No
   - File
   - Date
   - Query
   - Choice list— You can enter a list of values in the Validation Object field. Each value is entered on a new line. The list of values (LOV) on the Option page shows the values as a list.
   - REST LOV
8. In **Validation Object**, enter a list of lookup objects to be validated by type.

   To enter a list of values if you chose the **Choice list** validation type above, click , then enter the list of values on the Validation Object page, and then click **OK**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



9. In **Condition List**, provide a condition list based on one or all the following:
   - Equal
   - In
   - Like
   - Between

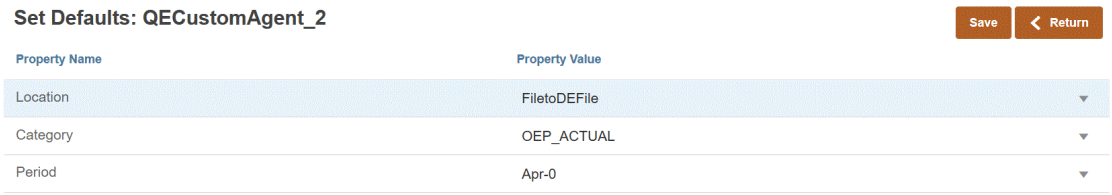
   The condition values can be `Equal`, `Like`, or `In`. The condition values are stored as a drop-down list in CONDITION_LIST field. If the value is `EQ,IN`, then show only `Equal` and `In` in the Condition drop down. If the value is `EQ LIKE`, then show only Equal, Like in the drop down. Any combination of `EQ,IN,LIKE` are stored in the field. If only one value is provided in the Condition List, then the condition cannot be changed. The default value is `EQ`. When specifying multiple conditions, prefix the condition with a comma. For example, to use the `IN` and `LIKE` conditions, enter: `,IN,LIKE`
10. Click **Save**.

Setting Default Options

Use default options to define default values to be used as application filters.

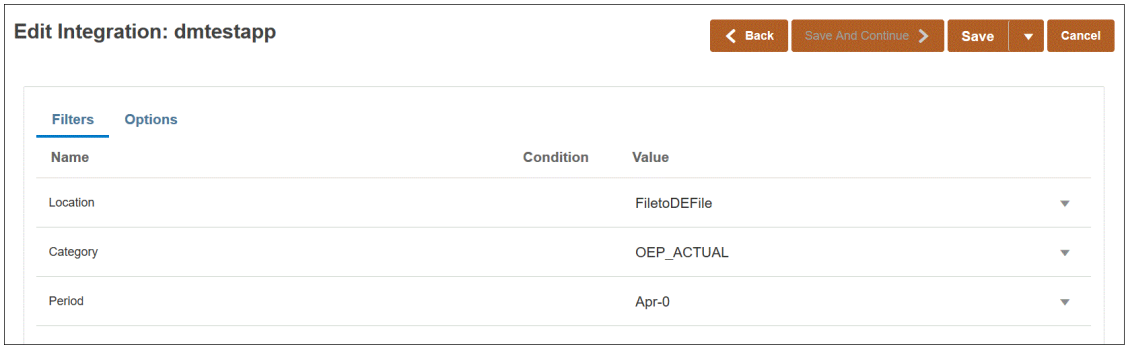
Use the **Set Default** option to define default values to be used as application filters on the Integration Options page.

In the following example, default values have been defined through the Set Default option for "Location," "Category," and "Period" filters in the QECustomAgent_2 application.



When the user selects an integration based on the QECustomAgent_2, the default values are shown for the "Location," "Category," and "Period" filters.

Copyright © 2017, 2026, Oracle and/or its affiliates.

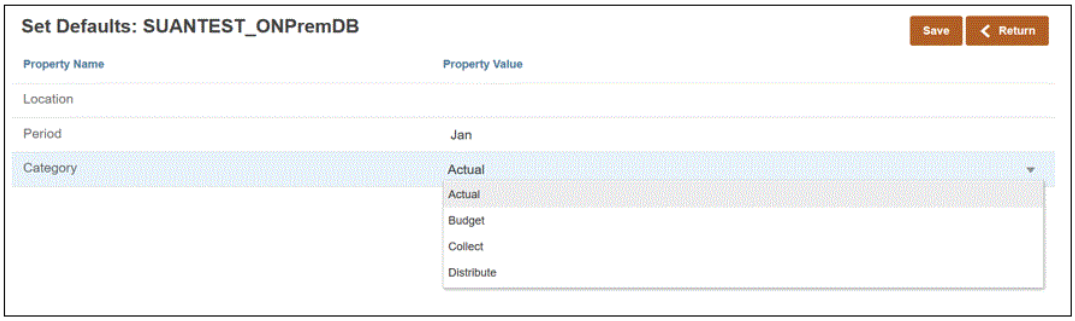


To set default options:

1. From the **Application** page, click ••• to the right of the data source or data export to file application, and then select **Set Defaults**.
2. From **Set Defaults**, navigate to the **Property Name**, and then specify the **Property Value**.

   For example, for the property name "Period,", specify **Jan-10** as the property value.

   If a property has a down arrow button, you can select from a list of values to use as a default value.



3. Click **Save**.

Refreshing Members

Use the Refresh Member option to update members in each dimension manually when artifacts such as dimensions and members have been modified in the target application.

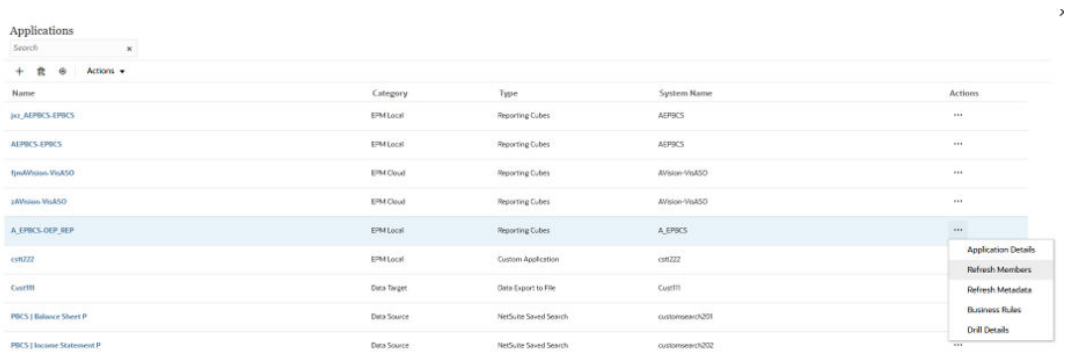
You can update members in each dimension manually when artifacts such as dimensions and members have been modified in the target application.

> **Note**
>
> Parallel execution of Refresh Members isn't supported.

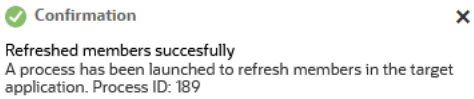
To refresh members for a selected target application:

1. From the **Application** page, click ••• to the right of the target application.
2. From the **Actions** drop-down, select **Refresh Members**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



A confirmation message displays when the refresh is complete.



Refreshing Metadata

Refresh metadata for Predictive Cash Forecasting customers

The Refresh Metadata option enables Predictive Cash Forecasting customers to refresh metadata seamlessly from Fusion ERP data sources to the integrated to the Oracle Fusion Cloud EPM, with the exception of input currencies.

When you use the Refresh Metadata option, the system replaces metadata with new metadata from the load file. Replacement is useful for making minor changes to the metadata, such as adding an account. For example, if your application includes a North America entity, and you load entities from a metadata file, the attributes for the North America entity in the file replace the attributes for the North America entity in the application.

To refresh metadata:

1. From the **Application** page, click ••• to the right of the target application.
2. From the **Actions** drop-down, select **Refresh Metadata**.

   When the metadata has been updated, the systems displays the message:

   "Refreshed metadata successfully."

Copyright © 2017, 2026, Oracle and/or its affiliates.

6

Configuring Source Connections

Use Configure Connections to register, manage, and delete direct connection information to the following sources:

- Configuring an Oracle ERP Cloud Connection
- Configuring an Oracle HCM Cloud Connection
- Configuring an Oracle NetSuite Connection
- Configuring an Oracle Autonomous Database Connection

Configuring an Oracle ERP Cloud Connection

Define Oracle ERP Cloud source connections to connect to Oracle ERP Cloud data sources.

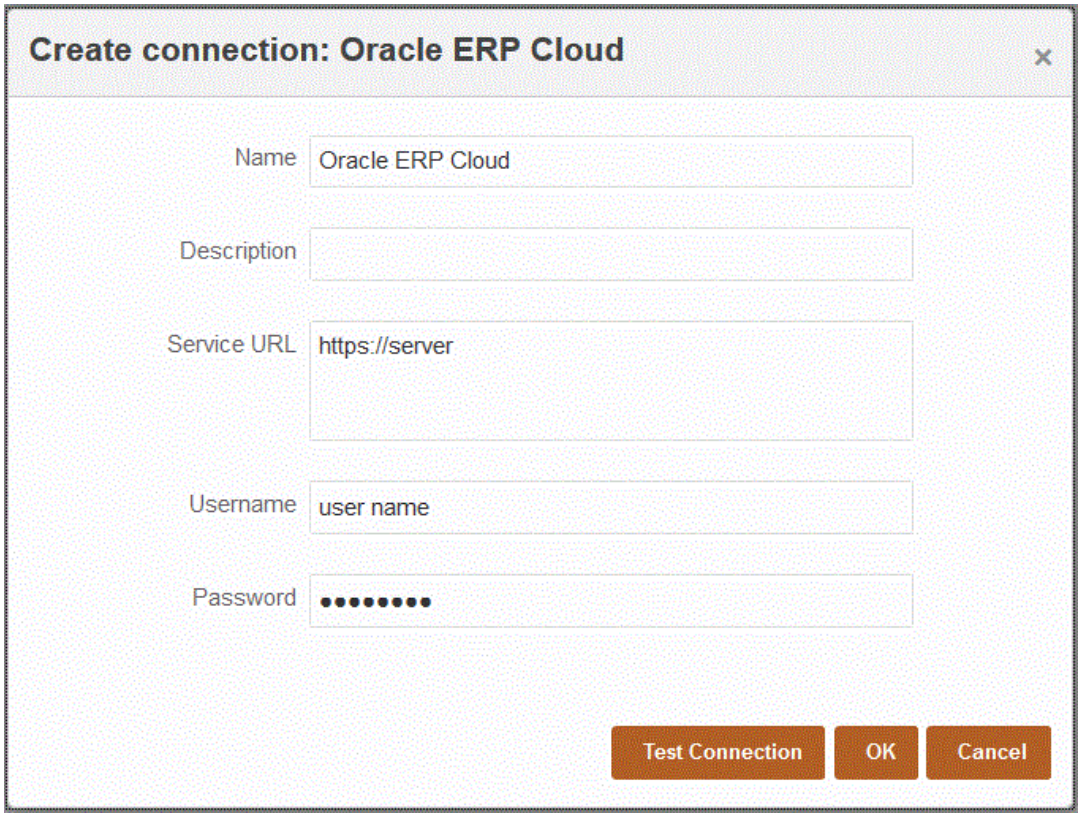
Oracle ERP Cloud source connections can be used to register and manage connection the following data sources:

- Oracle ERP Cloud
- Oracle ERP Cloud (Receivables Transactions)
- Oracle ERP Cloud (Trial Balance Average)
- Oracle ERP Cloud (Custom)
- Oracle ERP Cloud (Payables Transactions)
- Oracle ERP Cloud (Trial Balance)
- Project Management
- Budgetary Control commitments, obligations, expenditures, and write-back data sources. Budget Review data sources

To create an Oracle ERP Cloud Connection:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, from the **Actions** menu, click **Configure Connections**.
3. On the **Connections** page, click ⋯, and then click **Oracle Cloud ERP**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



4. In **Name**, enter the source system name.
5. In **Description**, enter a description of the source system.
6. In **Service URL**, enter the server information for the web services.

> **Note**
>
> When configuring connections for Oracle Fusion Cloud EPM and Oracle ERP Cloud, the URLs specified for connections must use the secure HTTP protocol, for example, "https://server".

7. In **Username**, enter the Oracle ERP Cloud user name.

   Enter the name of the Oracle ERP Cloud user who launches the process requests to send information between Oracle Fusion Cloud Enterprise Performance Management and the Oracle ERP Cloud . This user must have an assigned Oracle General Ledger job role such as "Financial Analyst," "General Accountant," or "General Accounting Manager."
8. In **Password**, enter the Oracle ERP Cloud password.

   You must update this password anytime you change your Oracle ERP Cloud password.
9. Click **Test Connection**.

   When the connection has been tested correctly, the information message "Connection to [*source system name*] successful" is displayed.
10. Click **OK**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Configuring an Oracle HCM Cloud Connection

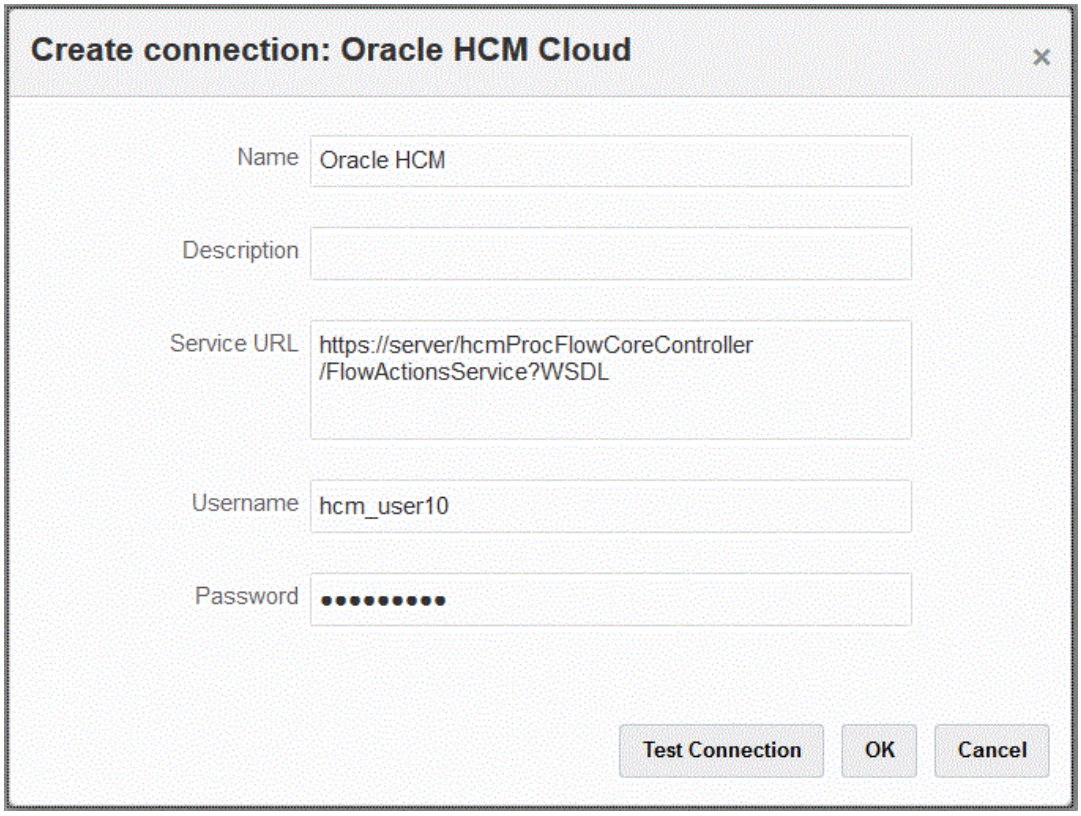
Define a Oracle Human Capital Management Cloud source connection to connect to a Oracle HCM Cloud data source.

To begin integrating Oracle HCM Cloud with the Workforce business process of Planning Modules or Strategic Workforce Planning, first create and register the source system with the type "Oracle HCM Cloud."

After the source system and connection information are specified, you initialize the source system. This process creates a target application record for each Oracle HCM Cloud extract.

To configure an Oracle HCM Cloud connection:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, from the **Actions** menu, click **Configure Connections**.
3. On the **Connections** page, click ⋯, and then click **Oracle Cloud HCM**.



4. In **Name**, enter the source system name.
5. In **Description**, enter a description of the source system.
6. In **Service URL**, enter the server information for the web services.
7. In **Username**, enter the Oracle HCM Cloud user name.
8. In **Password**, enter the Oracle HCM Cloud password.

   You must update this password anytime you change your Oracle HCM Cloud password.
9. Click **Test Connection**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

When the connection has been tested correctly, the information message "Connection to [*source system name*] successful" is displayed.

10. Click **OK**.

Configuring an Oracle NetSuite Connection

Shows the configuration links for Oracle NetSuite SuiteApps.

For information on configuring an Oracle NetSuite connection, see the following:

Table 6-1 Netsuite SuiteApp Installation and Setup References.

SuiteApp	Reference
Oracle EPM Connector	Oracle EPM Connector Installation and Setup
NSPB Sync	NSPB Sync Setup Guide
NetSuite Planning and Budgeting	Integration Guidelines
Account Reconciliation Sync	Account Reconciliation Sync SuiteApp Installation and Setup

Configuring an Oracle Autonomous Database Connection

To create an Oracle Autonomous Source Connection, see Configuring an Oracle Autonomous Database Source Connection.

Copyright © 2017, 2026, Oracle and/or its affiliates.

7

Managing Period Mappings

Describes the "default" and "explicit" types of period mappings and when to use the "global", "application" or "source" mapping options.

In your Oracle Fusion Cloud Enterprise Performance Management application, you can also use different calendars (for example, monthly, weekly, or daily), based on your application requirements (for example, different levels of periods). Because Data Integration extracts the ERP source system data to the target Cloud EPM application, it establishes the mapping relationship by defining a period mapping between the source system periods and the target Cloud EPM application periods.

Data Integration supports two types of period processing:

1. Default Period Processing
2. Explicit Period Processing

Use default period processing when the source and target applications use consistent period definitions and period naming. That is, an integration uses the Period Key and Prior Period Key defined in Data Integration to determine the source General Ledger periods mapped to each Data Integration period when the integration is run. For example, when a monthly calendar is used in both systems and the names of the periods like Jan-20 match both systems, then you use the Period Mapping Type of Default in the integration. No additional mapping is required.

Use explicit period processing when the source and target applications period definition or period naming conventions are different. Explicit period mappings enable also support additional Oracle General Ledger data sources where periods are not defined by start and end dates. Explicit mappings are set up using calendars periods in the source system option. You select the actual calendar to use for the integration from the Calendar drop-down.

In this case, you are required to define an Application Period and Source Period mapping.

You can define period mappings in three ways:

- **Global Mapping**—You define a global mapping in cases where you do not have many target applications getting data from multiple source systems with different types of source calendars. Use a global mapping to ensure that various periods are accommodated in the individual mapping. As a first step, define a global mapping.
- **Application Mapping**—If you have multiple target applications, getting data from various source systems with complex period types, you can create application mappings in addition to global mappings. When you define an application mapping, you can modify the Target Period Month as necessary.
- **Source Mapping**—Specifies source period mapping for file and adapter-based integrations.

Global Mapping—Sample Monthly Period Mapping

The following table shows how a monthly calendar from a source map to monthly periods in a target application.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> You should define global mapping at the most granular level. For example, if you have a monthly calendar and a weekly calendar, define your global mapping at the lowest level of granularity. In this case, the period keys are at the week level and you map weeks to months. You can create application mappings for the higher-level periods.

Table 7-1 Sample Monthly Period Mapping

Period Key	Prior Period Key	Period Name	Target Period Month	Target Period Quarter	Target Period Year	Target Period Day	Year Target
Jan 1 2021	Dec 1 2020	January 1, 2021	Jan	Q1			FY21
Feb 1 2021	Jan 1 2021	February 1, 2021	Feb	Q1			FY21
Mar 1 2021	Feb 1 2021	March 1, 2021	Mar	Q1			FY21
April 1 2021	March 1 2021	April 1, 2021	Apr	Q2			FY21
May 1 2021	April 1 2021	May 1, 2021	May	Q2			FY21

Global Mapping—Sample Weekly Period Mapping

The following table shows how a weekly calendar from an Enterprise Resource Planning (ERP) source system maps to monthly periods in the Cloud EPM application.

Table 7-2 Sample Weekly Period Mapping

Period Key	Prior Period Key	Period Name	Target Period Month	Target Period Quarter	Target Period Year	Target Period Day	Year Target
Jan 26 2020	Jan 19 2020	January 26, 2021	Jan	Q1			FY20
Feb 2 2020	Jan 26 2020	February 2, 2021	Feb	Q1			FY20
Feb 9 2020	Feb 2 2020	February 9, 2021	Feb	Q1			FY20
Feb 16 2020	Feb 9 2020	February 16, 2021	Feb	Q1			FY20

Application Mapping—Sample Target Application Sourcing from a Monthly Calendar Source

The following table shows a sample where the target application is sourcing from a monthly calendar. This mapping is performed on the Application Mapping tab.

Table 7-3 Sample Application Mapping—Target Application #1 with a Monthly Calendar Source

Period Key	Target Period Month	Target Period Quarter	Target Period Year	Target Period Day	Year Target
Jan 1 2020	Jan	Q1			FY20
Feb 1 2020	Feb	Q1			FY20
Mar 1 2020	Mar	Q1			FY20

Copyright © 2017, 2026, Oracle and/or its affiliates.

Application Mapping—Sample Target Application #2 Sourcing from a Weekly Calendar Source

The following table shows a sample where the target application is derived from a weekly calendar. This mapping is performed on the Application Mapping tab.

Table 7-4 Sample Application Mapping—Target Application #2 with a Weekly Calendar Source

Period Key	Target Period Month	Target Period Quarter	Target Period Year	Target Period Day	Year Target
Jan 26 2020	Jan	Q1			FY20
Feb 2 2020	Feb	Q1			FY20
Feb 9 2020	Feb	Q1			FY20
Feb 16 2020	Feb	Q1			FY20

Global Mappings

Define one global mapping to map various periods to an individual mapping.

You can define one global mapping to map various periods to the individual mapping.

To define a global mapping:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Period Mapping**.
2. Select the **Global Mapping** tab.
3. Click
   •••
   and then click **Add Mapping**.
4. In **Period Key**, specify the last day of the month of the current fiscal year to be mapped from the source. This is a date value that is stored in the database during the trial-balance load process and that becomes part of a key that identifies a set of trial-balance records.

   Specify the date in *MM/dd/yyyy* format.

   You can also select the Period Key by clicking

   and selecting the date.

   > **Note**
   >
   > The integration uses the Period Key and Prior Period Key defined in Data Integration to determine the source General Ledger periods mapped to each Data Integration period when the integration is run.

5. The **Prior Period Key** should be specified as the period key one month prior to your current period key, and you specify the format the same as the period key, for example, specify the date in *MM/dd/yyyy* format.

   Note that this feature may also be used when pulling from the Fusion GL when you want to pull multiple periods from the source into a single period in the Oracle Fusion Cloud Enterprise Performance Management. For example, if you want to pull YTD data from Fusion, you would specify the prior period key as the period key of the last fiscal period. If

Copyright © 2017, 2026, Oracle and/or its affiliates.

you are in May, and the fiscal year ends in December, and you want YTD from Fusion for May, you can set the prior period key to the date in December.

6. In **Period Name**, specify the current fiscal period key of the period.
7. In **Target Year**, specify the year value in the target application that you are mapping the period key to.

   Year value in the target application that you are mapping the period key to.
8. In the following target periods, specify the target period value to which to load data.

   Target period value options include:

   a. **Target Period - Month**; for example, August.
   b. **Target Period - Quarter**
   c. **Target Period - Year**
   d. **Target Period - Day**

   The target period options use the Frequency values defined in Category Mappings. When running an integration, the system looks at the frequency value in Category Mapping and then uses that value to query the period mapping for the target period value.

   For example, if you selected "Monthly" as the frequency value in Category Mapping, the system uses the value that is entered for Target Period – Month when deciding where to load the data. If "Quarterly" was selected as the frequency value, the system uses the value entered in Target Period – Quarter, and so on.
9. To delete a mapping, click the (**Delete**) icon.
10. Click **Save**.

Application Mappings

Define a special period mapping for a specific target application.

You can define application mappings in cases where you want to define a special period mapping for a specific target application. The mappings that you create here applies to an individual target application. If no application mappings are selected for the export target application, the system uses the global mappings defined for periods in global mappings.

To create period mappings for an application:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Period Mapping**.
2. Select the **Application Mappings** tab.
3. Click
   •••
   and then click **Add Mapping**.
4. In **Target Application**, select the target application for which to add or modify an application period mapping.
5. In **Period Key**, specify the last day of the month for the current fiscal year mapped from the source system.

   Specify the date in *MM/dd/yyyy* format.

   You can also select the Period Key by clicking and selecting the date.

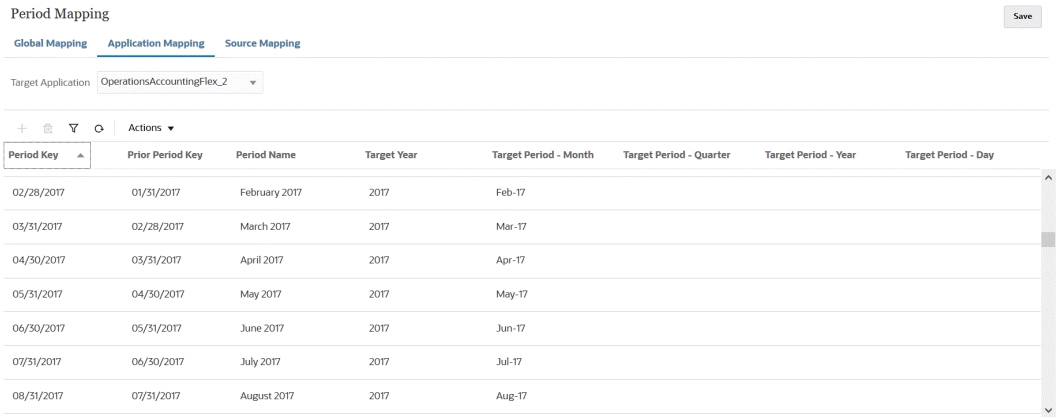
Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> The **Prior Period Key** does not apply to Account Reconciliation integrations.

> **Note**
>
> The integration uses the Period Key and Prior Period Key defined in Data Integration to determine the source General Ledger periods mapped to each Data Integration period when the integration is run.

6. In **Prior Period Key**, specify the prior date for the last day of the month of the current fiscal mapped from the source system.

   Specify the date in *MM/dd/yyyy* format.



7. In **Period Name**, specify the current fiscal period key of the period.
8. In **Target Year**, specify the year value in the target application that you are mapping the period key to.
9. In the following target periods, specify the target period value to which to load data.

   Target period value options include:

   a. **Target Period - Month**; for example, August.
   b. **Target Period - Quarter**
   c. **Target Period - Year**
   d. **Target Period - Day**

   The target period options use the Frequency values defined in Category Mappings. When running an integration, the system looks at the frequency value in Category Mapping and then uses that value to query the period mapping for the target period value.

   For example, if you selected "Monthly" as the frequency value in Category Mapping, the system uses the value that is entered for Target Period – Month when deciding where to load the data. If "Quarterly" was selected as the frequency value, the system uses the value entered in Target Period – Quarter, and so on.

10. To delete a mapping, click the (**Delete**) icon.
11. Click **Save**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Source Mappings

Define explicit period mappings to ensure that the Data Integration periods map correctly to the source system calendar periods.

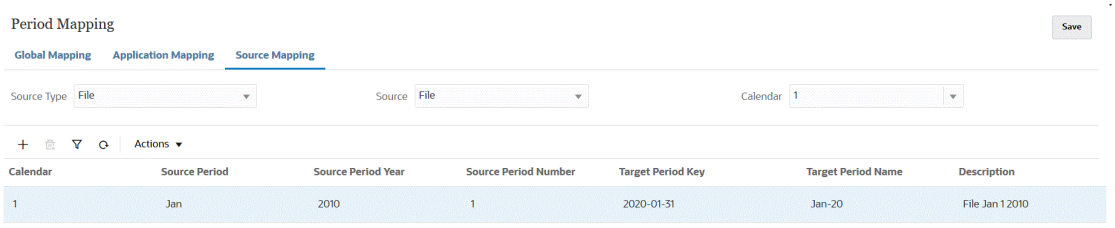
Source mappings include explicit and adjustment period mappings. You can create explicit period mappings to ensure that the Data Integration periods map correctly to the source system calendar periods. An adjustment period mapping is used only when you select the Include Adjustment Periods option when creating the integrations.

Source period mappings vary depending on the source system type. For information on source period mapping based on the source system type, see the following:

- Source Mapping for a File Source Type
- Source Mapping for a Data Source Source Type
- Source Mapping for an Oracle EPM Cloud Source Type
- Source Mapping for an Oracle ERP Cloud Source Type
- Source Mapping for an Oracle HCM Cloud Source Type

Source Mappings for a File Source Type

Source mappings can be added to a file-based integration when you need to add source period mappings between the file-based source system and a target application.



To create a source mapping for a file source type:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Period Mapping**.

   > **Note**
   >
   > To use an explicit period mapping in a file-based integration, you must include the period dimension on the **Map Dimensions** tab in the integration definition. The explicit period mapping is defined on the **Source Mapping** tab in period mappings. The **Map Dimensions** tab is also referred to as the ."import format."

2. Select the **Source Mapping** tab.
3. From the **Source Type**, select **File**.
4. In **Source**, select the name of the file.
5. From **Calendar**, select the name of the source period mapping calendar.
6. Click
   •••
   and then click **Add Mapping**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

7. In **Source Period**, enter the file-based source period date value.

   For example, you might select Jan-20.

   > **Note**
   >
   > Period names cannot include spaces if used in a batch script.

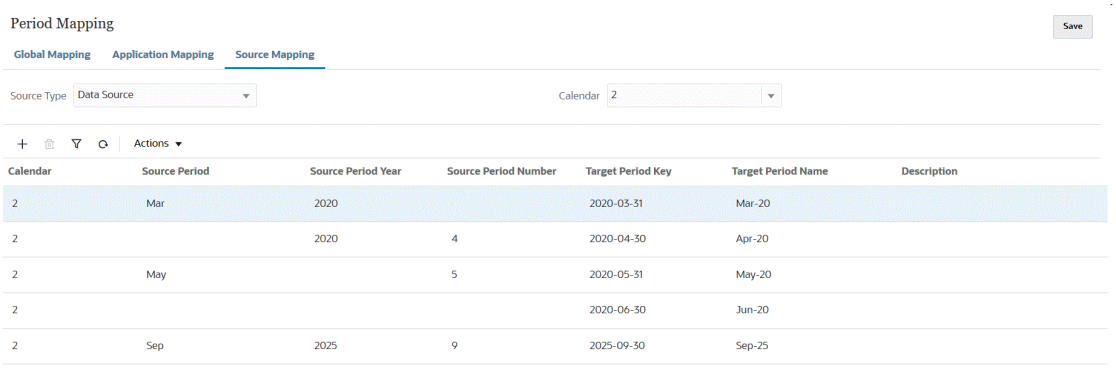
8. In **Source Period Year**, specify the year associated with the file-based source period date value.
9. In **Source Period Number**, enter the number of the period within the fiscal year.
10. In **Target Period Key**, enter the last day of the month for the current fiscal year to map to the target system.

    You can select another Target Period Key by clicking and selecting the period key from the Search and Select: Period Key page.
11. The **Target Period Name** is automatically populated by when the Target Period Key is selected.

    You can select another target period name by clicking and selecting another (Target) Period Key from the Search and Select: Period Key page. This page lists all the global mapping periods that are not used in the source mappings for the source.
12. In **Description**, enter a description of the period mapping.
13. To delete a mapping, click the (**Delete**) icon.
14. Click **Save**.

Source Mappings for a Data Source Type

You can create source period mappings for a data source type when you need to add periods between a data source application and an Oracle Fusion Cloud Enterprise Performance Management application.



To create a source mapping for a source type of Data Source:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Period Mapping**.
2. On the **Period Mapping** page, select the **Source Mapping** tab.
3. From the **Source Type** drop-down, select **Data Source**.
4. From **Calendar**, select the name of the source period mapping calendar.

Copyright © 2017, 2026, Oracle and/or its affiliates.

5. Click
   •••
   and then click **Add Mapping**.
6. From **Source Period**, enter the source period name.

   Typically, the source period name refers to a date value, such as Jan-20.

   > **Note**
   >
   > Period names cannot include spaces if used in a batch script.

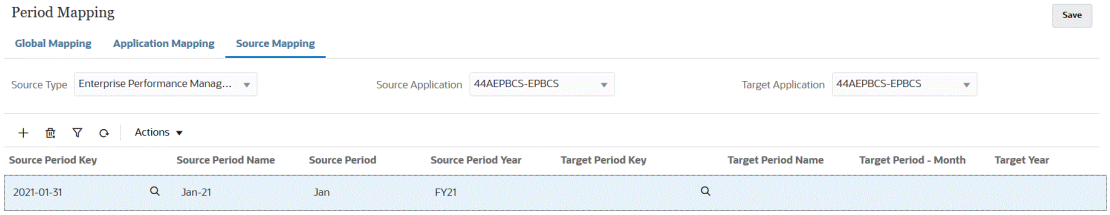
7. In **Source Period Year**, specify the year associated with the source period date value.
8. In **Source Period Number**, enter number of the period within the fiscal year.
9. In **Target Period Key**, enter the last day of the month for the current fiscal year to be mapped to the target system.

   You can select another Target Period Key by clicking and selecting the period key from the the Search and Select: Period Key page.
10. The **Target Period Name** is automatically populated by when the Target Period Key is selected.

    You can select another target period name by clicking and selecting another (Target) Period Key from the The Search and Select: Period Key page. This page lists all the global mapping periods that are not used in the source mappings for the source.
11. In **Description**, enter a description of the source period mapping.
12. To delete a mapping, click the (**Delete**) icon.
13. Click **Save**.

Source Mappings for a Cloud EPM Source Type

Source Mappings can be defined when you need to add period mappings between Oracle Fusion Cloud Enterprise Performance Management business processes.



To create a source mapping for a Cloud EPM source type:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Period Mapping**.
2. On the **Period Mapping** page, select the **Source Mapping** tab.
3. From the **Source Type** drop-down, select **Enterprise Performance Management**.
4. From **Source Application**, select the name of the source application from which to add a period mapping.

Copyright © 2017, 2026, Oracle and/or its affiliates.

5. From **Target Application**, select the name of the target application to which to add the source period mapping.
6. Click
   •••
   and then click **Add Mapping**.
7. In **Source Period Key**, click to open the Search and Select: Period Key window. Search and select the last day of the month for the current fiscal year to be mapped from the source system. The **Source Period Name**, **Source Period**, and **Source Period Year** are populated by default based on the selected **Source Period Key**.

   > **Note**
   >
   > Period names cannot include spaces if used in a batch script.

8. In **Target Period Key**, click to open the Search and Select: Period Key window. This page lists all the global mapping periods that are not used in the source mappings for the source. Specify the last day of the month for the current fiscal year to be mapped to the target system. The **Target Period Name**, **Target Period - Month**, and **Target Year** are populated by default based on the selected **Target Period Key**.
9. In **Description**, enter a description of the source period mapping.
10. To delete a mapping, click the (**Delete**) icon.
11. Click **Save**.

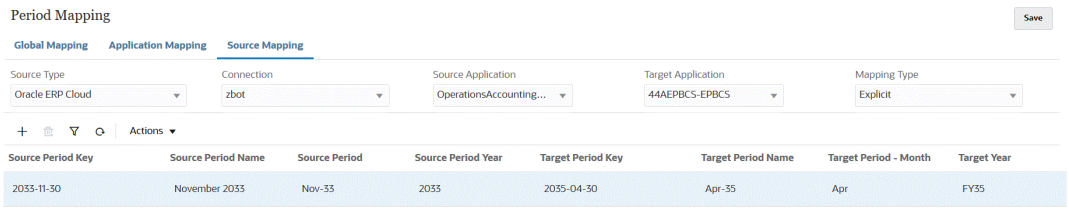
Source Mappings for an Oracle ERP Cloud Source Type

Use an Oracle ERP Cloud source type when defining source period mappings between Oracle ERP Cloud and Oracle Fusion Cloud Enterprise Performance Management applications.

For example, you may need to define explicit period mappings between the EPM Planning Projects module (Projects) and Oracle Fusion Cloud Project Management (Project Management).

To create a source mapping for an Oracle ERP Cloud source type:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Period Mapping**.
2. On the **Period Mapping** page, select the **Source Mapping** tab.
3. From the **Source Type** drop-down, select **Oracle ERP Cloud**.



4. From **Connection**, select the connection name associated with the Oracle ERP Cloud integration to use with the source period mapping.
5. From **Source Application**, select the name of the source application from which to add a period mapping.

Copyright © 2017, 2026, Oracle and/or its affiliates.

6. From **Target Application**, select the name of the target application to which to add the source period mapping.
7. From **Mapping Type**, select either **Explicit** or **Adjustment**.

   Explicit—The integration uses the Explicit period mappings defined in Data Integration to determine the source General Ledger Periods mapped to each Data Integration Period included when the integration is run. Explicit period mappings enable support of additional General Ledger data sources where periods are not defined by start and end dates.

   Adjustment—The integration uses regular period and adjustment periods. If the adjustment period does not exist, then only the regular period is processed. For more information, see Processing Oracle General Ledger Adjustment Periods in *Administering Data Management for Oracle Enterprise Performance Management Cloud*.
8. Click
   •••
   and then click **Add Mapping**.
9. In **Source Period Key**, click to open the Search and Select: Period Key window. Search and select the last day of the month for the current fiscal year to be mapped from the source system. The **Source Period Name**, **Source Period**, and **Source Period Year** are populated by default based on the selected **Source Period Key**.

   > **Note**
   >
   > Period names cannot include spaces if used in a batch script.

10. In **Target Period Key**, click to open the Search and Select: Period Key window. This page lists all the global mapping periods that are not used in the source mappings for the source. Specify the last day of the month for the current fiscal year to be mapped to the target system. The **Target Period Name**, **Target Period - Month**, and **Target Year** are populated by default based on the selected **Target Period Key**.
11. In **Description**, enter a description of the source period mapping.
12. To delete a mapping, click the (**Delete**) icon.
13. Click **Save**.

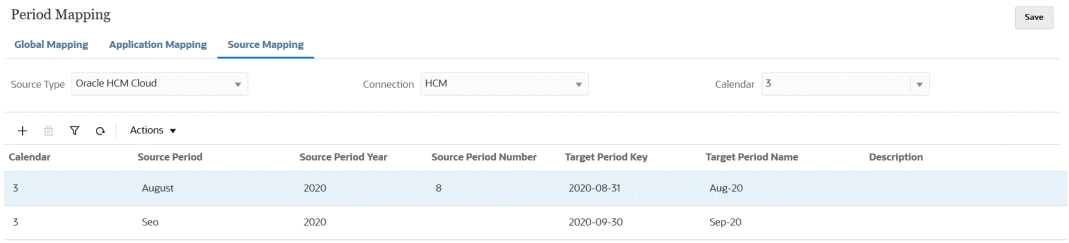
Source Mappings for an Oracle HCM Cloud Source Type

Use an Oracle Human Capital Management Cloud source type when defining source period mappings between Oracle HCM Cloud and Oracle Fusion Cloud Enterprise Performance Management applications.

To create a source mapping for an Oracle HCM Cloud source type:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Period Mapping**.
2. From the **Period Mapping** page, select the **Source Mapping** tab.
3. From the **Source Type** drop-down, select **Oracle HCM Cloud**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



4. From **Connection**, select the connection name associated with the Oracle HCM Cloud integration to use with the source period mapping.
5. From **Calendar**, select the name of the source period mapping calendar used for integrating data.
6. Click
   •••
   and then click **Add Mapping**.
7. In **Source Period Key**, click to open the Search and Select: Period Key window. Search and select the last day of the month for the current fiscal year to be mapped from the source system. The **Source Period Name**, **Source Period**, and **Source Period Year** are populated by default based on the selected **Source Period Key**.

   > **Note**
   >
   > Period names cannot include spaces if used in a batch script.

8. In **Target Period Key**, click to open the Search and Select: Period Key window. This page lists all the global mapping periods that are not used in the source mappings for the source. Specify the last day of the month for the current fiscal year to be mapped to the target system. The **Target Period Name**, **Target Period - Month**, and **Target Year** are populated by default based on the selected **Target Period Key**.
9. In **Description**, enter a description of the source period mapping.
10. To delete a mapping, click the (**Delete**) icon.
11. Click **Save**.

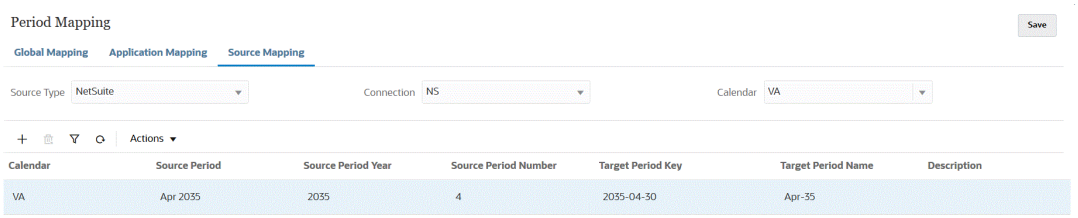
Source Mappings for an Oracle NetSuite Source Type

Use an Oracle NetSuite source type when defining source period mappings between Oracle NetSuite and Oracle Fusion Cloud Enterprise Performance Management applications.

To create a source mapping for an Oracle NetSuite source type:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Period Mapping**.
2. Select the **Source Mapping** tab.
3. From the **Source Type** drop-down, select **NetSuite**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



4. From **Connection**, select the connection name associated with the Oracle NetSuite integration to use with the source period mapping.
5. From **Calendar**, select the name of the source period mapping calendar used for integrating data.
6. Click
   •••
   and then click **Add Mapping**.
7. In **Source Period Key**, click to open the Search and Select: Period Key window. Search and select the last day of the month for the current fiscal year to be mapped from the source system. The **Source Period Name**, **Source Period**, and **Source Period Year** are populated by default based on the selected **Source Period Key**.

   > **Note**
   >
   > Period names cannot include spaces if used in a batch script.

8. In **Target Period Key**, click to open the Search and Select: Period Key window. This page lists all the global mapping periods that are not used in the source mappings for the source. Specify the last day of the month for the current fiscal year to be mapped to the target system. The **Target Period Name**, **Target Period - Month**, and **Target Year** are populated by default based on the selected **Target Period Key**.
9. In **Description**, enter a description of the source period mapping.
10. To delete a mapping, click the (**Delete**) icon.
11. Click **Save**.

Period Mapping Options

Describes the various period mapping options including: import from and export to Excel, and download and import template.

In Period Mapping, use the available period options available on the **Actions** menu to complete the following:

- Import from Excel—Import period mapping from an Excel spreadsheet.
- Export to Excel—Export period mapping to an Excel spreadsheet.
- Download Import Template—Provides a template in which you can fill out period mapping details and then upload it.
- Delete Periods—Delete all or a range of periods.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Importing Period Mappings from Excel

import period mappings from an Excel spreadsheet.

You can import period mappings from an Excel spreadsheet. Imported period mappings are populated automatically on the Period Mapping page.

To import period mappings from Excel:

1. From the **Data Integration** home page, then from **Actions** menu, and then from the **Setup** drop-down, select **Period Mapping**.
2. From the **Period Mapping** page, then from **Actions** menu, select **Import From Excel**.
3. In **Select File to Import** page, click **File Browser** and specify the name of the Excel file from which to import the period mapping.

   If the Excel spreadsheet is stored locally or on another drive, navigate to the file and then click **Upload**.
   Optional: To download an imported period mappings from an Excel spreadsheet, select the spreadsheet (with an xls or xlsx extension) on the **File Browser** page, and then click . You are prompted to open or save the spreadsheet.
4. Click **OK**.

Exporting Period Mappings to Excel

Export period mappings to an Excel spreadsheet.

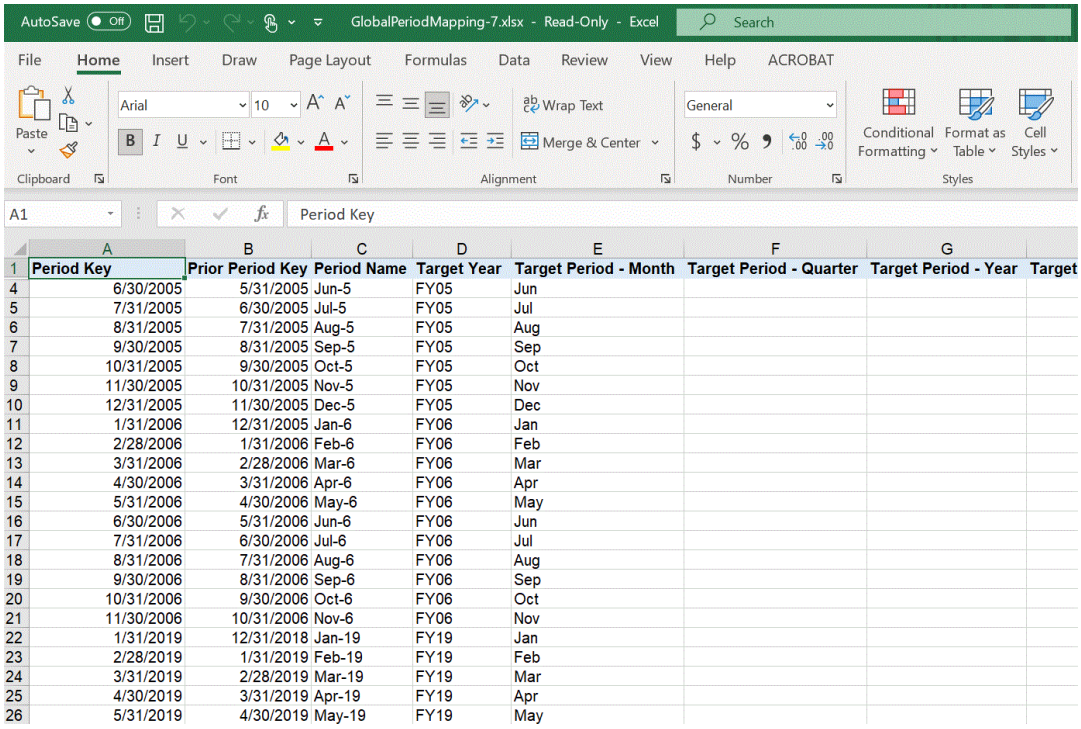
To export period mappings to Excel:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Period Mapping**.
2. On the **Period Mapping** page, from the **Actions** menu, select **Export To Excel**.

   The period mapping is exported to an Excel file and downloaded to your system.

Copyright © 2017, 2026, Oracle and/or its affiliates.

The following shows exported period mappings in an Excel spreadsheet.



	A	B	C	D	E	F	G
1	Period Key	Prior Period Key	Period Name	Target Year	Target Period - Month	Target Period - Quarter	Target Period - Year
4	6/30/2005	5/31/2005	Jun-5	FY05	Jun		
5	7/31/2005	6/30/2005	Jul-5	FY05	Jul		
6	8/31/2005	7/31/2005	Aug-5	FY05	Aug		
7	9/30/2005	8/31/2005	Sep-5	FY05	Sep		
8	10/31/2005	9/30/2005	Oct-5	FY05	Oct		
9	11/30/2005	10/31/2005	Nov-5	FY05	Nov		
10	12/31/2005	11/30/2005	Dec-5	FY05	Dec		
11	1/31/2006	12/31/2005	Jan-6	FY06	Jan		
12	2/28/2006	1/31/2006	Feb-6	FY06	Feb		
13	3/31/2006	2/28/2006	Mar-6	FY06	Mar		
14	4/30/2006	3/31/2006	Apr-6	FY06	Apr		
15	5/31/2006	4/30/2006	May-6	FY06	May		
16	6/30/2006	5/31/2006	Jun-6	FY06	Jun		
17	7/31/2006	6/30/2006	Jul-6	FY06	Jul		
18	8/31/2006	7/31/2006	Aug-6	FY06	Aug		
19	9/30/2006	8/31/2006	Sep-6	FY06	Sep		
20	10/31/2006	9/30/2006	Oct-6	FY06	Oct		
21	11/30/2006	10/31/2006	Nov-6	FY06	Nov		
22	1/31/2019	12/31/2018	Jan-19	FY19	Jan		
23	2/28/2019	1/31/2019	Feb-19	FY19	Feb		
24	3/31/2019	2/28/2019	Mar-19	FY19	Mar		
25	4/30/2019	3/31/2019	Apr-19	FY19	Apr		
26	5/31/2019	4/30/2019	May-19	FY19	May		

Downloading a Period Mapping Import Template

Download period mappings import template, complete it and then upload it.

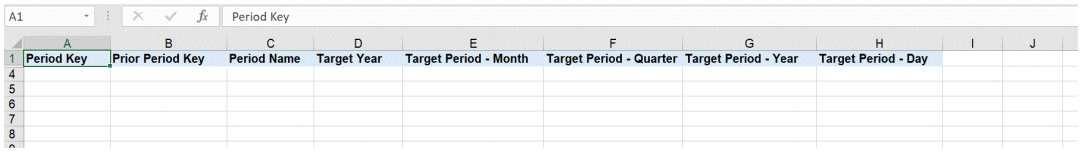
You can download a period mapping import template that you can fill out with period mapping details and then upload it.

To import period mappings from Excel:

1. From the **Data Integration** home page, then from **Actions** menu, and then from the **Setup** drop-down, select **Period Mapping**.
2. On the **Period Mapping** page, from the **Actions** menu, select **Download Import Template**.

   The import template is saved as an Excel worksheet to your system.

   The following shows a downloaded import template:



	A	B	C	D	E	F	G	H
1	Period Key	Prior Period Key	Period Name	Target Year	Target Period - Month	Target Period - Quarter	Target Period - Year	Target Period - Day
4								
5								
6								
7								
8								

3. **Optional**: When you have added period mapping detail to the import template, you can upload the Excel spreadsheet to Period Mapping using Importing Period Mappings from Excel.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Deleting Period Mappings

Delete an individual period mapping, range of period mappings, or all period mappings.

About This Task

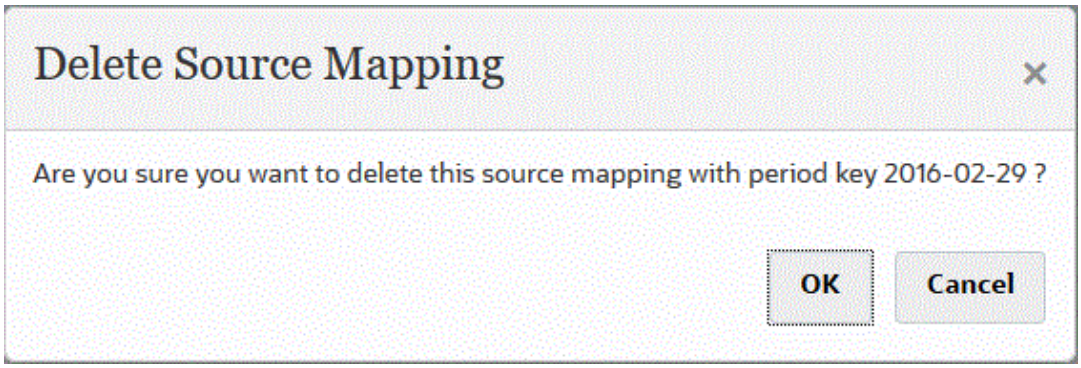
You can delete an individual period mapping, a range of period mappings, or all period mappings.

Deleting An Individual Period Mapping

To delete an individual period mapping:

1. From the **Data Integration** home page, from the **Actions** menu, then from the **Setup** actions drop-down, select **Period Mapping**.
2. Select the individual period mapping to delete from the **Global Mapping**, **Application Mapping**, or **Source Mapping** tab.
3. Click .

   The system prompts you to confirm the deletion of the period mapping.



4. Click **OK**.

Deleting A Range of Period Mappings

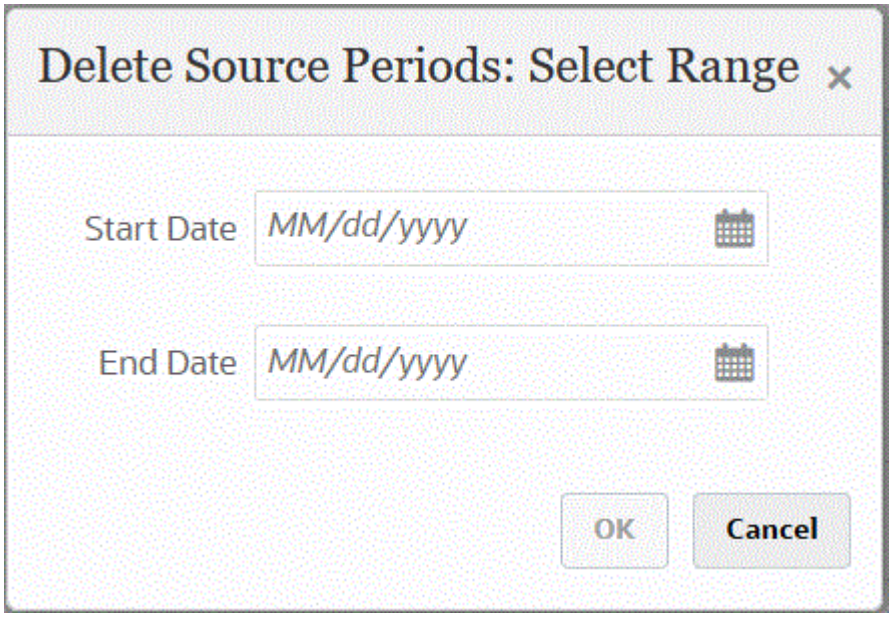
To delete a range of period mappings:

1. From the **Data Integration** home page, from the **Actions** menu, then the **Setup** actions drop-down, select **Period Mapping**.
2. Select the range of period mappings to delete from the **Global Mapping**, **Application Mapping**, or **Source Mapping** tab.
3. From **Actions** menu, then **Delete Periods**, select **Select Range**.
4. From the **Select Range** page, select the beginning date of the range to delete in the **Start Date** field and the ending date of the range to delete in the **Ending Date**.

   When entering dates, use the format MM/dd/yyyy. For example, enter **08/31/2021**.

   You can also specify dates by clicking and selecting dates from the Calendar.

Copyright © 2017, 2026, Oracle and/or its affiliates.



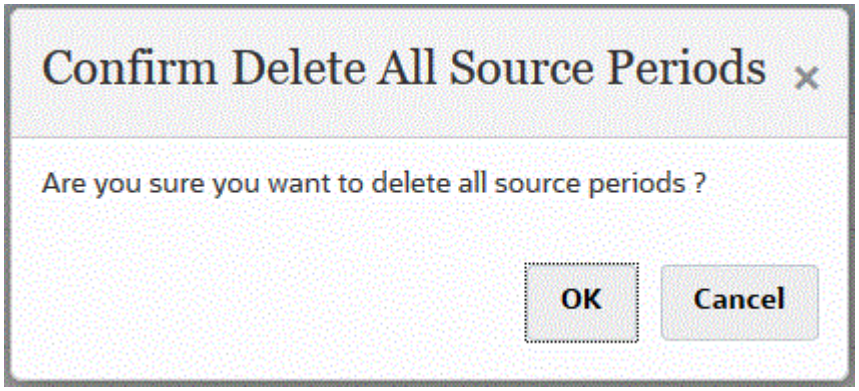
5. Click **OK**.

Deleting All Period Mappings

To delete all period mappings:

1. From the **Data Integration** home page, from the **Actions** menu, then the **Setup** actions drop-down, select **Period Mapping**.
2. Select the period mappings to delete from the **Global Mapping**, **Application Mapping**, or **Source Mapping** tab.
3. From **Actions** menu, then **Delete Periods**, select **All**.

   The Confirm Delete All Periods page displays:



4. Click **OK**.

Loading Multiple Periods for Cloud EPM or File-Based Source Systems

Load multiple periods for Oracle Fusion Cloud Enterprise Performance Management or file-based sources.

Copyright © 2017, 2026, Oracle and/or its affiliates.

For an Cloud EPM or file-based source system, Data Integration supports a "Period" as a column in a data file. If you have data for multiple periods in a single file, then you can include the year and period on each row of the data. In Import Formats, you select the source period rows of Year and Period, so that they are identified as columns in the file, and then map them to the appropriate dimension in the target system. Then you run the data load rule and select a range of dates to load. The range of dates can be based on a default or explicit period mapping type.

For example, in the following sample file, there is multiple period data, "Jan" and "Feb" in a single data file.

```
E1,100,2022,Jan,USD,100
E2,100,2022,Jan,USD,200
E3,100,2022,Feb,USD,300
E4,100,2022,Feb,USD,400
```

In another example, if you select a Jan-March period range, and the file includes: Jan, Feb, Mar, and Apr, then Data Integration only loads Jan, Feb, and Mar.

```
E1,100,2022,Jan,USD,100
E2,100,2022,Jan,USD,200
E3,100,2022,Feb,USD,300
E4,100,2022,Feb,USD,400
E4,100,2022,Mar,USD,400
E4,100,2022,Mar,USD,400
E4,100,2016,Apr,USD,400
E4,100,2016,Apr,USD,400
```

Data Integration loads the periods specified on the Run Integration page, and ignores rows in the file that don't match what you selected to load.

Copyright © 2017, 2026, Oracle and/or its affiliates.

8

Managing Category Mappings

You define category mappings for categorizing and mapping source system data to a target Scenario dimension member. For example, in a Planning application, the same source system data is stored using the Scenario dimension member "Current". In Data Integration, you can create category mapping for the scenario dimension member. Ensure that the target category value specified does exist in Planning under the Scenario Dimension.

Every integration is created with a reference to Category (POV Category). When you want to export data to a different scenario than the source application, then category mapping can be used accordingly (either Application mapping or Global Mapping). (Category mapping do not have any impact during the extract step. The extract depends on the source filter if any have been provided.)

Global Mappings

Define a global mapping to map a Scenario dimension category to an individual mapping.

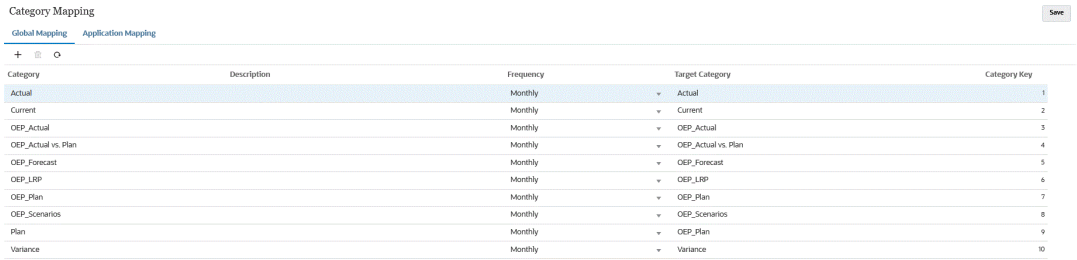
You can define one global mapping to map various Scenario dimensions to an individual mapping.

The global category mapping lets you define mappings that cross multiple applications. For example, you may have a case where a source category of an actual maps to a target of an actual in most cases. But you may have a case where you have a target application where the actual maps to current. In this case, it provides the ability to override the global mapping on an application basis.

> **Note**
>
> Avoid using special characters in names or spaces if you plan to use batch scripts. Some characters may cause issues when run from a command line.

To define a global category mapping:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Category Mapping**.
2. From the **Category Mapping** page, select **Global Mapping**.



Copyright © 2017, 2026, Oracle and/or its affiliates.

3. Click
   •••
   and then click **Add Mapping**.
4. In the **Category** drop-down, select the dimension to use for the category mapping.
5. In **Description**, enter a description of the category.
6. In **Frequency**, select the frequency of the category for each mapping.

   Frequency options include:

   - Monthly
   - Daily
   - Quarterly
   - Yearly

   The category indicates the frequency used in Period Mapping including: Target Period – Month, Target Period – Quarter, Target Period – Year, and Target Period – Day. When running an integration, the system looks at the frequency value in Category Mapping and then uses that value to query the period mapping for the target period value.

   For example, if you selected "Monthly" in the category mapping in Period Mapping, the system uses the value that is entered for Target Period –Month when deciding where to load the data. If "Quarterly" was selected in the category mapping, the system uses the value entered in Target Period – Quarter, and so on.
7. In **Target Category**, specify the target category.
8. **Optional**: In **Category Key**, specify the internal key for the category. The key can be used for debugging drill down or data load issues.
9. Enter the target category.

   For example, the target category is the Scenario dimension in the Planning application.
10. Click **Save**.
11. **Optional:** Perform these tasks:

    - To edit a mapping, select the mapping, make changes as necessary, and then click **Save**.
    - To delete a mapping, click 🗑.

Application Mappings

Define application category mappings for a target application.

To define application category mappings:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Category Mapping**.
2. In **Category Mapping**, select the **Application Mapping** tab.
3. From **Target Application**, select the target application.
4. Click
   •••
   and then click **Add Mapping**.

   A blank entry row is displayed.

Copyright © 2017, 2026, Oracle and/or its affiliates.

5. In **Category**, select the category.
6. In **Target Category**, specify the target category or click to select a dimension or dimension member.

   For example, you might specify the Scenario dimension in Planning.

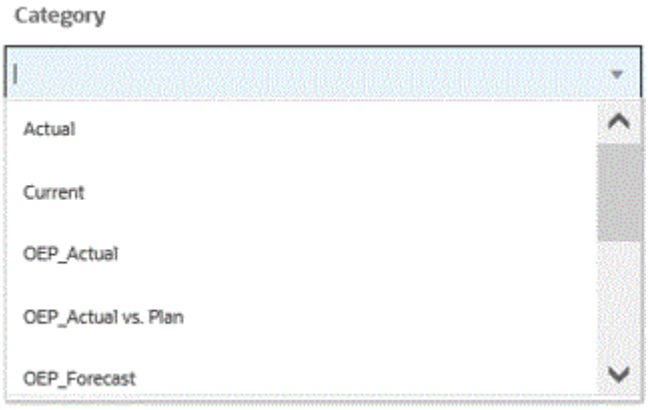
   For more information, see Selecting Members for the Target Category.
7. Click **Save**.
8. **Optional:** Perform these tasks:
   - To edit a mapping, select the mapping, and then make changes as necessary. Then, click **Save**.
   - To delete a mapping, click the (**Delete**) icon.

Selecting Members for the Target Category

You can include specific members within a dimension to include in the target category.

To select a member for a target category:

1. On the **Application Mapping** tab, from **Target Application**, select the target application.
2. Click
   ...
   and then click **Add Mapping**.
3. From the **Category** drop-down, select the dimension from which to select a member.



4. In **Target Category**, specify the dimension or dimension member or click to search on or select a dimension or dimension member.
5. On the **Select Members** page, perform the following:
   a. From **Cube**, select the plan type of the target system.

      The Cube determines the structure of available dimensions.
   b. **Optional**: In **Search Member**, enter search criteria (member name or alias only) and click **Enter**.

      The search isn't case-sensitive. You can search for a word, multiple words, or wildcard character.

Copyright © 2017, 2026, Oracle and/or its affiliates.

c. The second pane from the left shows the first level of parent dimensions available. Make selections by clicking the check box(es) next to each member in the pane to map.

   To drill down to the members in a selected dimension and display the results in the third pane, click .

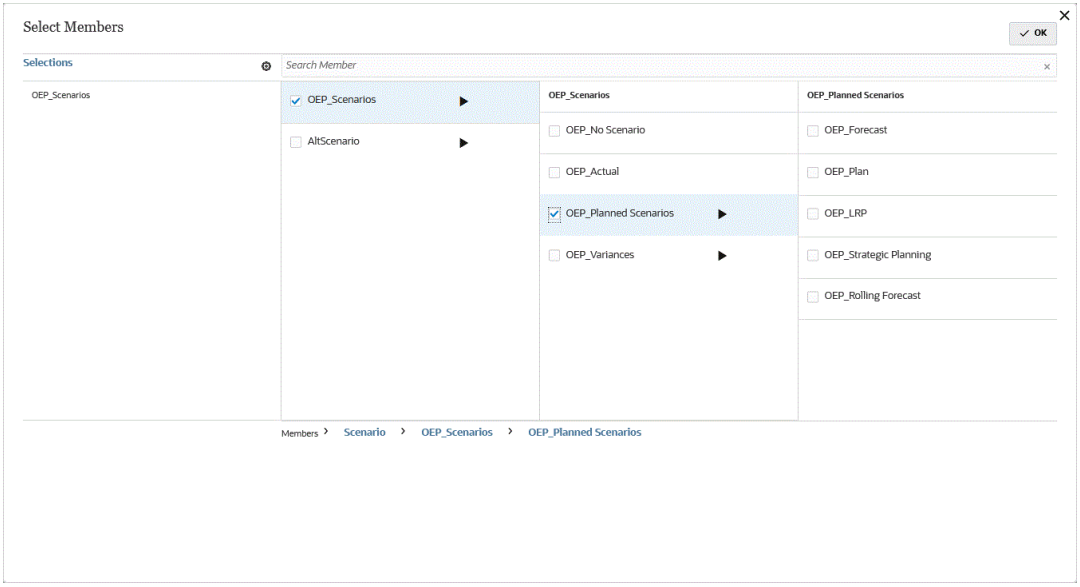
d. The third pane from the left are the members or siblings for the selected dimensions selected in step b. Make selections by clicking the check box(es) next to each member to map.

   To drill down to the members of a selected dimension and display the results in the fourth pane, click .

e. The fourth pane from the left are the members (leaf level) results from the selected member/sibling in step d. Make selections by clicking the check box(es) next to each member to map.

   Selected dimensions/members that display a check mark are moved to the **Selections** pane.

f. Click **OK**.



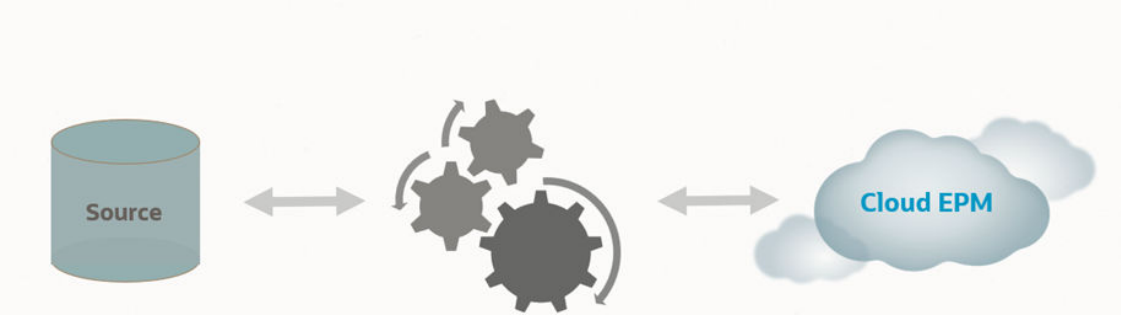
To clear selections you have made, select the dimension/member from the **Selections** pane, and click the ⚙ icon, select either: **Remove** or **Remove All**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

9

Defining a Data Integration

You define a Data Integration by extracting metadata and data from an Enterprise Resource Planning (ERP) source system (a direct integration) or data from a file, and then pushing it to an Enterprise Performance Management (EPM) target application.



File-based imports are supported for those users who do not have a direct connection to their Enterprise Resource Planning (ERP) source data but have data available from their sources in a text file. Any file, such as a delimited file, can be easily imported into a target EPM application. You can select where the account, entity, data values reside in the file, as well as which rows to skip during the data import. This feature enables a business user to easily import data from any source, and requires limited technical help, if any, when loading into a target application.

For information on file-based integrations, see Creating File-Based Integrations.

For customers who have a direct connection to other Oracle Cloud Services (out of the box or packaged integrations), define the source data, create mapping rules to translate data into the required target format, and execute and manage the periodic data loading process.

For direct integration-based sources, the import and export of data can be from:

- Oracle ERP Cloud—General Ledger. Load Actuals from General Ledger and write-back Budgets and Actual Adjustment Journals.
- Oracle ERP Cloud—Budgetary Control. Load commitments, obligations, and expenditures and write-back Budgets.
- Oracle ERP Cloud—Sub-Ledgers. Load sub-ledger transaction data.
- Oracle NetSuite—Load actuals and metadata from Oracle NetSuite.
- Oracle Human Capital Management Cloud—Load Employee attribute, salary, and other related Job information.
- Oracle E-Business Suite— Load Actuals from General Ledger.
- Oracle Peoplesoft—load Actuals from General Ledger.
- Custom on-premises data source—use an on-premises database adapter and EPM Integration Agent.
- Oracle Autonomous Database—extract source data from staging or other applications running on the Oracle Autonomous Database. You can also export data from the Oracle

Copyright © 2017, 2026, Oracle and/or its affiliates.

Fusion Cloud EPM to selected reporting data warehouses on the Oracle Autonomous Database.

- Local Oracle Fusion Cloud Enterprise Performance Management applications—Data synchronize between Input and Reporting Cubes moving data between different Input Cubes.
- Cloud EPM applications from another service instance—Data synchronizes between applications in separate business processes.

> **Note**
>
> For business process to business process integrations, both business processes must be at the same release level.

Tutorials

Tutorials provide instructions with sequenced videos and documentation to help you learn a topic.

Your Goal	Learn How
Understand the essential tasks for loading file-based data in Data Integration.	Loading File-Based Data in Data Integration for Oracle Enterprise Performance Management Cloud
Register applications, define period mappings, define category mappings, and then create and run a file-based data integration. The sections build on each other and should be completed sequentially.	Loading Data Using Data Integration

Videos

Your Goal	Watch This Video
Load data using Data Integration in Planning	Using Data Integration in Planning and Budgeting Cloud.
Overview: Using Data Integration in Planning	Overview: Using Data Integration in Planning
Integrating Oracle ERP Cloud in Cloud EPM	Overview: Integrating Oracle Cloud ERP with Oracle Cloud EPM
Set File-Based Data Integration Options and Run Integrations in Cloud EPM	Set File-Based Data Integration Options and Run Integrations in

Copyright © 2017, 2026, Oracle and/or its affiliates.

Your Goal	Watch This Video
Extract Data Using Data Integration in Oracle Financial Consolidation and Close	Extracting Data Using Data Integration in Oracle Financial Consolidation and Close

Data Integration Use Case Videos

The following is a series of Data Integration Use Case

Table 9-1 Data Integration Use Case Videos

Use Case	Watch this
Learn about the types of Data Integration tasks you can perform. With Data Integration you can perform integration tasks using an easy-to-navigate interface. This video is part of a series that showcases common data integration scenarios. In this first part of this series, see how you can use Data Integration to load a file to EPM, extract a file out of EPM, and pull balances from the Fusion GL and pull sub-ledger data from the Fusion GL.	Reviewing Data Integration Use Cases: Common Scenarios
In this second part of this series, see how you can use Data Integration to pull from a sub-ledger using a BI Publisher report and load to Account Reconciliation. You also pull directly out of an Oracle database using the EPM Integration agent.	Reviewing Data Integration Use Cases: Extracting Data from External Sources
In this third part of this series, see how you can use Data Integration to extract data from EPM, create a file with that extracted data and use SQL loader to perform a high volume data load.	Reviewing Data Integration Use Cases: Performing a High Volume Write Back
In this fourth part of this series, see how to call a Groovy rule from a pipeline.	Reviewing Data Integration Use Cases: Calling a Groovy Rule to Load Data

Creating File-Based Integrations

Shows how to create file-based integrations. This feature enables a business user to easily import data from any source by way of a file format, and requires limited technical help, if any, when loading into an Oracle Fusion Cloud EPM application.

To create a file-based integration:

1. From the Data Integration home page, click ⋯ before the **Actions** menu and then click **Create Integration**.

   The General page is displayed in Create Integration view.
2. In **Name** and **Description**, enter a name and description for the new integration.
3. In **Location**, enter a new location name, or pick an existing location to specify where to load data.

Copyright © 2017, 2026, Oracle and/or its affiliates.

A location is used to link a source and target along with the associated member mappings. You can define multiple integrations within a same location with different set of Options and Filter criteria.

If you enter a new location, you must provide the source and target. When you save the integration, the system creates the location automatically.

If you select an existing location, the system populates the source and target automatically and you cannot change it.

4. To associate the integration with Quick Mode method, tap the **Quick Mode** slider on.

   The Quick Mode method provides significant performance improvement over the standard workflow method. It supports simple transformations using import expressions and does not support member mappings. Data is not loaded to a staging table so you cannot view the data in Workbench. For example, the Level 0 extract method allows you to apply filters and select columns to include in the output file. This method is ideal for extracting large data slices from the system without reaching query processing limits.

   When you associate an integration job with the Quick Mode method and then save the job, you cannot reverse the Quick Mode association. However, the integration job can be deleted.

   For Quick Mode considerations, see Quick Mode to Export Data.

5. In **Source** page, select **File**.

6. From the **File Browser**, select the file and click **OK**.

   You can double-click the **inbox** or **outbox** folders, or any other folders to see a list of files in folders.

   You can also click **Upload** and navigate to a file to upload it.

   Optionally, you can create a file-based load in which you designate the file to be loaded at runtime. However, the column number and column name are designated only when you map the dimensions. See Mapping Dimensions.

7. From **Cube**, select the plan type of the target system.

   You can register a custom cube as a separate application with no limit to the number of custom cube applications that can be registered.

8. From **Category**, select the category for the integration.

   The categories listed are those that you created in the setup, such as "Actual." See Managing Category Mappings.

9. **Optional**: Select any applicable location attributes for the integration. For more information, see Selecting Location Attributes.

10. To map the file or change the header columns associated with the file, click **File Options**.

11. From the **File Import - File Options** page, select the **Type**, **Encoding**, **Delimiter**, and **Drill URL** fields. Then from **Use Header for Column Names**, use the arrow keys to select a header row or no header row, and then click **Next**.

    For more information on the File Import - File Options page, see Specifying File Options.

12. From the **File Import - File Column Mapping** page, modify any column headers as needed and click **Finish**.

    For more information, see Editing a File Column Header.

13. For a new location only, click (Select a Target).

14. From the **Select a Target** page, select a target application.

Copyright © 2017, 2026, Oracle and/or its affiliates.

15. Click **Save**.

    For more information about File-based integrations, see Loading File-Based Data in Data Integration for Oracle Enterprise Performance Management Cloud.

Selecting Location Attributes

Shows how to specify location attributes when creating an integration definition.

When selecting general integration options, you can add or edit the attributes or properties assigned to the location.

Location attributes include:

Location Attribute	Description
Functional Currency	Select the currency of the location. For Financial Consolidation and Close and Tax Reporting customers: To load data to actual currency rather than entity currency when the currency is fixed, set the currency in the Functional Currency field in the Location option. You can also add a Currency row in the import format and map it.
Parent Group	Select the parent assigned to the location. Parent mapping are used to share mappings with other locations. Enter mappings at the parent location, and the related locations can use the same mappings. Multiple locations can share a parent. This attribute is useful when multiple locations use one chart of accounts. Changes to a child or parent mapping table apply to all child and parent locations.
Logic Account Group	Select the logic account group to assign to the location. A logic group contains one or more logic accounts that are generated after a source file is loaded. Logic accounts are calculated accounts that are derived from the source data.
Check Rule Group	Select the check rule group to assign to the location. System administrators use check rules to enforce data integrity. A set of check rules is created within a check rule group, and the check rule group is assigned to a location. Then, after data is loaded to the target system, a check report is generated.
Check Entity Group	Select the check entity group to assign to the location. When a check entities group is assigned to the location, the check report runs for all entities that are defined in the group. If no check entities group is assigned to the location, the check report runs for each entity that was loaded to the target system. Check reports retrieve values directly from the target system, source data, or Data Integration converted data.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Location Attribute	Description
Consolidation Value	For Financial Consolidation and Close customers, in the **Consolidation Value** field, you can load the consolidation dimension associated with the FCCS_Entity Input entity, or if you are using a multi-currency application, you can load different override rates or amounts to different source members. The Consolidation dimension allows you to report on details used to perform the various stages of the consolidation process including: • FCCS_Entity Input—Amount represents the input for base entities. By way of journals, it is also available to be added for parent entities. • FCCS_Amount Override—Used for amount override loads. • FCCS_Rate Override - Used for exchange rates loads. By default, the consolidation dimension value is associated with the FCCS_Entity Input entity. Optionally, you can specify the consolidation dimension value to create and post journals to different currencies other than the entity currency • Parent Input • Contribution Input • Translated Currency Input A selected integration can load to one consolidation dimension value at a time. For Tax Reporting applications, a rates cube does not have a consolidation dimension. For this reason, leave this field blank so that you can load exchange rates for Tax Reporting applications.

Specifying File Options

You map files to specify the mapping type, drill URL, delimiter, and header row.

To map a file:

1. From the **Data Integration** home page, click ••• to the right of a file-based integration, and then select **General**.

   The General page is displayed in "Edit Integration" view.

2. Click **File Options**.

3. From the **File Options** page, then from **Type**, select the data type of the file to load.

   Available types:

   - Delimited—Load numeric data from a delimited file format.
   - Fixed Width—Not supported at this time.
   - Multi-column–Numeric—Load numeric data for multiple dimension members of a selected dimension in a single row of data. The definition of the members to load can be included in a header record in the load file, or in the import format definition.

     > **Note**
     >
     > Multi-column numeric data loads are only applicable for single year data loads.

   - Delimited–All Data Type—Load all data types from a delimited file format.

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Fixed–All Data Type—Not supported at this time.
- Multi Column–All Data Type—Load all data types for multiple dimension members of a selected dimension in a single row of data. The definition of the members to load can be included in a header record in the load file, or in the import format definition.

You can load data files that support the following data types:

- numbers
- text
- Smartlists
- date

4. In **Encoding**, specify the method for mapping bit combinations to characters for creating, storing, and displaying text.

   Each encoding has a name; for example, UTF-8. Within an encoding, each character maps to a specific bit combination; for example, in UTF-8, uppercase A maps to HEX41.

   > **Note**
   >
   > Oracle recommends you use the UTF-8 encoding and Pipe delimiters.

5. From **Delimiter**, select the character to use for delimiting columns in the output file.

   A delimited file contains one or more records set off from each other by a specified delimiter.

   Available options:

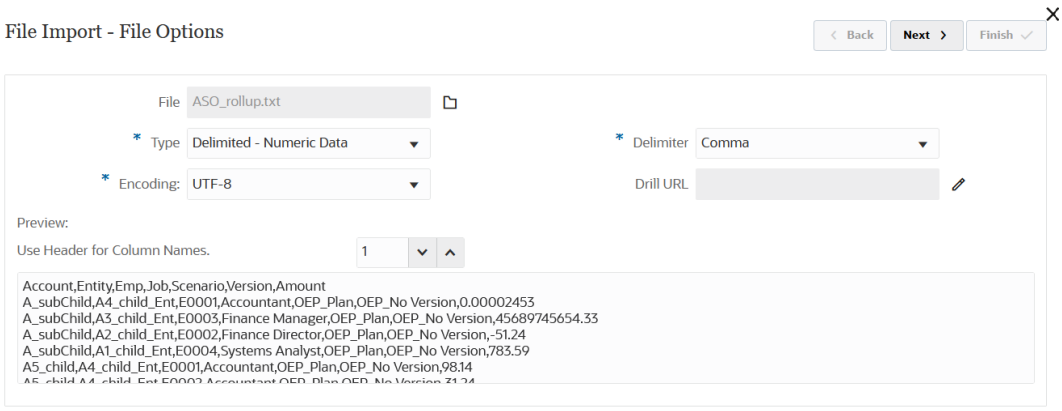
   - Comma (,)
   - Pipe (|)
   - Exclamation (!)
   - Semicolon (;)
   - Colon (:)
   - Tab
   - Tilde (~)

   > **Note**
   >
   > File encoding options are defined in the System/User Settings option in Data Management. For more information, see File Character Setting in *Administering Data Management*.

6. In **Drill URL**, specify the URL used for the drill-through.
7. From **Use Header for Column Names**, use the arrow keys to select a header row or no header row, and then click **Next**.

   To specify no header, select **0** in **Use Header for Column Names**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

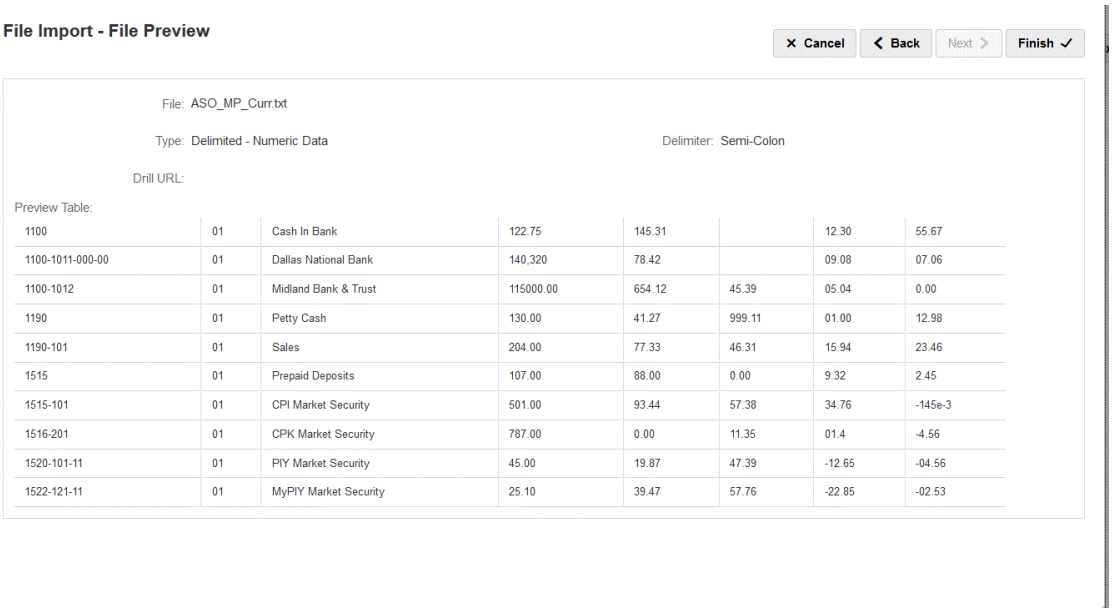


8. Click **Next**.

Previewing File Options

Describes how to view the contents of a file to be loaded.

Use the File Preview page to view the contents of the file to be loaded.



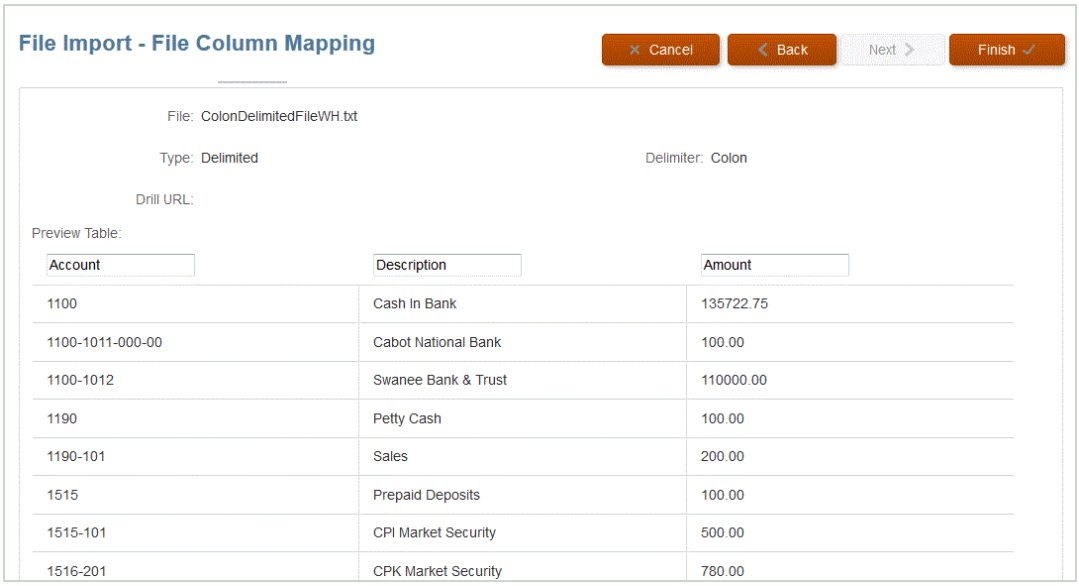
Editing a File Column Header

When the column names of a delimited file haven't been defined or you need to rename columns to map to the target application, use the File Columns Mapping page to change column names.

To edit column headers in a file:

1. After completing file mapping detail for a file-based integration on the **File Options** page, click **Next**.
2. On the **File Import - Preview** page, and then in **Preview Table**, select a column header field and edit the value as needed.

Copyright © 2017, 2026, Oracle and/or its affiliates.



3. Click **Finish**.

Using the File Browser

The File Browser provides direct access to data files found on your local and hosting server. From the File Browser, you can navigate to files stored the default `inbox` and `outbox` in Data Integration. You can also delete files, download a file to a local folder and upload files from local folders.

The `inbox` is the default directory from which source files are imported and acts, as a central repository for all ledger extract files. Because source files can be retrieved from any accessible directory, you are not required to place import files in this directory.

The `inbox` includes the batches and archivestore directories.

The `outbox` folder stores export files created by Data Integration, any error log files from a target application, drill region load files, and drill load logs from the target.

The `outbox` includes the following sub-folders:

- `archivestore`—Reserved for future use.
- `logs`—Stores logs generated by the load processes in the format EPM-APPLICATION-NAME_PROCESS-ID.log. These logs can be viewed using the Show Log link in the Process Details page of Data Integration .
- `reports`—The reports folder stores report output generated by the batch reporting process in PDF, HTML or XLS. Also includes output from reports run in offline mode.

Launch the File Browser

From the **Data Integration** home page, and then from the **Actions** menu, select **File Browser**.

Search for a File

From the **File Browser** page, then in the **Search**, enter the file to search for.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Upload a File

From the **File Browser** page, navigate to the folder to which to upload a file, and then click **Upload**. Then from your local folder, select the file to upload then click **Open**.

Delete a File

From the **File Browser** page, navigate to the folder from which to delete a file. Then select the file, and then from under **Actions**, click ••• and click **Delete**.

Download a File

From the **File Browser** page, navigate to the file to downloaded. Then from under **Actions**, click ••• and click **Download**. The file is downloaded to your system.

Using Quick Mode for File-Based Loads

Quick mode enables you to process high volume data loads that provides significant performance improvement over the standard workflow method when complex transformations are not required. Quick mode by-passes most of the steps and database tables in the workflow process. Quick Mode for file-based loads supports source and target expressions for simple transformations and Show Explicit, Like and Regular Expression mapping types.

Quick Mode - File-Based Load Considerations

Note the following considerations when using Quick Mode - file-based loads:

- Quick Mode for file-based sources supports numeric and non-numeric data types.
- Quick Mode for file-based loads does not require you to use the EPM Integration agent to load data. To source data from on-premises relational database using Quick Mode, see Quick Mode to Extract Data Process Description.
- The `ProcessMap` target expression type is exclusively used the Quick Mode for file - based loads. This target expression enables you to perform transformation using mappings specified in Map Members.

  When you select `ProcessMap` as the target expression, this tells the system to use the member mapping feature on the next step of the wizard. There are exceptions to what you can do with mapping if you select the `ProcessMap` option. Here are the exceptions:

  1. BETWEEN, MULTIDIM and #SQL mappings are not supported in quick mode if you have selected the `ProcessMap` option for the target expression.
  2. Wild card mappings don't work the same way in quick mode as they do in regular mapping. In non-quick mode processing, you can enter a mapping as follows: * -> FA_*. In this case, the mapping takes the source value and adds a prefix of **FA_*.** In quick mode, the target is set as **FA_*,** because wild cards are not supported on the target.
- The SQL target expression type cannot be applied to any dimension in a Quick Mode - file-based load.
- With Quick Mode you can define member mappings. In order to use member mapping for a dimension you need to specify the target expression type `processMap()` for the dimension. Simply defining the member mappings is not sufficient like in standard mode. You have to explicitly define the `processMap()` expression.

Copyright © 2017, 2026, Oracle and/or its affiliates.

- For mapping members, this feature supports map members using the Explicit and Like mapping types and a Regular Expression mapping type which is used to map a set of strings based on common characteristics shared by each string in the set for find and find and replace algorithms.
- When running the Quick Mode load, valid export modes are:
  - For Planning—Replace, Merge, and Accumulate
  - For Financial Consolidation and Close—Replace, Merge, and Accumulate

  > **Note**
  >
  > Dry Run is not applicable for Quick Mode loads.

  Replace is the default mode.

  No import modes are available.
- Multi-year loads using the Replace export mode is not supported at this time.
- A direct drill through to source is required when data is loaded using Quick Mode method. For more information, see Defining Direct Drill.
- You can extract non-numeric data to an EPM data file target application with Quick Mode. Perform the extraction when application is not in use because the extraction process makes the application read-only when in progress.

Quick Mode for File-based Load Process Description

To create an integration for a Quick Mode file-based load:

1. From the Data Integration home page, click ***, and then click **Create Integration**.

   The General page is displayed in Create Integration view.
2. In **Name** and **Description**, enter a name and description for the new integration.
3. In **Location**, enter a new location name, or pick an existing location to specify where to load data.

   A location is used to link a source and target along with the associated member mappings. You can define multiple integrations within a same location with different set of Options and Filter criteria. A location is primarily used to control access to integration. You can use Location Security to control access to end users.

   If you enter a new location, you must provide the source and target. When you save the integration, the system creates the location automatically.

   If you select an existing location, the system populates the source and target automatically and you cannot change it.

   For more information about the location attributes, see Selecting Location Attributes.
4. Tap the **Quick Mode** slider on.

   When you associate an integration job with the Quick Mode method and then save the job, you cannot reverse the Quick Mode association. However, the integration job can be deleted.
5. In **Source**, select **File**.

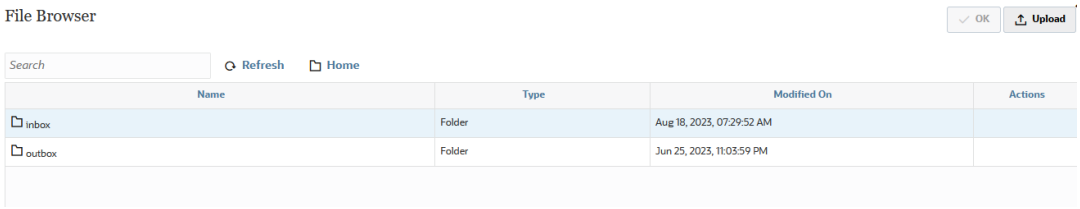
Copyright © 2017, 2026, Oracle and/or its affiliates.

Source	Type
EBS GL Balance	EBSGLBAL
File	FILE
P174052_Vision	PBCS
Payables Transactions	ERPPAYABLES
POD_Vision	PBCS

6. From the **File Browser**, select the file and click **OK**.

   Quick Mode for file-based loads support numeric and non-numeric data types.

   You can double-click the **inbox** or **outbox** folders, or any other folders to see a list of files in folders.



   You can also click **Upload** and navigate to a file to upload it.

7. From **Cube**, select the plan type of the target system.

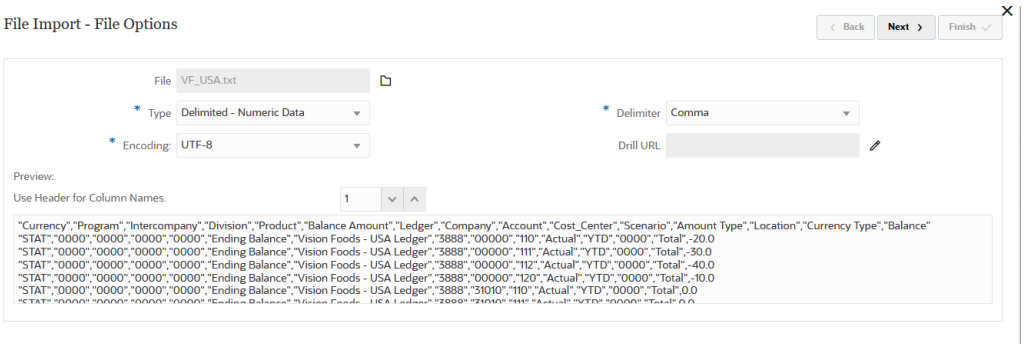
   You can register a custom cube as a separate application with no limit to the number of custom cube applications that can be registered.

8. From **Category**, select the category for the integration.

   The categories listed are those that you created in the setup, such as "Actual." See Managing Category Mappings.

9. **Optional**: To map the file or change the header columns associated with the file, from the **Create Integration** page, click **File Options**.

   a. From the **File Import - File Options** page, complete the **Type**, **Encoding**, **Delimiter**, and **Drill URL** fields. Then from **Use Header for Column Names**, use the arrow keys to select a header row or no header row, and then click **Next**.



   For more information on the File Import - File Options page, see Specifying File Options.

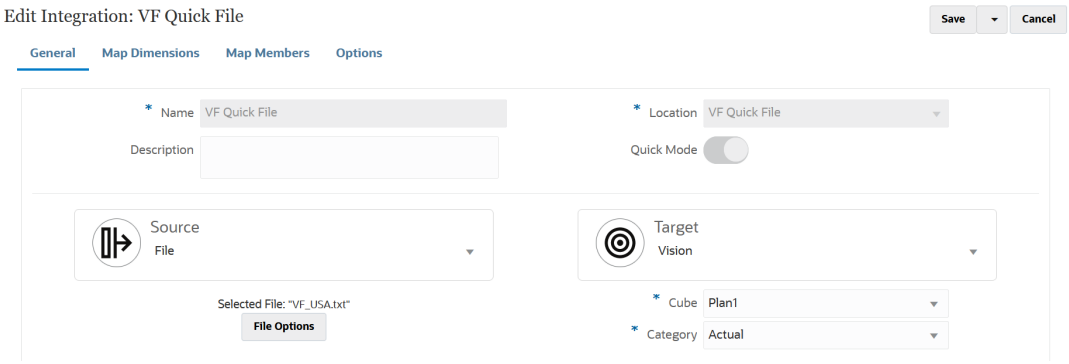
   b. From the **File Import - File Preview** page, modify any column headers as needed and click **Finish**.

   For more information, see Editing a File Column Header.

10. For a new location only, in **Target** page, select a target application.

Copyright © 2017, 2026, Oracle and/or its affiliates.

11. Click **Save**.



12. From the **Data Integration** home page, click ••• to the right of the integration, and then select **Map Dimensions**.
13. In **Import Format**, select the name of the import format to use for the integration.
14. In **Type**, select the format of the file.

    Available options:

    - **Delimited**—Load numeric data from a delimited file format.
    - **Multi-column–Numeric**—Load numeric data for multiple dimension members of a selected dimension in a single row of data. The definition of the members to load can be included in a header record in the load file, or in the import format definition.
    - **Delimited–All Data Type**—Load all data types from a delimited file format.
    - **Multi Column–All Data Type**—Load all data types for multiple dimension members of a selected dimension in a single row of data. The definition of the members to load can be included in a header record in the load file, or in the import format definition.
15. In **Delimiter**, select the character to use for delimiting columns in the output file.

    A delimited file contains one or more records set off from each other by a specified delimiter.

    - Comma (,)
    - Pipe (|)
    - Exclamation (!)
    - Semicolon (;)
    - Colon (:)
    - Tab
    - Tilde (~)
16. In the mappings grid, map the source columns in the source data-load file to the dimensions in the target application.

    The dimensions from the target application are populated automatically.

    If you select an existing import format, then the source and target are automatically mapped.

    If you are adding a new import format or editing an existing import format, complete the following:

    - In **Column**, specify the field number from the file to import.

Copyright © 2017, 2026, Oracle and/or its affiliates.

- In **Select Source Dimension**, specify the name of the source dimension to assign to the target application.

  Multiple source columns of the same dimension can be mapped to target dimensions. For example, you can map four "Account" source columns.

17. **(Optional)**: Add a source or target expression by assigning expressions that operates on values directly from the source or target.

    For more information for target expression types: see Using Target Expressions.

    > **Note**
    >
    > The `ProcessMap` target expression is exclusively used for Quick Mode for file-based loads. This target expression enables you to perform transformation using mappings specified in Map Members.

    For more information for source expression types, see Using Source Expressions

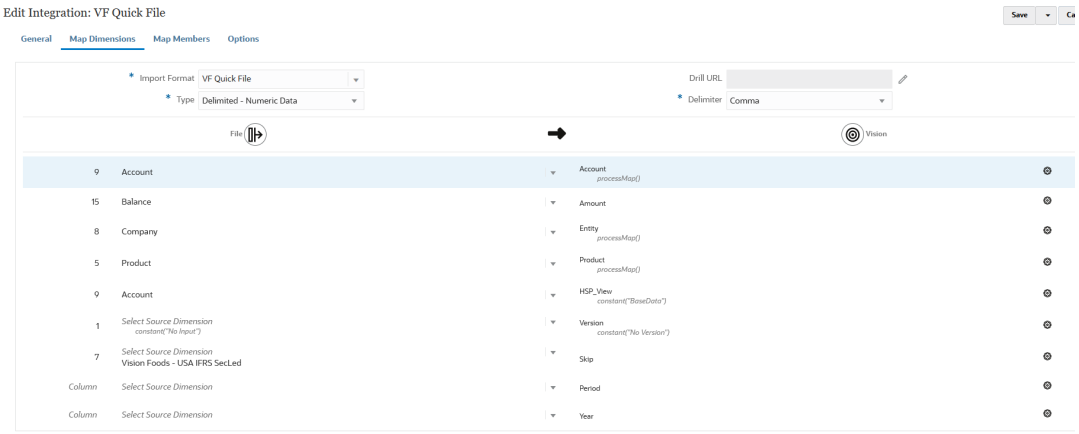
    > **Note**
    >
    > The SQL target expression type is not available for dimensions used in Quick Mode - file-based loads.

18. To use a member mapping for a dimension, you need to specify explicitly the target expression **processMap** for the dimension.

    For example, if you plan to map a member for the **Entity** dimension, you need to define the processMap expression for the Entity dimension.

    a. In the mappings grid, select the target value to which to add a target expression by clicking ⚙ to the right of the dimension, and then selecting **Edit Target Expression**.

    b. From the **Edit Target Expression** page, click **Expression Type**, then select **Process Map** and then click **OK**.



19. Click **Save**.

20. From the **Data Integration** home page, click ••• to the right of the integration, and then select **Map Members**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Map members to translate source values to valid members in each target dimension. If you have defined target expressions to derive the target members for a given dimension, then you don't have to define member mapping.

> **Note**
>
> If your data transformations can be achieved using target expressions, then its highly recommended to use it instead of mapping members. Transformation using target expression perform significantly better than member mapping for large data sets.

21. From the **Dimension** drop-down, select the dimension to which to edit or add member mappings.

> **Note**
>
> To use a member mapping for a dimension, you must have alread specified explicitly the target expression **processMap** to the dimension when mapping dimensions.

22. Click to add a new member mapping.

    You can also just edit existing mappings as needed without adding a new mapping by selecting a mapping and clicking the Edit icon .

23. From **Add Member Mapping** page, then **Source**, select the member mapping type from the **Mapping Type** drop down, and then specify the source value.

    Types of member mappings:

Type of Mapping	Description	See Also
== Explicit	Match and replace the source value exactly with the target value. Explicit mappings are one to one mappings, for example, source value "ABC" is replaced with target value "123."	Using Explicit Mappings

Copyright © 2017, 2026, Oracle and/or its affiliates.

Type of Mapping	Description	See Also
Is Like	Use special characters to match a string in the source value and map it to a target value. Like mappings can use wildcard characters in the source record only: a single asterisk (*) and question marks (?). Asterisks are placeholders for any number of characters. For example, 1190* maps the accounts 1190, 1190100, and 1190-200 to the Cash target account. Question marks are placeholders for one character. For example, the source account of 119? maps only to source accounts that contain four characters and that begin with 119.	Using Like Mappings
* Regular Expression	Regular expression that uses a sequence of characters that specifies a match pattern in text. This mapping type is available for source records only. Only Java supported regular expressions can be used. For example, if you want to apply a regular expression that ignores a row that has five digits in the source record, you can specify **.* \d{5}** and in the target, you specify **IGNORE**. Other examples: 110[1-9]{3}999 - Starts with 110 any 3 numbers ends with 999 ^[A-Z][1-5]{5} - Starts with uppercase alphabet followed by 5 numbers between 1 & 5 ^1[1-5]{5}9{3} - Starts with 1 any 5 number between 1 & 5 and ends with 999	See: • Lesson: Regular Expressions • Regular Expressions

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> When processing the source values for transformations, multiple mappings may apply to a specific source value. The order of precedence is: Explicit, Like, and then Regular Expression.

24. In **Target**, enter the target value for the dimension member name.

    You can enter an individual member as target or select a value using member selector by clicking by clicking .

> **Note**
>
> You cannot use wildcards on the target value.

25. In **Processing Order**, specify the order of the mapping.

    The processing order determines the level of precedence within a mapping type. Mappings are processed in alphabetical order by the name within a mapping type. Numbers may also be used to help with ordering. For example, if numbering by tens or hundreds, you can insert a new value between existing ones. When mappings are numbered 10, 20, and 30, you can add a mapping that starts with 25 and need not rename other mappings.

26. In **Description**, enter a description of the mapping.

27. Click **OK**.

28. Optionally, you can edit a member mapping by selecting the mapping and clicking .

    You can delete a member mapping by selecting the mapping and clicking .

29. **Run the integration**:

    a. From the **Data Integration** home page, select the integration job associated with the Quick Mode - File-based load and then click .

    b. From the **Run Integration** page, the default value for the **Mode** is **Replace**.

    c. If no period was defined on the **Options** page, then from the **Period** drop-down, select the single period from the source file from which to load data.

       If you map a period dimension and specify a target expression for period on the Map Dimension page for the integration, then the **Period** drop-down is not available for selection because the period is derived from the mapping.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Defining Periods in a Quick Mode - File-Based Load

Period processing for Quick Mode - file-based loads do not use the periods defined in the Period Mapping option. Instead, periods are handled like any other dimensions and can be derived from the header record in the source file. If there is no Period and Year header record in the source file, you are prompted for the start and end period when you run the load during execution in which case all the data is loaded to the single period. Another option is to derive the period dimension based on the source system period name from the header record using the `toPeriod` and `toYear` target expression types to a specific format. Other target expressions that can be used for the Period dimension include `substring()`, `split()`, and `map()`.

Purging Files

Whenever you load data(files) through Data Integration, three types of files are stored;

1. The file you uploaded, which is stored in the `inbox` directory.
2. The export file, which is stored in the `outbox` directory.
3. The data file, which is stored in the data directory.

All these files remain on the server and over time, disk storage space can get full. Your administrator can then delete some of these files using the Maintain Application Folder option in Data Integration. This option enables you to purges file from the `inbox`, `outbox`, and `data` folder directories. Data Integration accepts a separate Days to Keep parameter for each of the folders. When the value is not specified for a specific folder, Data Integration skips the folder.

In addition, Data Integration checks the `inbox` and `outbox` subdirectories under the respective folders and deletes any files. In the `data` subdirectory, Data Integration skips the `scripts` directory because it holds customer scripts.

For more information, see Maintain Application Folder .

Creating Direct Integrations

Shows how to define a direct connection to the Oracle Fusion Cloud EPM (out of the box or packaged integrations), map the source data to a target application, Cloud EPM application and then load the data.

For customers who have a direct connection to other Oracle Cloud Services (out of the box or packaged integrations), define the source data, create mapping rules to translate data into the required target format, and execute and manage the periodic data loading process.

To create a direct integration:

1. From the Data Integration home page, click ••• before the **Actions** menu, and then select **Create Integration**.

   The General page is displayed in Create Integration view.
2. In **Name** and **Description**, enter a name and description for the new direct integration.
3. In **Location**, enter a new location name, or pick an existing location to specify where to load data.

   A location is used to link a Source and Target along with the associated member mappings. You can define multiple integrations within a same location with different set of Options and Filter criteria. The location is primarily used to control access to an integration. You can use Location Security to control access to end users.

Copyright © 2017, 2026, Oracle and/or its affiliates.

If you enter a new location, you must provide the Source and Target. When you save the integration, Data Integration creates the location automatically.

If you select an existing location, Data Integration populates the Source and Target values automatically and you cannot change them.

4. In **Source**, select the direct integration source system.
5. In **Target**, select a target application.
6. In **Location Attributes**, specify type of data loaded by location.

   For example, you can select a functional currency such as the US dollar.

   For more information, see Defining Location Attributes.
7. From **Category**, select the category that categorize and maps source system data to a target Cloud EPM Scenario dimension member.

   For more information, see Using Category Mappings.
8. Click **Save**.

Mapping Dimensions

You map the source data to the target dimension in the application. In addition, you can define simple transformation rules using expressions.

When mapping dimension for a file-based integration, if the source file has a header record or you defined column headings in the preview section, select source columns for each dimension. Data Integration automatically determines the column number and populate the column number. You can also manually enter the column number.

For direct integrations, you can pick the dimensions from the Source Application and map to the dimension in the Target Applications.

For both file-based and direct integration sources, you can also apply source and target expressions that operate on source and target dimensions.

Watch this tutorial for information on mapping dimensions and members: Mapping Dimensions and Members for File-Based Data Integrations in Enterprise Planning Cloud.

Creating the Dimension Maps

To map dimensions:

1. From the **Data Integration** home page, click ••• to the right of the integration, and then select **Map Dimensions**.
2. From **Import Format**, select the name of the import format to use for the integration.

   You can also add a user defined import format name.
3. **File-Based Sources Only**: From **Type**, select the format of the file.

   Available options:

   - **Delimited**—Load numeric data from a delimited file format.
   - **Multi-column–Numeric**—Load numeric data for multiple dimension members of a selected dimension in a single row of data. The definition of the members to load can be included in a header record in the load file, or in the import format definition.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> You can't use a multi-column format type when importing a subset of data or metadata from your on-premises data sources and then loading it directly to the Oracle Enterprise Performance Management Cloud using the EPM Integration Agent. Instead, you need to pivot every amount into a single row. For more information on pivoting dimensions, see Data Export Target Application Options.

- **Delimited–All Data Type**—Load all data types from a delimited file format.
- **Multi Column–All Data Type**—Load all data types for multiple dimension members of a selected dimension in a single row of data. The definition of the members to load can be included in a header record in the load file, or in the import format definition.

> **Note**
>
> At this time, the Fixed Width – Numeric Data and Fixed Width – All Data types are not supported.

4. In **Drill URL**, specify the URL used for the drill-through.

> **Note**
>
> Not applicable for Oracle Fusion Cloud Enterprise Performance Management and Oracle ERP Cloud GL data sources.

5. **File-Based Sources Only**: From **Delimiter**, select the character to use for delimiting columns in the output file.

   A delimited file contains one or more records set off from each other by a specified delimiter.

   Available options:

   - Comma (,)
   - Pipe (|)
   - Exclamation (!)
   - Semicolon (;)
   - Colon (:)
   - Tab
   - Tilde (~)

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> When importing file-based sources, a tab delimiter cannot be used to delimit columns in the following output files:
>
> - Quick Mode method file
> - Data Synchronization (all cases)
> - Oracle Autonomous Database sources
> - EPM Integration Agent sources

> **Note**
>
> If a source application contains members with a comma, you can use a different delimiter (excluding the TAB delimiter) in the import format definition to process the export. Note the output file is created using the column delimiter option value defined in Data Exchange Target Application Options.

6. In the mappings grid, map the source columns in the source data-load file to the dimensions in the target application.

   The dimensions from the target application are populated automatically.

   If the import format has already been defined for the file, then the source and target columns are mapped automatically.

   If you are adding a new import format or editing an existing import format, complete the following:

   - In **Column**, specify the field number from the file to import.
   - In **Select Source Dimension**, specify the name of the source dimension to assign to the target application.

     Multiple source columns of the same dimension can be mapped to target dimensions. For example, you can map four "Account" source columns.
   - Add a source or target expression: assign an expression that operates on values directly from the source or target.

     See Using Source Expressions and Using Target Expressions.

7. **Optional**: For a comma delimited file, select an additional row to map in the import format by clicking to the right of a row and selecting the row to add from the drop-down.

   Available rows:

   - Source Period
     - Year
     - Period
     - Period Number
   - Currency
   - Attribute
   - Description
   - Dimension Row

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Account
- Version
- Entity
- View

Import Definition Option	Description
Skip	The skip option is used to indicate rows in the input file that should be skipped. For example, rows with no data, negative numbers, or for specific accounts. The specification for a skip row is defined in the same way as that for a data row, and the system looks for the exact textual match for text entered in the expression field in the indicated location in the input file. Data Integration automatically skips rows in the input file that have "spaces" and "non-numeric" characters in the amount location, so a skip specification is only needed when non-amount data is present in a row of the input file in the same location as the amount. For example, the input file may contain a label named "date." Add an entry for a skip row to indicate the starting column of the text "date," the length of the text, and the exact text to match. The Skip row option is available for both fixed and delimited file types.
Attribute	You can import up to 40 attribute fields from a file or other data sources. For a file, you specify the location of the input field and for other data sources you can specify source dimension from the Source Application. You can also plug a fixed value using expression. The attribute fields are generally used to help compose a drill-through URL or for history or documentation needs. **Note** If you integrate a Financial Consolidation and Close or Tax Reporting source with an explicit period mapping type, the system stores the mapping year (SRCYEAR) and mapping period (SRCPERIOD) in the ATTR2 column and year in ATTR3 columns. For this reason, when importing data from Financial Consolidation and Close or Tax Reporting, attribute columns ATTR2 and ATTR3 should not be used for any other dimension mappings. Similarly, when you map a Movement source attribute to any target dimension, the system automatically creates another map for mapping the Movement to the ATTR1 column.
Description	You can import two description columns and load these columns in the same way as the attribute columns. You can specify the location in the input row that contains a description or specify an explicit value by entering it in the expression field in the mapping table.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Import Definition Option	Description
Currency	Data Integration supports the ability to load data that is of a currency different from the default currency of the selected location. This option enables you to specify the location in the input line that specifies the currency for the related amount field. For the file import format, specify a currency on each row of data, or make sure that a currency is specified in the location that uses the selected import format. **Note** You may encounter issues with loading data if the currency is not specified correctly.
Source Period	"Period" dimensions are supported as columns in a data file. If you have data for multiple periods in a single file, then you can include the year and period on each row of the data file that gets loaded to the target application. You load a period as a column from a data file by defining the load definition through the import format, and then selecting the source period when executing the integration.
Dimension Row	Data Integration supports multiple entries for a dimension in the import format when the dimension specification is spread between multiple locations on the same line. This feature enables you to concatenate fields for file-based data. To use this option, select the dimension, start and end positions, and the expression.

8. **Optional**: To duplicate a row in the import format, click to the right of the row to duplicate, and click **Duplicate** from the drop-down.
9. **Optional**: To delete a row from the import format, click to the right of the row to delete, and click **Delete** from the drop-down.
10. Click **Save**.

Using Target Expressions

When importing data, you can apply target expressions to the mapped dimensions. Target expressions enable you to transform the source value read from the source to the target dimension values to be loaded to target application. These expressions can be used instead of member mappings for performing simple data transformations. For large data sets using import expressions, the transformation improves data load performance.

Supported target expression types:

- Copy Source Value
- Prefix
- Suffix
- Map
- Substring
- Replace
- Default

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Rtrim
- Ltrim
- Rpad
- Lpad
- Constant
- Round
  The Round expression is only available for the Amount dimension.
- Conditional
- Split
- SQL
- Upper
- Lower
- Process Map

> **Note**
>
> Except for the Round expression type, target expression types cannot be applied to an Amount dimension or the Attribute, Currency, and Skip rows.

To assign a target expression:

1. From the **Data Integration** home page, click to the right of the integration, and then select **Map Dimensions**.
2. In the mappings grid, select the target value to which to add a target expression by clicking to the right of the dimension, and then selecting **Edit Target Expression**.
3. From the **Edit Target Expression** page, click **Expression Type**, and then select a target expression.
4. From the **Edit Target Expression** screen, click **Expression Type**, and then select a target expression.
5. Select all parameters associated with the target expression and click **OK**.

   The parameters shown on the page depend on the expression type.

   To remove a target expression type, click .

Copy Source Value

Use the Copy Source expression type to copy a source value to the target.

The parameters are: `copysource()`.

Prefix

Use the Prefix expression type to add a fixed prefix to a string.

The parameter is `prefix`.

Copyright © 2017, 2026, Oracle and/or its affiliates.

For example, to specify prefix string LE01 in the parameter, use the resulting expression `prefix("LE01")`.

Map

Use the Map expression to map a source value to target value.

> **Note**
>
> The Map expression type is available only for Quick Mode - file-based target expressions.

The parameter is `(null,)`.

For example, if you want to map source periods: Jan, Feb, Mar to target periods: P1, P2, P3, then the expression is: `map(PERIOD,Jan:P1|Feb:P2|Mar:P3)`. You can use a dimension name or one of the expressions as the first parameter.

Suffix

Use the Suffix expression type to add a fixed suffix to a string.

The parameters are: `suffix("")`. For example, to add the suffix "East" to a customer code dimension, specify: `suffix("East")` as the expression.

Substring

Use the Substring expression type to extract and return characters from a string based on a character position (starting position) and substring length (number of characters to extract).

Use parameters that follow this form: `substr(Dimension, position, length)`. For example, if the source Entity number is "031010," to return the substring starting from the second character position with a substring length of 5, specify: `substr(Entity, 2, 5)` as the expression. In this example, the expression returns: 31010.

Replace

Use the Replace expression type to search and replace a string with a specific replacement value. (The replacement value can be characters within the string.)

Use parameters that follow this form: `replace(Dimension, "x","0")`. For example, if an account number is "123x456x" and you want to replace each "x" with a "0", specify: `replace(ACCOUNT , "x","0")` as the expression. In this example, the result is: 12304560.

Default

Use the Default expression type to assign the default value when the source is blank. Otherwise, it assigns the source value.

Use parameters that follow this form: `default(Dimension, "Default Value")`. For example, to assign the default value "Working" to the user-defined "Version" dimension, specify: `default(UD1, "Working")` as the expression. In this example, the result is: Working.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Rtrim

Use the Rtrim expression type to trim trailing characters from a string (on the right side).

Use parameters that follow this form: `rtrim(Dimension,"0")` where you specify both the dimension and the character to trim. For example, to trim all trailing "0 " characters when the source is "123000", specify: `rtrim(Dimension, "0")` as the expression. In this example, the result is: 123.

Ltrim

Use the Ltrim expression type to trim leading characters from a string (on the left side).

Use parameters that follow this form: `ltrim(Dimension,"0")` where you specify both the dimension and the character to trim. For example, to trim all leading "0" characters when the source is "000123", specify: `ltrim(Dimension, "0")` as the expression. In this example, the result is: 123.

Rpad

Use the Rpad expression type to pad the right side of a string with a specific set of characters up to a given length. When you specify a value for a length that is shorter than the length of the expression, then Rpad truncates the expression to the specified length. A space is the default padding character when no character to pad is specified.

Use parameters that follow this form: `rpad(Dimension, length, "characters to pad")`. For example, to pad the account value of 1234 on the right side with zeros up to six characters in length, specify: `rpad(Dimension, 6, "0")` as the expression. In this example, the result is 123400.

Lpad

Use the Lpad expression type to pad the left side of a string with a specific set of characters up to a given length. When you specify a value for a length that is shorter than the length of the expression, then Lpad truncates the expression to the specified length. A space is the default padding character when no character to pad is specified.

Use parameters that follow this form: `lpad(Dimension, length, "characters to pad")`. For example, to pad the account value of 1234 on the left side with zeros up to five characters in length, specify: `lpad(Dimension, 5, "0")` as the expression. In this example, the result is: 01234.

Constant

Use the Constant expression type to assign a constant value to the column. For target columns, the constant applies, regardless of the source column.

The parameters are: `constant("")`. For example, to assign the value "P_001" to each row in the Product column, specify: `constant("P_001")`. In the example, the result is: P_001.

Round

Use Round expression type to round a number to a specified decimal length or precision value in the Amount dimension target expression.

A precision value from -12 up to 12 can be specified in the Round expression.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Use parameters that follow this form: `round(AMOUNT,<decimal_precision>)`

In this example, the Round expression `round(AMOUNT,1)` rounds to 1 decimal precision:

Amount Value If:	Result
10.18	10.2
10.14	10.1
10.15	10.2

In this example, the Round expression `round(AMOUNT,1)` rounds to 0 decimal precision value:

Amount Value If:	Result
10.18	10
10.5	11
10.491	10

A Round precision value can be less than zero (for example, -2). When the round precision is less than zero, than the rounding occurs to the left of the decimal instead of the right.

For example, if you use a Round expression of -2 when the Amount is 574.194, then the results is 600. (`round(574.193,-2)`.

Conditional

Use a Conditional expression type to specify an if-then-else statement, which returns a value when a condition evaluates to true or returns a different value when the condition evaluates to false.

The parameters are: `condition=()`. For example, to return "Cash" when the account value is "1100", or "ShortTermRec " when the account value is 1300-101, or "AccruedTax" when the account value is 1300-102, or "No Member" when the account value is null or equals "", use the following expression:

```
if (ACCOUNT == "1100") return "Cash"
else if (ACCOUNT == "1300-1else return Liability 01") return "ShortTermRec"
else if (ACCOUNT == "1300-102") return "AccruedTax"
// if then with or/and
if (ACCOUNT == null or ACCOUNT == "") return "No Member"
```

In the "Conditional" source expression, users are able to use the term RECORD to reference the entire input line for the expression instead of just the selected dimension. For example:

For example:

When the sample input file is:

```
account,entity,icp,100
```

and the sample expression is:

```
if (split(RECORD,",",3)=="icp") return "icp 2022"
```


Copyright © 2017, 2026, Oracle and/or its affiliates.

The resulting source value would be set to: "icp 2022". Once the source has been set using the expression, then a target expression or mapping may be used to transform the source as needed.

Split

Use the Split expression type to split the source value based on a delimiter and return the *n* value after splitting the value. This expression type is useful for splitting segment values from strings

Use parameters that follow this form: `split(Dimension, "delimiter", component number)`, where you select the character delimiter to a separates string, and the component of the string to return. For example, if the source account number is 110-20-312300-500 and you want to see the third component of the string, specify: `split(ACCOUNT, "-", 3)`. In this example, the result is: 312300.

SQL

Use the SQL expression type to assign any SQL expression that can be used in a SQL INSERT statement for a value. To use a source value, enclose the value inside $ $. For example, if you want to reference UD1, specify it as $UD1$. Use only one source value in the source expression.

> **Note**
>
> You cannot use dimension name such as $MyAccount$. You can reference only dimensions that are used in the import format, and not any column in the TDATASEG table. (You can find the names of the columns for dimensions in the Data Table Column Name field on the Application Details page.)

The parameters are `sql("")`. For example, to return the value "S1" when the UD4 source value is "031010," or otherwise return the value "S2", specify the SQL expression: `sql("CASE WHEN $UD4$ = '031010' THEN 'S1' ELSE 'S2' END")`.

toPeriod

Use the toPeriod target expression type to derive the Period dimension member based on the source system period name.

Use the Java simple date format to provide the format of the Source Period Name. For example, if the source period is Jan-20, then to derive the Period dimension name, use the expression `toPeriod(PERIOD, "MMM-yy", "Mon")`.

toYear

Use the toYear target expression to derive Year dimension member based on the source system period name.

Use Java simple date format to provide format of the Source Period Name. For example, if the source period is Jan-20 then to derive the Year dimension name, use the expression `toYear(YEAR, "MMM-yy", "FY+YY")`

Upper

Use the Upper expression type to convert the column to uppercase.

Copyright © 2017, 2026, Oracle and/or its affiliates.

The parameter is (), for example, `upper()`.

Lower

Use the Lower expression type to convert the column to lowercase.

The parameter is (), for example, `lower()`.

Process Map

Use the Process Map expression to explicitly designate a dimension as a member to be mapped.

The parameter is processMap().

> **Note**
>
> The Process Map expression type is available only for Quick Mode - file-based target expressions.

Using Source Expressions

When importing data, you can apply source expressions to source dimensions. Source expressions can be used to enhance and transform the source value read from the file. For example, use a Rpad expression type to pad (add extra characters) to the right of a source dimension value to ensure a specific length.

For an "Amount" source dimension, the following source expression types are available:

- Fill: See Converting from European to U.S. Notation (Fill).
- DRCRSplit: See Displaying Debit and Credit Columns (DRCRSplit).
- Sign: See Using Nonstandard Numeric Sign Conventions (Sign).
- Factor: See Multiplying by Whole-Number and Decimal Factors (Factor).
- NZP: See Disabling Zero Suppression (NZP).
- Driver: See Driver

For information about applying source expressions to the Amount dimension, see Using Source Expressions for the Amount Dimension.

The following source expression types are available for non-Amount source dimensions:

- Prefix
- Suffix
- Concat
- Substring
- Replace
- Default
- Rtrim
- Ltrim
- Rpad

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Lpad
- Constant
- Conditional
- Split

> **Note**
>
> All Target Expressions except SQL are available for the Source.

To assign a source expression:

1. From the **Data Integration** home page, click ••• to the right of the integration, and then select **Map Dimensions**.
2. From **Import Format**, select the name of the import format to use for the integration.

   You can also add a user-defined import format name.
3. **File-Based Sources Only**. **Optional**: From **Type**, select the format of the file.

   For more information on file formats, see Creating the Dimension Maps.
4. In the mappings grid, select the source value to which to add a source expression by clicking to the right of the dimension, and then selecting **Edit Source Expression**.
5. From the **Edit Source Expression** screen, click **Expression Type**, and then select a source expression.

   > **Note**
   >
   > The source expression types that can be applied to the Amount dimension include: Fill, DRCRSplit, Sign, Factor, and NZP. Multiple source expressions can also be applied to the Amount dimension. The Amount dimension does not accept Constant, Lpad, and Rpad expression types.
6. From the **Edit Source Expression** screen, select all parameters associated with the source expression, and click **OK**.

   The parameters shown on the page depend on the expression type.

   To remove a source expression type, click .

Using Source Expressions for the Amount Dimension

For an "Amount" source dimension, the following source expression types are available:

- Fill–See Converting from European to U.S. Notation (Fill).
- DRCRSplit–See Displaying Debit and Credit Columns (DRCRSplit).
- Sign–See Using Nonstandard Numeric Sign Conventions (Sign).
- Factor–See Multiplying by Whole-Number and Decimal Factors (Factor).
- NZP–See Disabling Zero Suppression (NZP).
- Driver-See Driver.

Copyright © 2017, 2026, Oracle and/or its affiliates.

For information on applying source expression types, see Using Source Expressions.

Converting from European to U.S. Notation (Fill)

Use the Fill expression type to fill an Amount dimension with a key word to convert number formats. Typically, Fill is used with the parameter `fill=EuroToUS` to trigger a number format conversion from (.,) to (,.) format.

The parameters are `fill=`. For example, specify: `fill=EuroToUS` to convert 10.000,00 to 10,000.00.

Using Nonstandard Numeric Sign Conventions (Sign)

Use the Sign expression type to manage nonstandard numeric sign conventions. Numbers with leading and trailing minus signs and numbers within parentheses are interpreted as negative numbers. You can also use other leading and trailing characters to indicate negative numbers.

Numbers within <> are also treated as negative. For example, if you specify (100.00) and <100.00> both are treated as negative numbers.

If positive numbers are unsigned (`1,000.00`), and negative numbers are followed by `CR` (`1,000.00CR`), the expression is `Sign=,CR`.

Use parameters that follow this form: `Sign=[Positive String],[Negative String]`.For example, if positive numbers are followed by `DR` (`1,000.00DR`), and negative numbers are followed by `CR` (`1,000.00CR`), specify: `Sign=DR,CR`.

Multiplying by Whole-Number and Decimal Factors (Factor)

Use the Factor expression type to multiply a source file amount by a user-defined value. This type of expression lets you scale data in the file by any numeric factor. Using this expression, you can double or halve data.

Use parameters that follow this form: `Factor=`, where you specify a user-defined whole or decimal number by which to multiply your data. For example, use `Factor=1000` to convert the amount from 12500 to 12500000.

Displaying Debit and Credit Columns (DRCRSplit)

Use the DRCRSplit expression type to parse split numeric columns for fixed format text files. By default, numeric values in Amount fields are considered debits. However, you can position debit values on the left and credit values on the right.

To use this expression, you specify the start of the Debit (for example, column 56) and the length of the column to the end of the Credit column (for example, 40 characters). The midpoint might be 20, saying that numbers to the left of 20 are Debits and to the right are Credits.

Use parameters that follow this form: `drcrsplit=Mid Point of the DR and CR columns`. For example, use the expression: `drcrsplit=16` to specify that numbers left of the midpoint (16) are a debit and amount to right of the midpoint (16) are a credit.

When the file is imported, credit amounts are assigned negative signs (and thus are interpreted as positive), and debit amounts are unchanged (and thus are interpreted as negative).

Copyright © 2017, 2026, Oracle and/or its affiliates.

Disabling Zero Suppression (NZP)

The `NZP` expression is used to disable zero suppression during the data-load process. By default, the system bypasses accounts in the trial balance that have zero balances. In certain circumstances, you may want to load all accounts, to ensure that values that should be zero are replaced. You enter `NZP` in the Expression column of the Amount field to disable zero suppression.

The parameter is `NZP`.

> **Note**
>
> Blank rows cannot be processed in Data Integration. You need to replace the blank with "0" in your source file and then use the NZP expression in the Import format for the Amount column to load 0s to your target.

Driver

For file-based source only integrations when the **Type** is selected as "Multi-Column - All Data Type " in Map Dimensions, use the Driver source expression to load all data types for multiple dimension members of a selected dimension in a single row of data. The definition of the members to load can be included in a header record in the load file, or in the import format definition. Supported data types are: Numeric, Text, Smartlist, a Date. A driver dimension member can be assigned in the integration (as well as the header record or import format (member expression)). This source expression is only displayed in the integration when the members are not specified in the import format, or if the header row is not specified in the import format.

The parameter is `null,"", "","",""`. For example, specify: **Driver=Account;HeaderRow=2;Column=3|5,8;** to define the column dimension as "Account"; import columns 3 and 5 through 8; and use header row 2 to determine values for Account.

In another example, you could specify **Driver=Account;Member="OWP_Employee Type","OWP_FTE","OWP_Applicable Union Code", "OWP_Merit Month";Column=3|4|7|5;**. In this case, the column dimension is "Account" and import columns are 3,4,5,7. Target values for Account are listed after the member.

To specify a range of columns, use a , (Comma). For example 3,8 is column 3 through 8. Use | (pipe) to specify distinct columns 3|6|8 is columns 3,6,8.

Concat

The Concat expression type enables you to combine text from different columns to create a new column with a column name separated by a delimiter. For example, use the expression type: Concat parameters ("1-5", "*") to combine text from columns 1-5 with a new column name separated by an asterisk (*) delimiter.

The parameters for the Concat expression type are: ("", "") parameters.

The Concat expression type is available only for Quick Mode - File-based source expressions.

Mapping Members

Translate source member values to valid members in each target dimension.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Mapping a member allows you to translate source values to valid members in each target dimension. If you have defined target expressions to derive the target members for a given dimension, then you don't have to define member mapping. If you use conditional target expressions, then you can define member mapping for the remaining source values not covered by the conditions. If your data transformations can be achieved using target expression, then its highly recommended to use it instead of mapping members. Transformation using target expression perform significantly better than member mapping for large data sets.

Mappings can be shared across multiple integrations using parent location. Mappings are processed for each dimension in default order of Account, Entity, ICP, UD1, UD2, etc. You can override the mapping order by changing the calculation sequence in Application definition.

Adding Member Mappings

When processing the source values for transformations, multiple mappings may apply to a specific source value. The order of precedence is Explicit, Between, In, Multi-Dimensional, and Like.

To add a member mapping:

1. From the **Data Integration** home page, click ••• to the right of the integration, and then select **Map Members**.
2. If you want to share the map from another integration, select the **Location Reference**.

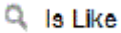
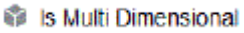
   You are required to provide a Location reference if you define maps for the current integration.
3. From the **Dimension** drop-down, select the dimension to which to edit or add member mappings.
4. Click ➕ (Add icon) to add a new member mapping.

   You can also just edit existing mappings as needed without adding a new mapping by selecting a mapping and clicking the Edit icon .
5. From **Add Member Mapping** page, then **Source**, select the member mapping type from the **Mapping Type** drop down, and then specify the source value.

   Types of member mappings:

Type of Mapping	Description	See Also
== Explicit	Match and replace the source value exactly with the target value. Explicit mappings are one to one mappings, for example, source value "ABC" is replaced with target value "123."	Using Explicit Mappings
<> Between	Replace a continuous range of source values with a single target value. For example, a range from "001" to "010" is replaced as one value. In another example, you may need to map accounts 300000 through 3001999 to Retained Earnings, but accounts 310000 might be Capital contributions or dividends.	Using Between Mappings

Copyright © 2017, 2026, Oracle and/or its affiliates.

Type of Mapping	Description	See Also
	List non-sequential (non-continuous) source values to be mapped to one target value. In this case, multiple values are mapped to one value within one mapping, eliminating the need to create multiple rules (as is required for an Explicit map). For example, you could have source accounts 1503, 1510, and 1515 map to the target account 15000010.	Using In Mappings
	Use special characters to match a string in the source value and map it to a target value. Like mappings use wildcard characters: asterisks (*) and question marks (?). Asterisks are placeholders for any number of characters. For example, 1190* maps the accounts 1190, 1190100, and 1190-200 to the Cash target account. Question marks are placeholders for one character. For example, the source account of 119? maps only to source accounts that contain four characters and that begin with 119.	Using Like Mappings
	Multi-dimension mapping enables you to assign a target value for a specific combination of source column values. This functionality provides you with the ability to load data into dimensions unavailable in the target application. For example, the mapping for the Account dimension can be based on source values of Entity, Product, and Project. In addition, Lookup dimensions added in the target application registration can be selected. These dimensions contain source dimensions that do not exist in the target application. They provide even more flexibility in creating multi-dimension filters. It is a way to facilitate conditional data loading.	Using Multi-Dimensional Mappings

Copyright © 2017, 2026, Oracle and/or its affiliates.

Type of Mapping	Description	See Also

* Regular Expression
 	
(The Regular Expression mapping type is available for Quick Mode- File Based loads.
 	
For more information, see Using Quick Mode for File-Based Loads.)
	Regular expression that uses a sequence of characters that specifies a match pattern in text.
	
	This mapping type is available for source records only.
	
	While importing map members, enclose the regular expression in double quotes as shown below:
	`"#REGEX <!SCRIPT>$RECORD$[a-z,A-Z]{3},[0-9]{4},[0-9]{4},[0-9]{4},[0-9]{4},[a-z,A-Z,\s]*,[a-z,A-Z,\s,-]*,3121,14020,[0-9]*,[a-zA-Z]*,[A-Z]*,[0-9]*,[A-Z,a-z]*,[0-9]*.[0-9]*<!SCRIPT>"`
	Only Java supported regular expressions can be used.
	
	For example, if you want to apply a regular expression that ignores a row that has five digits in the source record, you can specify **.*\d{5}** and in the target, you specify **IGNORE**.
	
	Other examples:
	
	110[1-9]{3}999 - Starts with 110 any 3 numbers ends with 999
	
	^[A-Z][1-5]{5} - Starts with uppercase alphabet followed by 5 numbers between 1 & 5
	
	^1[1-5]{5}9{3} - Starts with 1 any 5 number between 1 & 5 and ends with 999
	
	A regular expression can be defined for an entire record in addition to an individual field. This enables you to define a regular expression on multiple fields on the whole record string. The expression is specified in the format: `$RECORD$regex`.
	
	For example, to filter a record with values 3121,14020 from the following,
	
	`"USD","0000","0000","0000","0000","Ending Balance","Vision Foods - USA IFRS SecLed","3099","31010","000","Actual","YTD","0000","Total",0.0`
	`"USD","0000","0000","0000","0000","Ending Balance","Vision Foods - USA IFRS SecLed","3111","11010","000","Actual","YTD","0000","Total",-70346.2`
	`"USD","0000","0000","0000","0000","Ending Balance","Vision Foods - USA Ledger","3121","14020","000","Actual","YTD","0000","Total",4.228475022E7`
		See:
		• Lesson: Regular Expressions
		• Regular Expressions

Copyright © 2017, 2026, Oracle and/or its affiliates.

Type of Mapping	Description	See Also
	use the expression `$RECORD$[a-z,A-Z]{3},[0-9]{4},[0-9]{4},[0-9]{4},[0-9]{4},[a-z,A-Z,\s]*,[a-z,A-Z,\s,-]*,3121,14020,[0-9]*,[a-zA-Z]*,[A-Z]*,[0-9]*,[A-Z,a-z]*,[0-9]*.[0-9]*`	

> **Note**
>
> When processing the source values for transformations, multiple mappings may apply to a specific source value. The order of precedence is: Explicit, Between, In, Multi-Dimensional, and Like. Within Between and Like types mappings, can overlap.

6. In **Target**, enter the target value for the dimension member name.

   You can enter an individual member as target or select a value using member selector by clicking by clicking .

7. In **Processing Order**, specify the order of the mapping.

   The processing order determines the level of precedence within a mapping type. Mappings are processed in alphabetical order by the name within a mapping type. Numbers may also be used to help with ordering. For example, if numbering by tens or hundreds, you can insert a new value between existing ones. When mappings are numbered 10, 20, and 30, you can add a mapping that starts with 25 and need not rename other mappings.

8. In **Description**, enter a description of the mapping.

9. Select **Change Sign** to reverse the sign of the target account specified.

   The option is often used with General Ledger source data when the trial balance has negative signs for Revenue and Liability/Equity source accounts. In applications, numbers are often stored as positive values so you can use the change sign option to make the credit balances positive.

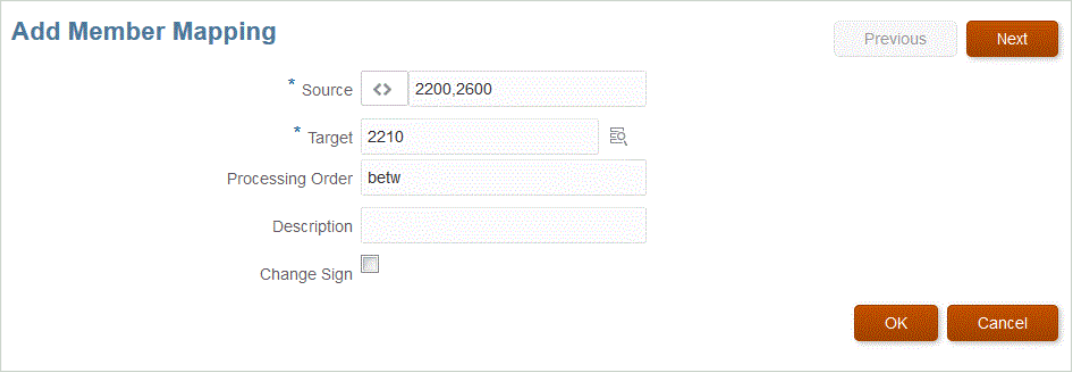
> **Note**
>
> The Change Sign option is not supported for Quick Mode loads.

10. Click **OK**.

    Optionally, you can edit a member mapping by selecting the mapping and clicking .

    You can delete a member mapping by selecting the mapping and clicking .

Copyright © 2017, 2026, Oracle and/or its affiliates.



Using Explicit Mappings

To use an Explicit member mapping:

1. From the **Data Integration** home page, click ••• to the right of the integration, and then select **Map Members**.
2. Click **Edit**.
3. From the **Dimension** drop-down, select the dimension to which to edit or add member mappings.
4. Click + (Add icon) to add a new member mapping.

   You can also just edit existing mappings as needed without adding a new mapping.
5. In **Source**, click == from the **Mapping Type** drop-down and then specify the source value.

> **Note**
>
> When processing the source values for transformations, multiple mappings may apply to a specific source value. The order of precedence is Explicit, Between, In, Multi-Dimensional, and Like.

6. In **Target**, enter the target value for the dimension member name.
7. In **Description**, enter a description of the mapping.
8. Select **Change Sign** to reverse the sign of the target account specified.

   The option is often used with General Ledger source data when the trial balance has negative signs for Revenue and Liability/Equity source accounts. In Oracle Fusion Cloud Enterprise Performance Management applications, positive numbers are often loaded as credits and all negative numbers are loaded as debits. Consequently, you can reverse the sign.

> **Note**
>
> The Change Sign option is not supported for Quick Mode loads.

9. Click **Save**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Using Between Mappings

To use a Between member mapping:

1. From the **Data Integration** home page, click ••• to the right of the integration, and then select **Map Members**.
2. Click **Edit**.
3. From the **Dimension** drop-down, select the dimension to which to edit or add member mappings.
4. Click (Add icon) to add a new member mapping.

   You can also edit existing mappings as needed without adding a new mapping.
5. In **Source**, click from the **Mapping Type** drop-down and then specify the source values.

   Separate source values with a comma, for example, specify: `100,199`. Between mappings do not support special characters, such as an asterisk.

   > **Note**
   >
   > When processing the source values for transformations, multiple mappings may apply to a specific source value. The order of precedence is Explicit, Between, In, Multi-Dimensional, and Like.
6. In **Target**, enter the target value for the dimension member name.
7. In **Processing Order**, specify the order of the mapping.

   The processing order determines the level of precedence within a mapping type. Mappings are processed in alphabetical order by the name within a mapping type. Numbers may also be used to help with ordering. For example, if you use Number for processing, note that the processing order is an alphanumeric sort order. If have 10, 20, 30, 100 as the order, the processing order will be 10, 100, 20, 30. When using numbers for the processing order, use the same number of digits for all maps.
8. In **Description**, enter a description of the mapping.
9. Select **Change Sign** to reverse the sign of the target account specified.

   The Change Sign option is often used with General Ledger source data when the trial balance has negative signs for Revenue and Liability/Equity source accounts. In EPM applications, positive numbers are often loaded as credits and all negative numbers are loaded as debits. Consequently, you can reverse the sign.

   > **Note**
   >
   > The Change Sign option is not supported for Quick Mode loads.
10. Click **Save**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Using In Mappings

To use an In member mapping:

1. From the **Data Integration** home page, click ••• to the right of the integration, and then select **Map Members**.
2. Click **Edit**.
3. From the **Dimension** drop-down, select the dimension to which to edit or add member mappings.
4. Click + (Add icon) to add a new member mapping.

   You can also just edit existing mappings as needed without adding a new mapping.
5. In **Source**, click IN from the **Mapping Type** drop-down and then specify the source values.

   Separate source values with a comma, for example, specify: `100,199`. Between mappings do not support special characters, such as an asterisk.

   > **Note**
   >
   > When processing the source values for transformations, multiple mappings may apply to a specific source value. The order of precedence is Explicit, Between, In, Multi-Dimensional, and Like.
6. In **Target**, enter the target value for the dimension member name.
7. In **Processing Order**, specify the order of the mapping.

   The processing order determines the level of precedence within a mapping type. Mappings are processed in alphabetical order by the name within a mapping type. Numbers may also be used to help with ordering. For example, if you use Number for processing, note that the processing order is an alphanumeric sort order. If have 10, 20, 30, 100 as the order, the processing order will be 10, 100, 20, 30. When using numbers for the processing order, use the same number of digits for all maps.
8. In **Description**, enter a description of the mapping.
9. Select **Change Sign** to reverse the sign of the target account specified.

   The Change Sign option is often used with General Ledger source data when the trial balance has negative signs for Revenue and Liability/Equity source accounts. In Oracle Fusion Cloud Enterprise Performance Management applications, positive numbers are often loaded as credits and all negative numbers are loaded as debits. Consequently, you can reverse the sign.

   > **Note**
   >
   > The Change Sign option is not supported for Quick Mode loads.
10. Click **Save**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Using Like Mappings

To use a Like member mapping:

1. From the **Data Integration** home page, click ••• to the right of the integration, and then select **Map Members**.
2. Click **Edit**.
3. Click + (Add icon) to add a new member mapping.

   You can also just edit existing mappings as needed without adding a new mapping.
4. In **Source**, click from the **Mapping Type** drop-down and then specify the source value.

   When specifying the source value for a Like mappings, you can use special characters as parameters. See Using Special Characters in the Source Value Expression for Like Mappings.

   > **Note**
   >
   > When processing the source values for transformations, multiple mappings may apply to a specific source value. The order of precedence is Explicit, Between, In, Multi-Dimensional, and Like.
5. In **Target**, enter the target value for the dimension member name.
6. In **Processing Order**, specify the order of the mapping.

   The processing order determines the level of precedence within a mapping type. Mappings are processed in alphabetical order by the name within a mapping type. Numbers may also be used to help with ordering. For example, if you use Number for processing, note that the processing order is an alphanumeric sort order. If have 10, 20, 30, 100 as the order, the processing order will be 10, 100, 20, 30. When using numbers for the processing order, use the same number of digits for all maps.
7. In **Description**, enter a description of the mapping.
8. Select **Change Sign** to reverse the sign of the target account specified.

   The Change Sign option is often used with General Ledger source data when the trial balance has negative signs for Revenue and Liability/Equity source accounts. In EPM applications, positive numbers are often loaded as credits and all negative numbers are loaded as debits. Consequently, you can reverse the sign.

   > **Note**
   >
   > The Change Sign option is not supported for Quick Mode loads.
9. Click **Save**.

Using Special Characters in the Source Value Expression for Like Mappings

The Source and Target Value expressions can have one or more special characters. Special characters are supported for Like mappings only.

Copyright © 2017, 2026, Oracle and/or its affiliates.

- **Asterisk (*)**

  An asterisk (*) represents the source value. The asterisk (*) can be prefixed or suffixed by one or more characters, which filters the source value by that prefix or suffix. The wild card takes whatever is present in the source and puts it in the target column, usually adding a prefix.

- **Question Mark (?)**

  The question mark (?) strips a single character from the source value. You can use one or more question marks (?) in the expression. You can also use question marks in combination with other expressions. For example, A?? finds members that start with A and have any two characters following and selects the members or strips off the two characters.

- **<1>, <2>, <3>, <4>, <5>**

  Processes rows that have concatenated values and extracts the corresponding value. The source member must use the "_" character as the separator.

  > **Note**
  >
  > <1>, <2>, <3>, <4>, <5> can be used with a question mark (?) but cannot be used with an asterisk (*).

- **<BLANK>**

  Processes only rows that contain the blank character (space).

  The system only reads the expression when the source member is ‘ ‘ as <BLANK>. In this case, single quotes surround a single space character. If the source has NULL, which is shown like , , or as a space surrounded by , then the NULL as a <BLANK> is not interpreted. Only the ‘<space char> expression is interpreted.

  > **Note**
  >
  > The <BLANK> notation may be used in both source and target expressions. If used in a target expression, it writes a blank space to the target.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 9-2 Examples of Expressions Using Special Characters

Special Character(s) Used	Mapping Type	Source Value	Target Value	Result	Notes
*	Data Load	*	1000	1000 returns 1000 WXYZ returns 1000	In this example, processes all rows and overrides the source value with a default value of 1000. In the expression, WXYZ also returns 1000. Because you entered an asterisk for the source value, any source value is replaced with the target value of 1000.
*	Data Load	*	*	1000 returns 1000 WXYZ returns WXYZ	In this example, processes all rows and replaces the source value as is.
*	Stripping	*	A*	101 returns A101	Processes all source members and adds an "A" as a prefix.
*	Stripping	*_DUP	*	1000_DUP returns 1000	Processes and strips off only source values ending with "_DUP."
?	Stripping	?*	*	A1000 returns 1000 B2000 returns 2000	This result processes only source values of one or more characters in length. Strips off the first character
?	Stripping	*????	*	1000_DUP returns 1000 A1000 returns A	This result processes only source values of four or more characters in length. Strips off the last 4 characters
<1>, <2>, <3>, <4>, <5>	Data Load	<1>	*	01_420 returns 01	
<1>, <2>, <3>, <4>, <5>	Data Load	<2>	*	01_420 returns 420	
<1>, <2>, <3>, <4>, <5>	Data Load	<3>	*	01_420_AB_CC1_001 returns AB	

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 9-2 (Cont.) Examples of Expressions Using Special Characters

Special Character(s) Used	Mapping Type	Source Value	Target Value	Result	Notes
<1>, <2>, <3>, <4>, <5>	Stripping	?<1>	*	`A01_420` returns `01`	
<BLANK>	Data Load	<BLANK>	[None]	`' '` returns `[None]` `'01_ '` returns `[None]`	Single quotation marks are shown for illustration only.

Using Special Characters in the Target Value Expression

You can use only an asterisk (*) in the target expression, and you can prefix or suffix any number of characters to the asterisk (*) character. When you run the integration, the asterisk (*) is replaced by the resulting source value (which may or may not have its own source expression). It is concatenated to any prefix or suffix that you have specified in the target expression.

For example:

Target Value:

```
A*
```

Result:

```
1000 = A1000
```

Target Value:

```
*_DUP
```

Result:

```
1000 = 1000_DUP
```

Using Multi-Dimensional Mappings

When using multi-dimensional mapping, the source needs to be less than or equal to 75 characters.

To add a multi-dimensional mapping:

1. From the **Data Integration** home page, click ⋯ to the right of the integration, and then select **Map Members**.
2. Click ➕ .
3. From the **Add Member Mapping** page, in **Source**, select **Is Multi Dimensional**.
4. Click **Dimension** and select the dimension of the source column value.
5. Click the Member Mapping Type icon to select the method to use for mapping source column values.

   Available types include:

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Explicit
- Between
- Like
- In
- Multi Dimensional

For more information on these types, see Adding Member Mappings.

6. In the entry box, specify the source value string.
7. Click (Add icon) to add a new member mapping type to the multi-dimensional condition.
8. In **Target**, enter the target dimension member for which you are defining a map.

   You can also click on the Select Members page.

   The target values for multi-dimensional mapping must be an explicit member name. Wildcard or special characters are not supported.
9. In **Processing Order**, specify the order in which the processing of maps occurs.

   The order can be alpha or numeric. If you specify an alphabetical order, mappings are processed in alphabetical order by the name within a mapping type.

   You might use an alphabetical order to process maps using the integration name so that maps are processed in alphabetical order of the name within a mapping type. Numbers may also be used to help with ordering. For example, if numbering by tens or one hundred, insert new integrations between existing ones. For example, if integrations are numbered 10, 20, and 30, add an integration that starts with 25 so that you do not need to rename other integrations.

   > **Note**
   >
   > When processing the source values for transformations, multiple mappings may apply to a specific source value. The order of precedence is Explicit, Between, In, Multi-Dimensional, and Like. Within Between and Like types, mappings can overlap.

10. In **Description**, enter a description of the mapping.
11. Select **Change Sign** to reverse the sign of the selected target account.

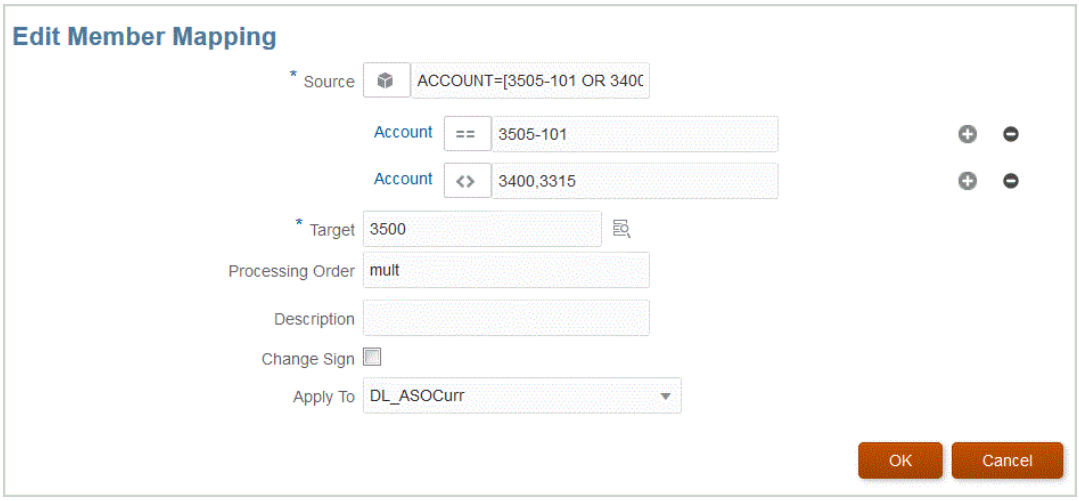
    The Change Sign option is often used with General Ledger source data when the trial balance has negative signs for Revenue and Liability/Equity source accounts. In EPM applications, positive numbers are often loaded as credits and all negative numbers are loaded as debits. Consequently, you can reverse the sign.

    > **Note**
    >
    > The Change Sign option is not supported for Quick Mode loads.

12. In **Apply To**, select the integration to which to apply the selected mapping.

    By default, mappings specified at a location are applicable to *all* integrations associated with the location.

Copyright © 2017, 2026, Oracle and/or its affiliates.



13. Click **Save**.

Using Special Characters in Multi-Dimensional Mapping

The Source and Target Value expressions can use special characters. These characters (typically ? and *) can be prefixed or suffixed by one or more characters, which filters the source value by that prefix or suffix.

Special characters include:

- Asterisk (*)—An asterisk (*) represents the source value. The asterisk (*) can be prefixed or suffixed by one or more characters, which filters the source value by that prefix or suffix. The wild card or strips (data load to write back) takes whatever is present in the source and puts it in the target column, usually adding a prefix. An asterisk (*) represents the source value. The asterisk (*) can be prefixed or suffixed by one or more characters, which filters the source value by that prefix or suffix. The wild card takes whatever is present in the source and puts it in the target column, usually adding a prefix.
- Question Mark (?)—The question mark (?) strips a single character from the source value. You can use one or more question marks (?) in the expression. You can also use question marks in combination with other expressions. For example: A?? finds members that start with A and have any two characters following and selects the members or strips off the two characters.

For multiple dimensions, the source dimension is the value brought over and the wild card applies to it alone. The dimensions can be present in a multiple dimensional rule and use wild cards. The prefix/suffix applies only to the source, which equals the target dimension (the dimension on which the rule resides).

Format Mask Mapping for Target Values

Use a format mask to define a target member based on a combination of source member details, and optional user-defined text. For example, use a format mask to designate a target member based on some part of the source member, an additional prefix, suffix, or replacement text for the target.

The format mask is available for the target member specification for all mapping types except explicit. Common usage of this mapping type falls into three categories: replacing segments from the source, replacing segments with string operations, and replacing segments with string operations using a prefix or a suffix.

Copyright © 2017, 2026, Oracle and/or its affiliates.

#FORMAT Mapping Type Components

The #FORMAT mapping type consists of the following components:

Table 9-3 #Format Mapping Type Components

Component	Description
#FORMAT	Indicates that a mapping type of FORMAT is specified in the target member.
<format mask>	User-defined format mask with the following characters used to define the format: • "?"—Include a character from a specific position in the source member or segment within a member. • "#"—Skip or drop a character from the source when creating the target member. • "character"—Include the user defined character on the target "as- is". Used for prefixing, suffixing or any fixed string or required character. This can be used in conjunction with the special format mask characters. • "*"—Include all characters from the source segment or source. When "*" is used as the only format mask character in a segment, then the entire segment value is copied from the source. When "*" is used in conjunction with "#" or the "?" character, then all remaining and unused characters are brought over. "*" is a wildcard character that takes the remaining characters not specified by "?" or "#". For example, when the source is "abcd" and "*" is used, then the target is "abcd." When the target is "?#*," then the result is "acd." If the system encounters a "*" within a segment, then anything specified after the "*" is ignored other than the "character" specified on the format.
<segment delimiter>	The optional segment delimiter defines the character that is used to delimit the segments in the source and target member. For this integration type, the source and target delimiter must be the same. When the segment delimiter is not specified, then the format mask is applied to the entire member independent of any segment specification or delimiter.

#FORMAT Mapping Example

The following is an example that uses all options provided by #FORMAT:

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 9-4 #Format Mapping Type Example

Source	Target	Result
12345-6789-012-3456ABC-001	#FORMAT("???-*-GROUP-AA##?#*X-GROUP","-") Explanation: Take the first three characters of the first segment, take the entire second segment, replace the third segment with the text "GROUP," prefix the fourth segment with AA, drop the third and fourth characters, keep the fifth character, drop the sixth character, keep ABC and add suffix "X", replace the fifth segment with the text "GROUP."	123-6789-GROUP-5ABCX-GROUP

Replacing Segments

You can use the format of the source member as the definition of the target member but replace some of the source segments rather than reuse the values from the source. For example, you may have a requirement to filter the source by the value of the 4th segment, replace the 7th segment with an explicit value, and then retain the values of the other segments as in the following:

Source:

```
??????-??????-?-012000000-??????-???-???????-??????-??????-??????-???
```

Target:

```
??????-??????-?-012000000-??????-???-GROUP-??????-??????-??????-???
```

Replacing Segments with String Operations

You can apply a string operation to a segment that is being replaced. For example, you may have a value of 11002293 but when the segments are written, you want to only take the last four digits, or the first six digits. Examples using the member 11002293:

- Ignore the first two characters and provide the result: 002293. Use #FORMAT("##*").
- Truncate the last three characters provide the result: 11002. Use #FORMAT("?????").
- Ignore the first two and truncate the last three with the result: 002. Use #FORMAT("##???").

Replace Segments with String Operations and Using a Prefix or Suffix

You can use the segment value from the source as-is in the corresponding segment in the target. For example, if the source is A100, you can map this value as the value in the target, and then you can map this value as the value in the target. In this case, use a wildcard on the source, and then specify the explicit value for that segment in the target based on the source.

Selecting Members

Use the Select Members page to view and select members with a dimension.

Copyright © 2017, 2026, Oracle and/or its affiliates.

To select a member:

1. From the **Data Integration** home page, click ••• to the right of the integration, and then select **Map Members**.
2. Click **Edit**.
3. In **Target**, click .
4. On the **Select Members** page, perform these tasks:
   a. From **Cube**, select the plan type of the target system.

      The Cube determines the structure of available dimensions.

   b. **Optional**: In **Search Member**, enter search criteria (member name or alias only) and click **Enter**.

      The search isn't case-sensitive. You can search for a word, multiple words, or wildcard characters.

   c. The second pane from the left shows the first level of parent dimensions available. Make selections by clicking the check box(es) next to each member in the pane to map.

      To drill down to the members in a selected dimension and display the results in the third pane, click .

   d. The third pane from the left are the members or siblings for the selected dimensions selected in step b. Make selections by clicking the check box(es) next to each member to map.

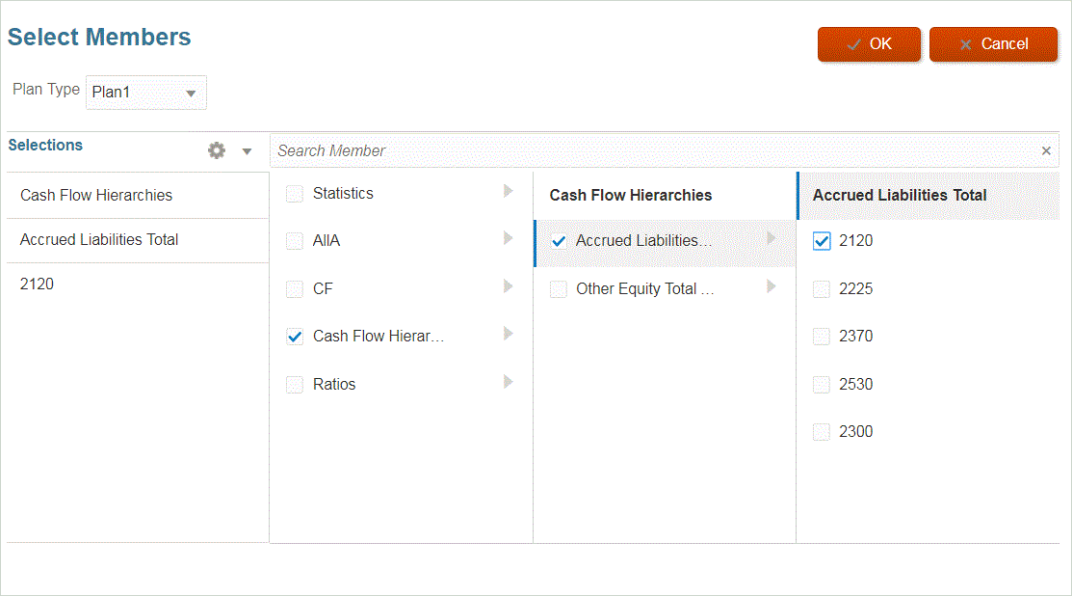
      To drill down to the members of a selected dimension and display the results in the fourth pane, click .

   e. The fourth pane from the left are the members (leaf level) results from the selected member/sibling in step c. Make selections by clicking the check box(es) next to each member to map.

   Selected dimensions/members that display a check mark are moved to the **Selections** pane.

   To clear selections you have made, select the dimension/member from the **Selections** pane, and from the ⚙ drop-down, select either: **Remove** or **Remove All**.
5. Click **OK**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



Filtering Members

You can apply filters to view specific member. You create filter criteria based on one or dimensions using a "Contains" or an "Equal" operand and a specific value.

To apply a member filter:

1. From the **Data Integration** home page, click ••• to the right of the integration, and then select **Map Members**.
2. Click **Add Filter**.
3. Select to apply the filter either to a **Source**, **Target**, or **Processing Order** value.
4. From the **Operand** drop down, select the operand and value for the filter.

   Available operands are:

   - Contains—Return values that contain the string characters in the Value field.
   - Equals—Return values that are equal to the string characters in the Value field.
5. In **Value**, specify the specific value to use for the filter.

Import Member Mappings

You can import either a current dimension or all member mappings from a selected CSV and TXT file format, and then you can create mappings. Import member mappings support merge or replace modes, along with validate or no validate options for target members.

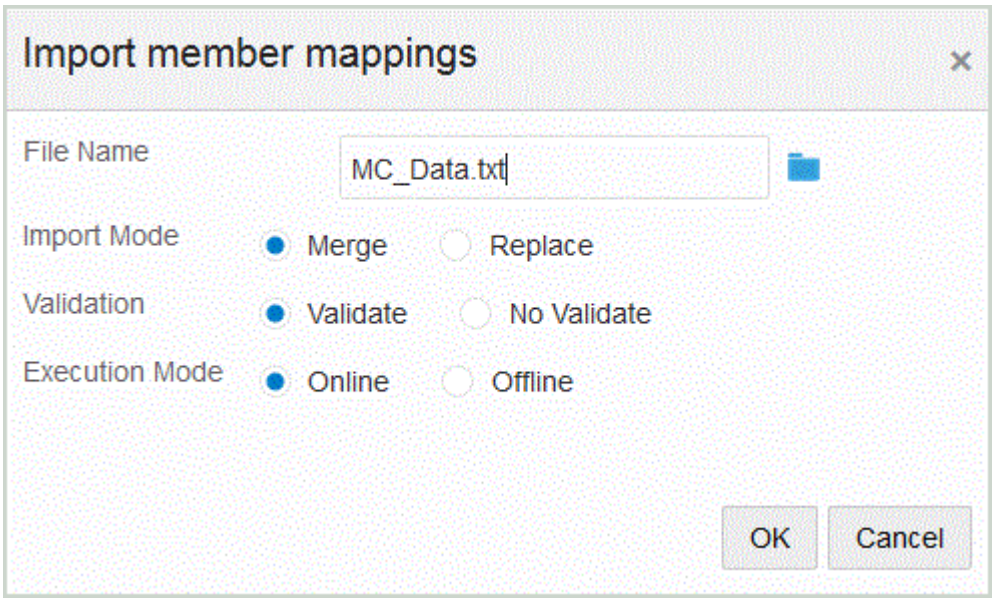
To import member mappings:

1. From the **Data Integration** home page, click ••• to the right of the file-based integration, and then select **Map Members**.
2. Click **Edit**.
3. From the **Dimension** drop-down, select the dimension to which to edit or add member mappings.
4. From the **Actions** drop-down, and then **Import**, select **Current Dimension**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

To import all dimensions, select **All Dimensions**.

5. From **Import member mappings**, and then **File Name**, specify the name of the file from which to import the member mapping.



6. From **Import Mode**, select the import mode:
   - Merge—Update existing maps. Existing maps are identified based on Source value, Processing Order and Integration Name if specified. If the map does not exist, then a new map is created.
   - Replace—Delete the existing maps and create new maps from the file.
7. From **Validate**, select to validate the member mappings.

   Validate ensures that all data in the imported General Ledger has a corresponding mapping.
8. In **Execution Mode**, select the mode for executing the import:
   - Online—Processes the import immediately.
   - Offline—Runs the import in the background.
9. Click **OK**.

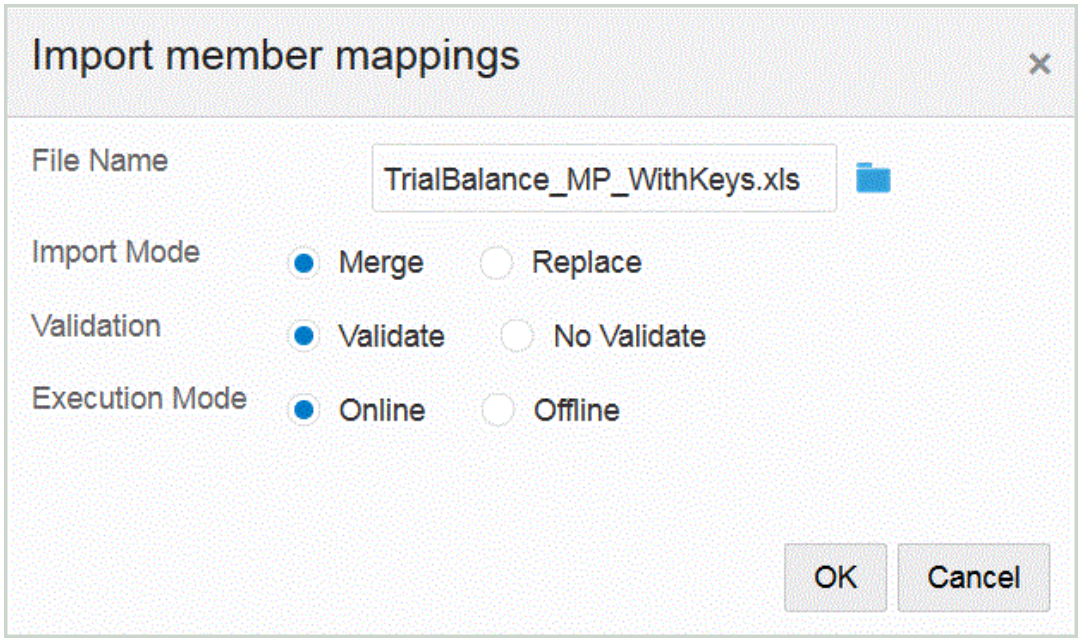
Importing Excel Mappings

You can import Excel mappings by selecting the Import option and selecting an Excel mapping.

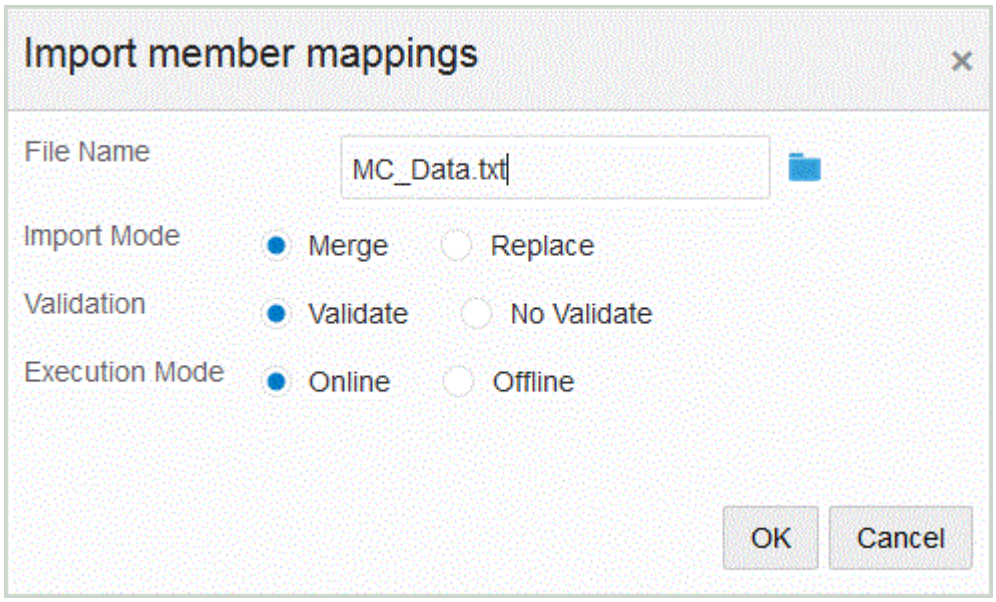
To import Excel mappings:

1. From the **Data Integration** home page, click ⋯ to the right of the integration, and then select **Map Members**.
2. From the **Actions** drop-down, select **Import**, and then select **Import from Excel**.
3. From **Import member mappings**, and then **File Name**, specify the name of the Excel.XLS file from which to import the member mapping.

Copyright © 2017, 2026, Oracle and/or its affiliates.



4. From **Import member mappings**, and then **File Name**, specify the name of the file from which to import the member mapping.



5. Click **OK**.
6. From **Import Mode**, select the import mode:
   - Merge—Overwrites the data in the application with the data in the data load file.
   - Replace—Clears values from dimensions in the data load file and replaces it with values in the existing file.
7. From **Validate**, select to validate the member mappings.

   Validate ensures that all data in the imported General Ledger has a corresponding mapping.
8. In **Execution Mode**, select the mode for executing the import:
   - Online—Processes the import immediately.

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Offline—Runs the import in the background.

9. Click **OK**.

Downloading an Excel Template (Mapping Template)

Download and use the Maploader template to upload your mapping tables from Excel instead of entering them by way of Data Integration. This feature enables you to map members by selecting and importing them form your Excel mapping. You can merge or replace mappings. it does not update map. It simply adds new entries into the database. Excel mapping templates with correct formatting are included in the `EPM_ORACLE_HOME/products/FinancialDataQuality/templates` directory.

The mapping template also includes a macro script that pulls Oracle Hyperion Financial Management dimensions directly from the target application to which you are connecting.

You must upload the Excel template to the Data Integration server, and then pick the Excel file as the file to load when you run the integration, or when prompted by the system if the file name is left blank. The system determines if the file being processed is an Excel file, and then reads the required formatting to load the file.

When working with a mapping template in Excel:

- Do not have any blank lines in the map template.
- You can insert lines in the template, but you must keep new lines in the designated area.
- Each template supports a single dimension.

To download an Excel template:

1. From the **Data Integration** home page, click ⋯ to the right of the integration, and then select **Map Members**.
2. On the **Map Members** tab, select **Actions**, then **Import**, and then **Download Excel Template**.

   The **MapLoaderTemplate.xls** template file is downloaded to you system.
3. Open **MapLoaderTemplate.xls**.
4. Click the **Map** sheet.
5. Enter the **Location** in **cell B1**, **Location ID** in cell **B2**, and select the dimension from the **Dimension** drop-down in cell **B3**.
6. Complete the following column fields:
   a. In **Source**, enter the source dimension value.

      You can specify wildcards and ranges when entering the source dimension.

      - Wildcards for unlimited characters—Use asterisks (*) to denote unlimited characters. For example, enter 548* or *87.8.
      - Wildcards for single character place holders—Use questions marks (?) to denote single character place holders. For example,
        - 548??98
        - ??82???
        - ??81*
      - **Range**—Use commas (,) to denote ranges (no wildcard characters are allowed). For example, specify a range as 10000,19999.

Copyright © 2017, 2026, Oracle and/or its affiliates.

(This range evaluates all values from 10000 to 19999 inclusive of both start and end values.)

In this case, Data Integration considers all values from 10000 to 19999 to include for both start and end values.

- **In map**—Use commas (,) to separate entries (no wildcard are characters allowed). You must have at least three entries or the map shows as a between map. For example, specify an In map as 10,20,30.
- **Multi-Dimension map**—Use `#MULTIDIM` to indicate its multidimensional mapping. Enter the `DIMENSION NAME=[VALUE]` and the value. The value follows the logic as wildcard, range, and In map. In the following example the search criteria are all accounts starting with 77 and UD1 = 240. For example, `#MULTIDIM ACCOUNT=[77*] AND UD1=[240]`.

b. In **Source Description**, enter a description of the source value.

c. In **Target**, enter the target dimension value.

d. In **Change Sign**, enter **True** to change the sign of the Account dimension. Enter **False** to keep the sign of the Account dimension. This setting is only used when mapping the Account dimension.

e. In **Rule Name**, enter the integration name when the mapping applies to a specific data rule name.

> **Note**
>
> If you are adding an Explicit mapping, the rule name must equal the source value.

	A	B	C	D	E
1	FDM Location:	TEXAS			
2	FDM Location ID:	751			
3	Dimension:	Account			
4					
5					
6	Source	Source Description	Target	Rule Name	Convert (True/False)
7		tDataMap			
8		SrcDesc	TargKey		ChangeSign
9	1*	Acct Like 1	*	w1x9	FALSE
10	4110,4120,4140	Acct in 4110,4120,4140	4110	w411010	FALSE
11	6*	Acct in 6 range	6110	w6x11	FALSE
12	7000,7999	Acct in 7000 range	#SCRIPT	w700012	FALSE

7. Click **Save**.

Restoring Member Mappings

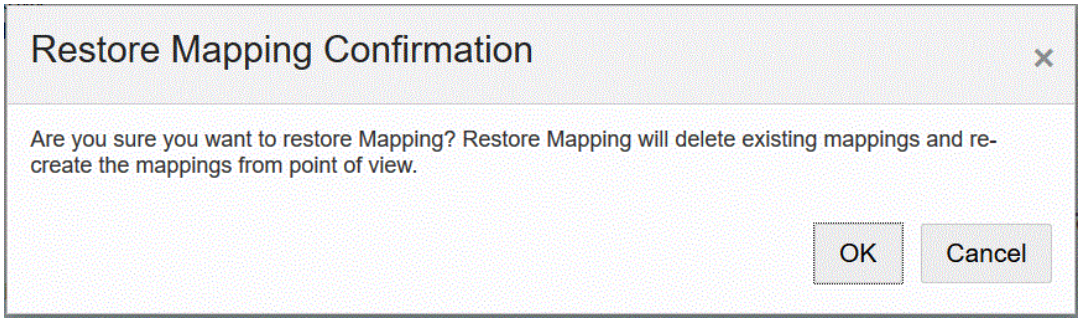
Restoring member mappings deletes mappings made in the current session and restores mappings based on the point of view.

You can restore member mappings only from the last data load for the POV.

To restore member mappings:

1. From the **Data Integration** home page, click ⋯ to the right of the integration, and then select **Map Members**.
2. From the Map Members page drop-down, select **Actions**, then **Restore Mappings**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



3. On the **Restore Mapping Confirmation** page, click **OK**.

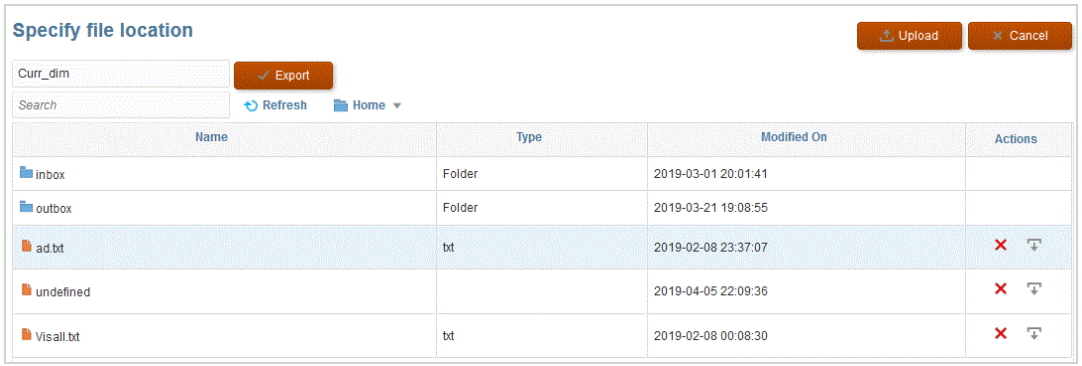
Exporting Member Mappings

You can export member mappings to a selected file, or an Excel file.

Exporting the Current Dimension or All Dimensions Map

To export member mappings for the current dimension or all dimensions:

1. From the **Data Integration** home page, click ••• to the right of the integration, and then select **Map Members**.
2. From the **Dimension** drop-down, select the dimension to which to edit or add member mappings.
3. From the **Actions** drop-down list, click **Export** and select **Current Dimension** or **All Dimensions**.
4. In **Specify file location**, specify the file name to which to export the mapping.



5. Click **Export**.
6. **Optional**: Click **Upload** and navigate to the file to export, and then click **OK**.

Exporting the Map to Excel

When you export to Excel, you cannot re-import in that format.

To export member mappings to Excel:

1. From the **Data Integration** home page, click ••• to the right of the integration, and then select **Map Members**.
2. From the **Dimension** drop-down, select the dimension to which to edit or add member mappings.

Copyright © 2017, 2026, Oracle and/or its affiliates.

3. From the **Actions** drop down, then **Export** drop-down, select **Export to Excel**.

   For the Export to Excel method, mappings are exported to a Microsoft Excel spreadsheet. The spreadsheet uses the location name for the name of the spreadsheet.

   The XLS file is downloaded to your hard drive.

4. Open the spreadsheet.
5. Select the **Map** tab.
6. Enter the **Location** in **cell B1**, **Location ID** in cell **B2**, and select the dimension from the **Dimension** drop-down in cell **B3**.
7. Complete the following column fields:
   a. In **Source**, enter the source dimension value.

      You can specify wildcards and ranges when entering the source dimension.

      - Wildcards for unlimited characters—Use asterisks (*) to denote unlimited characters. For example, enter 548* or *87.8.
      - Wildcards for single character place holders—Use questions marks (?) to denote single character place holders. For example,
        – 548??98
        – ??82???
        – ??81*
      - **Range**—Use commas (,) to denote ranges (no wildcard characters are allowed). For example, specify a range as 10000,19999.

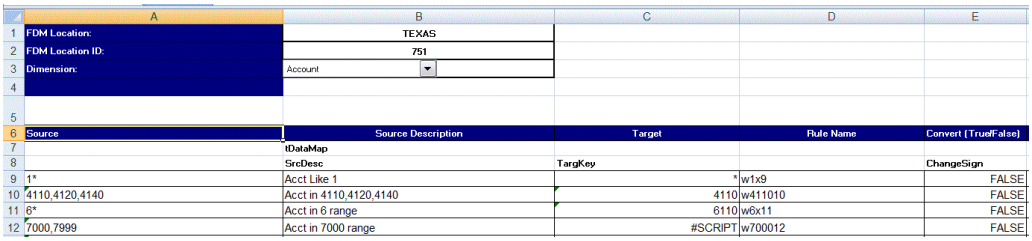
        (This range evaluates all values from 10000 to 19999 inclusive of both start and end values.)

        In this case, Data Management considers all values from 10000 to 19999 to include for both start and end values.
      - **In map**—Use commas (,) to separate entries (no wildcard are characters allowed). You must have at least three entries or the map shows as a between map. For example, specify an In map as 10,20,30.
      - **Multi-Dimension map**—Use `#MULTIDIM` to indicate its multidimensional mapping. Enter the `DIMENSION NAME=[VALUE]` and the value. The value follows the logic as wildcard, range, and In map. In the following example the search criteria are all accounts starting with 77 and UD1 = 240. For example, `#MULTIDIM ACCOUNT=[77*] AND UD1=[240]`.

   b. In **Source Description**, enter a description of the source value.
   c. In **Target**, enter the target dimension value.
   d. In **Change Sign**, enter **True** to change the sign of the Account dimension. Enter **False** to keep the sign of the Account dimension. This setting is only used when mapping the Account dimension.
   e. In **Rule Name**, enter the integration name when the mapping applies to a specific data rule name.

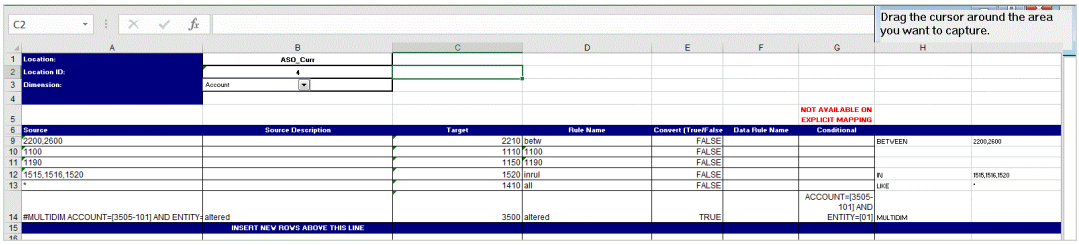
> **Note**
>
> If you are adding an Explicit mapping, the rule name must equal the source value.

Copyright © 2017, 2026, Oracle and/or its affiliates.



	A	B	C	D	E
1	FDM Location:	TEXAS			
2	FDM Location ID:	751			
3	Dimension:	Account			
4					
5					
6	Source	Source Description	Target	Rule Name	Convert (True/False)
7		tDataMap			
8		SrcDesc	TargKey		ChangeSign
9	1*	Acct Like 1	*	w1x9	FALSE
10	4110,4120,4140	Acct in 4110,4120,4140	4110	w411010	FALSE
11	6*	Acct in 6 range	6110	w6x11	FALSE
12	7000,7999	Acct in 7000 range	#SCRIPT	w700012	FALSE

8. Click **Export**.



Setting Data Integration Options

Specify general options for period processing, filters for importing data, and data load options.

Use Options to define general options for period processing, filters for importing data, and options on how to process the data when loading to target application.

In the general option section, specify the Category to load the data, and Period mapping type used to determine source periods. For file-based integrations, options also enable you to specify the default folder and file name.

In the target option section, specific methods can be selected to define how load data to the target, batch size for processing, enable drill region creation and other processing options depending on the application type.

For direct integrations, you can specify the filters to extract the data from the source. The Oracle Fusion Cloud Enterprise Performance Management applications and general Ledger applications filters are specified in terms of data slices using dimension and member filters. For other data sources, you can specify the source filter for each filter parameter.

Defining File-Based Options

For file-based integrations, you can specify the file name and inbox folder used to store the file. Using a file, you can load data to a single period or multiple periods. There are several methods to load data to multiple periods. One method is to use a single file with multiple columns and each column can be mapped to period either using header record or period mappings. The option is to use a single file for each period. To use one file per period, then you can use a specific file name format to identity the period.

Category and periods settings associated with the file may use predefined values. These settings may restrict the options available when editing selections.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> If the Period and Year are mapped in the import format (on the Map Dimensions page), the Period Mapping Type and Calendar show on the Options page. If the Period and Year are not mapped in the import format, the Period Mapping Type and Calendar fields do not show on the Options page.
> To use an explicit period mapping in a file-based integration, you must include the period dimension on the Map Dimensions tab in the integration definition (the Map Dimensions tab is also referred to as the import format). The explicit period mapping is defined on the Source Mapping tab in Period Mappings.

Watch this tutorial for information about setting file-based data integrations options: Setting File-Based Data Integration Options and Running Integrations in Enterprise Planning Cloud.

To select file-based options:

1. From the **Data Integration** home page, click ••• to the right of the file-based integration, and then select **Options**.
2. Select the **Options** tab.
3. From **File Name**, select the name of the source system file.

   If you do not specify a directory name, the system imports data files from the Data Integration home directory. In this case, reference the file in the Applications-Inbox/Outbox Explorer using the format: `epminbox/<filename>` in the file name field. `epminbox/<filename>` should not be specified in the Directory field.

   When only the file name is provided, then data must be entered for a single period on the Run Integration page.

   To load multiple periods, create a file for each period and append a period name or period key to the file name. When the integration is executed for a range of periods, the system constructs the file name for each period and uploads it to the appropriate POV.
4. In **Directory**, specify the folder where the file is located.

   > **Note**
   >
   > If you do not specify a directory name for a file, the system imports data files from the Data Integration home directory. In this case, reference the file in the Applications-Inbox/Outbox Explorer using the format: `epminbox/<filename>` in the **File Name** field, but do not specify `epminbox/<filename>` in this field.
5. From the **File Name Suffix** to load data into multiple periods, select either: **Period Name** or **Period Key**.

   A suffix is appended to the file name, and the system adds the file extension after adding the suffix. If you leave the file name blank, then the system looks for a file with suffix. When the file name suffix type is provided, then the file name is optional in this case, and it is not required on the Run Integration page.

   If the file name suffix type is a period key, the suffix indicator and period date format are required (as the suffix set) in the file name and must be validated as a valid date format.

   For example, specify:

   a. 1_Jan-2024.txt

Copyright © 2017, 2026, Oracle and/or its affiliates.

b. 1_Feb-2024.txt

c. 1_Mar-2024.txt

In this case, when you run the integration, enter **1_.txt** in the file name field and select **Period Name** for the suffix indicator. Then run the integration for the January to March periods.

6. In **Period Key Date Format**, specify the data format of the period key that is appended to the file name in JAVA date format (SimpleDateFormat).
7. From **Category**, select the category.

   The categories listed are those that you created in the in the Data Integration setup, such as "Actual." See Managing Category Mappings.
8. From **Cube**, specify the plan type to load the data.
9. Click **Save**.

Defining Direct Integration Options

When working with a direct integration, you can specify parameters specific to the direct integration source and target system. For example, for a Planning source system, you can the select cube and the period mapping type.

To select direct integration options:

1. From the **Data Integration** home page, click ••• to the right of the direct integration, and then select **Options**.
2. From **General Options**, then **Category**, select a category.

   The categories listed are those that you created in the Data Integration setup, such as "Actual." See Managing Category Mappings.
3. From **Cube**, select the plan type of the target system.

   Cubes are assigned in Data Integration. You can register a custom cube as a separate application with no limit to the number of custom cube applications that can be registered.

   > **Note**
   >
   > When running an integration in Replace mode to an ASO cube, if the scenario member is a shared member, then only a Numeric data load is performed. Be sure to specify the member name with fully qualified name including complete hierarchy. The All Data Type load method does not work when the scenario member is a shared member.

4. From **Period Mapping Type**, select the mapping type of the period.

   Available options:

   - Default—The integration uses the Period Key and Prior Period Key defined in Data Integration to determine the source General Ledger periods mapped to each Data Integration period when the integration is run.
   - Explicit—The integration uses the Explicit period mappings defined in Data Integration to determine the source General Ledger periods mapped to each period. Explicit period mappings support additional General Ledger data sources when periods are not defined by start and end dates.
5. From **Calendar**, select the source system calendar.

Copyright © 2017, 2026, Oracle and/or its affiliates.

6. In **Integration Option 1-4**, specify the free form text or value.

   When you create an integration, a business rule can be attached to run at certain stage. If a business rule has run-time prompts, the you can specify a value using the **Integration Option 1-4** fields to populate the run-time prompts.

   When creating business rules, you can only specify runtime prompts defined on the Independent tab and not the embedded tab. For more information, see Using Business Rules.

7. In **Data Extract Option**, select the type of member data to extract.

   Members can be extracted depending on how they have been flagged for calculation. For a member flagged as "stored," calculated data values are stored with the member in the database after calculation. For a member tagged as "dynamic calc," the member's data values are calculated upon retrieval.

   Available options:

   - All Data—Extracts stored values and dynamically calculated values for both the Dense and Sparse dimension.

     The All Data Extract option is not supported for a business process to business process data synchronization, and you cannot setup the integration in the target business process and extract the data from a remote business. For a business process to business process data synchronization, set up the integration with the All Data option in the source environment where the data is extracted, and then push it to a remote target system.

     Financial Consolidation and Close and Tax Reporting customers can extract dynamic calculated values by selecting the **All Data** option. It is a Data Integration prerequisite that the CONTROL TO-DATE VIEW STORAGE setting in Financial Consolidation and Close and Tax Reporting is enabled, or the Financial Consolidation and Close and Tax Reporting application is DSO (Dense Sparse Optimization enabled application) based, to extract dynamic calculated values. For more information, see Using the Control To Date View Option.

     > **Note**
     >
     > When pulling YTD from Financial Consolidation and Close and Tax Reporting, the optimal method is to convert the application to a DSO application, or use a DSO enabled application, and then select "All Data" as the extract data type, and then use "FCCS_YTD" as the filter for the View dimension.

     Applies to ASO and BSO Hybrid storage options.

   - Stored and Dynamic Calculated Data—Extracts stored dynamic calculated values for the Dense dimension only and not Spare dimensions.

     Applies to the BSO storage option.

   - Stored Data Only—Extracts stored data only. Dynamically calculated values are excluded in this type of extract.

     Applies to the BSO storage option.

8. In **Data Precision**, specify the number of decimal places displayed in numbers to be exported.

   Data precision refers to numeric data with the emphasis on precision (accuracy). Depending on the size of a data value and number of decimal positions, some numeric

Copyright © 2017, 2026, Oracle and/or its affiliates.

fields may be written in exponential format; for example, 678123e+008. You might consider using data precision when data ranges from very large to very small values. The output files typically are smaller and data values are more accurate.

The default value for this option is 16.

9. In **Data Number of Decimal**, specify the maximum number of decimal positions to be exported.

   Specify a value between **0** and **16**. If no value is provided, the number of decimal positions of the data to be exported is used, up to 16 positions, or a value determined by the Data Precision option if that value is specified.

   This parameter is used with an emphasis on legibility; output data is in straight text format. Regardless of the number of decimal positions in the data, the specified number is output. Note that it is possible the data can lose accuracy, particularly if the data ranges are from very large values to very small values, above and below the decimal point.

   By default, 16 positions for numeric data are supported, including decimal positions. If both the Data Precision and the Data Number of Decimal options are specified, the Data Precision option is ignored.

   The data precision option is only available for BSO cubes (and not ASO cubes).

10. (For General Ledger only): From **Include Adjustment Period**, select one of the following options for processing adjustment periods:

    - No—Adjustment periods are not processed. The system processes only regular period mappings (as setup for "default" and "explicit" mappings). **No** is the default option for processing adjustments.
    - Yes—If **Yes** is selected, then the regular period and adjustment period are included. If the adjustment period does not exist, then only the regular period is processed.
    - Yes (Adjustment Only)—If **Yes (Adjustment Only)** is selected, the system processes the adjustment period only. However, if the adjustment period does not exist, the system pulls the regular period instead.

11. (Planning and Oracle General Ledger non-administrator only) From **Replace for Non Admin Load Method**, select the Replace method when loading data to a Planning application for non-administrator users.

    Available methods:

    - None—Performs no replace.
    - Numeric Data Only—Performs replace when the load method is Numeric Data only.
    - All Load Methods—Perform replace for all load methods.

12. Click **Save**.

Defining Target Options

When integrating source applications with target applications, you can select options that support how you load date including load methods, date formats, batch, and purge options.

To define target options:

1. From the **Data Integration** home page, click ••• to the right of the integration, and then select **Options**.
2. Select the application options.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Target Option	Description
Load Method	Select the method for loading data to the target application. Available methods: • **Numeric**—Loads numeric data only. Planning data security is not enforced with this method. • **All data types with auto-increment of line item**—This method requires that you define the data and driver dimensions, and unique identifiers for the Planning application. You define data load and driver dimension in the Data Dimension for Auto-Increment Line Item field and Driver Dimension for Auto-Increment Line Item field below. Replace Mode is not supported for the load method "All data types with auto-increment of line item." • **All data types with security**—Loads Numeric, Text, Smartlist, Date data types. If the Planning administrator loads data, Planning data security *is not* enforced. If a Planning non-administrator user loads data, then Planning data security is enforced. A Planning non-administrator user can load only 500,000 cells of data.
Batch Size	Specify the number of rows read at a time from the file to memory. This parameter is mainly used for performance. When data is loaded, this setting determines how many records are stored in the cache. For example, when **1000** is specified; 1,000 records are stored in cache. Similarly, when **5000** is specified, 5,000 records are stored in cache and commit. Determine this setting by Server Memory and adjust as needed.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Target Option	Description
Drill Region	Tap the slider on to enable the drill region. When this option is enabled, a drillable region is created to use the drill through feature. When loading data, the drill region is loaded to Planning data. Drill region by scenarios are created. For any cube (Planning cubes or Planning databases), the name of the drill region is FDMEE_<name of the scenario member>. When creating the drill region, the system checks if a dimension is enabled for the drill. Members of enabled dimensions selected in data loads, are included in the drill region filter. If no dimensions are enabled, the Scenario, Version, Year, and Period dimensions are enabled by default. You can enable additional dimensions, and the subsequent data load considers members of newly enabled dimensions. If you disable any dimensions which were previously included in a drill region used for drill creation, members of such dimensions are not deleted during the subsequent data loads. If needed, you can remove obsolete members manually.
Purge Data File	Tap the slider on to delete the data file from the application `outbox` directory when a file-based data load is successful. Tap the slider on to delete the file or tap the slider off to retain the file.
Date format for date data	Select the format used for loading date data. Use the date format based on the locale settings for your locale. For example, in the United States, enter the date using the format **MM/DD/YY** format.
Data Dimension for Auto-Increment Line Item	Select the data dimension that matches the data dimension you specified in Planning This option is used for loading incremental data using a LINEITEM flag. See Loading Incremental Data using the LINEITEM Flag.
Driver Dimension for Auto-Increment Line Item	Select the driver dimension that matches the driver dimension you specified in Planning. This setting is used for loading incremental data using a LINEITEM flag. See Loading Incremental Data using the LINEITEM Flag.
Member name may contain comma	Select **Yes** to load data when a member name contains a comma.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Target Option	Description
Summary Drill Behavior when more than 1000 descendants	Select how a summary drill functions with more than 1,000 descendants. Available options: • Ignore • Limit If the number of descendants in summary drill is greater than 1,000 and the **Summary Drill Behavior when more than 1000 descendants** option is **Ignore**, then the system ignores the dimension in the filter and returns the drilled data based on filters on the other dimensions. A maximum of three dimensions can be ignored. If the number of descendants in summary drill is greater than 1,000 and the **Summary Drill Behavior when more than 1000 descendants** option is **Limit**, then the system considers only the first 1,000 members for the dimension while returning the data rows.
Default Import Mode	Sets the default import mode when you execute a data load rule in Data Management or run an integration in Data Integration. Available options: • Append • Replace
Default Export Mode	Sets the default export mode when you execute a data load rule in Data Management or run an integration in Data Integration. Available options: • Accumulate (Add Data) • Replace • Merge Data (Store Data) • Subtract

3. From **Load Method**, select the method for loading data to the target application.

   Available methods:

   - **Numeric**—Loads numeric data only. Planning data security is not enforced with this method.
   - **All data types with auto-increment of line item**—This method requires that you define the data and driver dimensions, and unique identifiers for the Planning application. You define data load and driver dimension in the Data Dimension for Auto-Increment Line Item field and Driver Dimension for Auto-Increment Line Item field below.

     > **Note**
     >
     > Replace Mode is not supported for the load method "All data types with auto-increment of line item."

   - **All data types with security**—Loads Numeric, Text, Smartlist, Date data types. If the Planning administrator loads data, Planning data security *is not* enforced. If a Planning

Copyright © 2017, 2026, Oracle and/or its affiliates.

non-administrator user loads data, then Planning data security is enforced. A Planning non-administrator user can load only 500,000 cells of data.

4. In **Batch Size**, specify the number of rows read at a time from the file to memory.

   This parameter is mainly used for performance. When data is loaded, this setting determines how many records are stored in the cache. For example, when **1000** is specified; 1,000 records are stored in cache. Similarly, when **5000** is specified, 5,000 records are stored in cache and commit.

   Determine this setting by Server Memory and adjust as needed.

5. In **Drill Region**, tap the slider on to enable the drill region.

   When this option is enabled, a drillable region is created to use the drill through feature.

   When loading data, the drill region is loaded to Planning data.

   The drill region by scenarios is created. For any cube (Planning cubes or Planning databases), the name of the drill region is FDMEE_<name of the scenario member>. When creating the drill region, the system checks if a dimension is enabled for the drill.

   Members of enabled dimensions selected in data loads, are included in the drill region filter. If no dimensions are enabled, the Scenario, Version, Year, and Period dimensions are enabled by default. You can enable additional dimensions, and the subsequent data load considers members of newly enabled dimensions. If you disable any dimensions which were previously included in a drill region used for drill creation, members of such dimensions are not deleted during the subsequent data loads. If needed, you can remove obsolete members manually.

6. In **Purge Data File**, tap the slider on to delete the data file from the application `outbox` directory when a file-based data load is successful.

   Tap the slider off to retain the file.

7. In **Date format for date data**, select the format used for loading date data.

   Use the date format based on the locale settings for your locale. For example, in the United States, enter the date using the format **MM/DD/YY** format.

8. From **Data Dimension for Auto-Increment Line Item**, select the data dimension that matches the data dimension you specified in Planning.

   This option is used for loading incremental data using a LINEITEM flag. See Loading Incremental Data using the LINEITEM Flag.

9. From **Driver Dimension for Auto-Increment Line Item**, select the driver dimension that matches the driver dimension you specified in Planning.

   This setting is used for loading incremental data using a LINEITEM flag. See Loading Incremental Data using the LINEITEM Flag.

10. From **Member name may contain comma**, select **Yes** to load data when a member name contains a comma.

11. From **Workflow Mode**, select the data workflow mode.

    Available modes:

    - Full—Data is processed in the TDATASEG_T table and then copied to the TDATASEG table.

      All four Workbench processes are supported (Import, Validate, Export, and Check) and data can be viewed in the Workbench.

      Drill-down is supported.

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Full No Archive—Data is processed in the TDATASEG_T table and then copied to TDATASEG table.

  All four Workbench processes are supported (Import, Validate, Export, and Check). Data can be viewed in the Workbench but only after the import step has been completed. Data is deleted from TDATASEG at the end of the workflow process.

  Drill-down is not supported.

- Simple—Data is processed in the TDATASEG_T table and then exported directly from the TDATASEG_T. table.

  All data loads include both the import and export steps.

  Data is not validated and any unmapped data result in load failure.

  Maps are not archived in TDATAMAPSEG.

  Data cannot be viewed in the Workbench.

  Drill-down is not supported.

  The Simple Workflow Mode is the default mode.

12. From **Enable Data Security for Admin Users**, select to enable data security for administrative users.

    Enables data validation when an administrative user loads data. In this case, all data validations in the data entry form are enforced while loading data. Due to the enhanced validations, the performance of data load is slower.

    > **Note**
    >
    > When running any of the Workforce Incremental rules (for example, OWP_INCREMENTAL PROCESS DATA WITH SYNCHRONIZE DEFAULTS), ensure that the target option **Enable Data Security for Admin Users** is set to **No**. This option can only be set by an administrator.

    When this option is set to **Yes**, data is validated for administrator and non-administrator data loads in the same manner. Validations include: security checks, intersection validations, read-only cells, dynamic calc cells, etc.

    In addition, a detailed error list for any rows that are rejected or ignored is available and no additional Planning permissions are needed. However, performance may be slower even for administrators.

13. From **Display Validation Failure Messages**, create an output file that shows rejected data cells, intersections, and the rejection reason when you load data.

    The limit for the number of rejections reported is 100. The data validation report is available for download from the Process Details page by clicking the **Output** link. In addition, a copy of the error file is stored in the `Outbox` folder.

    For more information, see Viewing Process Details.

14. (Planning and Oracle General Ledger non-administrator only) From **Replace for Non Admin Load Method**, select the Replace method when loading data to an Planning application for non-administrator users.

    Available methods:

    - None—Performs no replace.
    - Numeric Data Only—Performs replace when the load method is Numeric Data only.

Copyright © 2017, 2026, Oracle and/or its affiliates.

- All Load Methods—Perform replace for all load methods.

> **Note**
>
> When running an integration in Replace mode to an ASO cube, if the scenario member is a shared member, then only a Numeric data load is performed. Be sure to specify the member name with fully qualified name including complete hierarchy. All Data Type load methods does not work when the scenario member is a shared member.

15. In **Drill View from Smart View**, specify the custom view of columns from the Workbench when displaying customized attribute dimension member names in Oracle Smart View for Office drill-through reports.

    When the custom view has been defined, you can click the drill-through cell in Smart View and select **Open as New Sheet**, and the drill-through report opens based on the view defined in the Workbench.

    If no views are defined on the Application Details page, the default view is used, meaning that attribute dimensions do not display customized member names in Smart View.

    For more information about defining a custom view, see Defining a Custom View in the Workbench.

16. Click **Save**.

Data Export Target Application Options

The following options are available when defining the target application options for a data export to file application:

Option	Description
Download File Name	You can specify your own file name when generating a data export output file as part of a data export to file integration definition, for example, **PlanExport.csv**. The system creates the output file in the platform `inbox/outbox` folder. The `inbox/outbox` folder is accessible by navigating to the **Application** card from the Home page, then **Overview**, and then from the **Actions** drop-down, selecting **Inbox/Outbox Explorer**. You can also create the file under Data Integration folders by prefixing the folder to the file name, for example **inbox/PlanExport.csv** or **outbox/PlanExport.csv**. Sub-folders under the `inbox` are supported, for example, **inbox/userx/PlanExport.csv**. You cannot create sub-folders under the `outbox`. If the Download File Name field is left blank, the output file is created in the outbox folder with the name <**Application Name**>_<**Process ID**>**.dat**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Option	Description
Column Delimited	Select the character to use for delimiting columns in the output file. Available column delimiters are: • , • | • ! • ; • : The default delimiter is a comma (,). The tab delimiter for columns in a data export to file target application is not available. If a source application contains members with a comma, you can use a different delimiter (excluding the TAB delimiter) in the import format definition to process the export. For more information, see Creating the Dimension Maps. The output file is created using the column delimiter option value defined in Data Exchange Target Application Options.
Decimal Separator	Specify the symbol used to mark the border between the integer and the fractional parts of a decimal number. Available options: • Decimal Comma • Decimal Point The decimal point (.) is typically used in the United States and other English-speaking countries. The decimal comma (,) is typically used in continental Europe and several Arabic-speaking countries.
File Character Set	Specify the file character set. The file character set determines the method for mapping bit combinations to characters for creating, storing, and displaying text. Each encoding has a name; for example, UTF-8. UTF-8 is the default file character set.
End of Line Character	Select the operating system of the server associated with the End Of Line (EOL) character. Valid options are • Windows • Linux EOL indicates end of line. Some text editors like Notepad do not display files using Linux EOL correctly. For Oracle Fusion Cloud Enterprise Performance Management, Data Integration uses the Linux EOL character as the defaults. When customers views exported files in Windows, the EOL shows on a single line.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Option	Description
Include Header	Determines whether to include/exclude the header record in the output file. Select **Yes** to include the dimension name in the header record. The default is **Yes.** Select **No** to exclude the header record.
Export Attribute Columns	Include attribute columns if you have some static values to include in the export or file. You can also use attribute columns if you don't have a requirement to map the source values. This setting minimizes the need to define data load mapping. Select **Yes** to include attribute columns. Select **No** to exclude attribute columns.
Accumulate Data	Summarizes Account data before export and groups the results by one or more column. Select **Yes** to group the results by one or more columns. Select **No** to not group the results by one or more columns. The default value is **Yes**.
Sort Data	Determine if data is sorted based on the column order or not. Select **Yes** to include columns. Select **No** to exclude columns.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Option	Description
Pivot Dimension	Pivoting changes the orientation of the data in the export file enabling you to aggregate the results and rotate rows into columns. When you pivot between rows and columns, the system moves the selected dimension to the outermost row or column on the opposite axis. To use this feature, specify one dimension name from the export file. Applicable for numeric import formats only. It is not supported for nun-numeric import format types. When the Pivot Dimension is specified, Accumulate Data is forced to "Yes." The pivoted columns for the values contained in the Pivot dimension needs to be enclosed in ~~ within the SQL Insert statement. The Pivot feature works only when using a custom SQL Insert statement and does not work without the SQL Insert statement. When the Data Export application contains ACCOUNT, PERIOD & AMOUNT as dimension names, by specifying PERIOD as Pivot Dimension, all the periods contained in the PERIOD dimension are pivoted to a separate column Jan, Feb, Mar etc. For example, the SQL INSERT statement is: *INSERT INTO TABLE (ACCOUNT,JAN,FEB,MAR, ...,DEC) VALUES (~ACCOUNT~,~Jan~,~Feb~,~Mar~,,,,,,,~Dec~)*

Defining Oracle General Ledger Target Options

When integrating source applications with target applications, you can select options that support how you load date including load methods, date formats, batch, and purge options.

To define target options:

1. From the **Data Integration** home page, click ⋯ to the right of the integration, and then select **Options**.
2. From **Target Option**, then **Purge Data File**, tap the slider on to delete the data file from the application outbox directory when a file-based data load is successful.

   Tap the slider off to retain the file.
3. In **Balance Type**, select the type of balance to which to load data.

   Available balance types:

   - Actual
   - Budget
4. From **Journal Source**, enter a description of the journal source that matches the journal source defined in the Oracle ERP Cloud.
5. From **Journal Category**, enter a description of the journal category that the matches the journal category in the Oracle ERP Cloud.

Copyright © 2017, 2026, Oracle and/or its affiliates.

6. Click **Save**.

Defining Target Options for Budgetary Control Applications

When integrating source applications with Budgetary Control target applications, you can select options on how to load data.

To define target options:

1. From the **Data Integration** home page, click ⋯ to the right of the Budgetary Control integration, and then select **Options**.
2. Select the application options.

Target Option	Description
Purge Data File	Tap the slider on to delete the data file from the application `outbox` directory when a file-based data load is successful. Tap the slider on to delete the file or tap the slider off to retain the file.
Budget Type	Select the type of budget. Available options: • Actual • Budget
Journal Source	Enter a description of the journal source that the matches the journal source in the Oracle ERP Cloud.
Journal Category	Enter a description of the journal category that the matches the journal category in the Oracle ERP Cloud.
Source Budget Type	Select the budget type based on the Source Budget Name dimension in the control budget target application. Available options: • EPM Financials module • Planning

3. Click **Save**.

Defining Application Options for Financial Consolidation and Close

When loading journal entries or data to a Financial Consolidation and Close target application, you can control many aspects of the data that is loaded such as the level of journal information:

To define Financial Consolidation and Close application target options:

1. From the **Data Integration** home page, click ⋯ to the right of an integration with a Financial Consolidation and Close target application, and then select **Options**.
2. From **Options**, then **Target option**, select the target application setting and then click **Save**.

   Available Financial Consolidation and Close target application options are shown in the following table.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 9-5 Financial Consolidation and Close Application Options and Descriptions

Option	Description
Load Type	Select the type of load to push to the Financial Consolidation and Close application. Available options: • Data—Load numeric data only. This is the default load option. • Journal—Load journals.
Journal Status	The journal status indicates the current state of the journal. The status of a journal changes when you create, submit, approve, reject, or post the journal. Available options: • Working—Journal is created. It has been saved, but it may be incomplete. For example, more line items may need to be added. • Posted—Journal adjustments are posted to the database.
Journal Type	Select the type of journal to load. Available options: • Auto-Reversing—Loads an auto-reversing journal that contains adjustments that need to be reversed in the next period. That is, the journal posts in the next period by reversing the debit and credit. • Regular—Load journals using the Replace mode, which clears all data for a journal label before loading the new journal data.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 9-5 (Cont.) Financial Consolidation and Close Application Options and Descriptions

Option	Description
Journal Post As	Select the method for posting journal entries: Available options: • Journal-to-date— A Journal-to-Date journal entry carries forward from period to period, from the first instance of the journal entry, including a carry-forward across any intervening year-ends. The only difference between a Journal-to-Date entry and a Year-to-Date entry is that in the first period of each year, the data from Journal-to-Date entries in the last period of the prior year are reversed. For Year-to-Date entries, there are no reversals in the first period of any year. • Periodic—When you select the View member FCCS_Periodic, when the journal entries are posted, the data entered to the line detail is summarized and posted to the Consol cube based on the line detail POV. The data from one posted journal entry does not overwrite the data written from other posted journal entries. • Year-to-date—When you select the View member FCCS_YTD_Input, you can enter a year-to-date amount in the line detail debit / credit fields. A Year-to-Date journal entry must contain year-to-date entries on all detail lines. When Year-to-Date journal entries are posted, the appropriate periodic impact on the POV across the entries is calculated and then accumulated with any accumulation from posted Periodic journal entries In the first period of any year, the year-to-date View data is the same as Periodic data. In the first period of any year, the year-to-date View data is the same as Periodic data. In subsequent periods, the periodic calculated data posted to the Periodic View member for each unique POV equals the current period year-to-date entries accumulated across all Year-to-Date journal entries, minus the prior period year-to-date entries accumulated across all Year-to-Date journal entries.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 9-5 (Cont.) Financial Consolidation and Close Application Options and Descriptions

Option	Description
Create Drill Region	Tap the slider on to enable the drill region. When this option is enabled, a drillable region is created to use the drill through feature. When loading data, the drill region is loaded to Financial Consolidation and Close data. The drill region by scenarios is created. For any cube (Planning cubes or Planning databases), the name of the drill region is FDMEE_<name of the scenario member>. When creating the drill region, the system checks if a dimension is enabled for the drill. Members of enabled dimensions selected in data loads, are included in the drill region filter. If no dimensions are enabled, the Scenario, Version, Year, and Period dimensions are enabled by default. You can enable additional dimensions, and the subsequent data load considers members of newly enabled dimensions. If you disable any dimensions which were previously included in a drill region used for drill creation, members of such dimensions are not deleted during the subsequent data loads. If needed, you can remove obsolete members manually.
Enable Zero Loading	Select **Yes** to load 0 values during a multiple period load.
Enable Data Security for Admin Users	Enables data validation when an administrative user loads data. In this case, all data validations in the data entry form are enforced while loading data. Due to the enhanced validations, the performance of data load will be slower. When the 'Enable Data Security for Admin Users' is set to **No** (default value), then data loads by the administrator are performed using the Outline Load Utility (OLU). In this case, performance is fast but you are unable to get a detailed error report for any rows that are ignored for any reason. When this option is set to **Yes**, data is validated for administrator and non-administrator data loads in the same manner. Validations include: security checks, intersection validations, read-only cells, dynamic calc cells, etc. In addition, a detailed error list for any rows that are rejected or ignored is available and no additional Planning permissions are needed. However, performance may be slower even for administrators.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 9-5 (Cont.) Financial Consolidation and Close Application Options and Descriptions

Option	Description
Enable Drill from Summary	Select **Yes** to drill down from summary members in a Planning data form or report and view the detail source data that make up the number. After enabling this option and loading the data with the Create Drill Region option set to **Yes**, the Drill icon is enabled at the summary level. Drill is limited to 1000 descendant members for a dimension. Summary level drill downs are not available for the Scenario, Year, and Period dimensions. For these dimensions, you must perform a drill through on the leaf members. Summary drill is available for local service instances only. It is not available between cross service instances or hybrid deployments.
Movement	Specify the movement dimension that indicates the automated cash flow reporting dimension used through hierarchies and system calculations. By default, the system provides members in the Movement dimension to maintain various types of cash flow data, and FX to CTA calculations. If no movement, then specify as **FCCS_No Movement**. Otherwise, select the desired movement member. Example of Movement Dimension Members: • FCCS_No Movement • FCCS_Movements • FCCS_OpeningBalance • FCCS_ClosingBalance
Multi-GAAP	Specify the multi-GAAP dimension used to report your financial statements in both local GAAP and in IFRS or another GAAP. This dimension tracks the local GAAP data input as well as any GAAP adjustments.
Data Source	Specify the data source dimension. The default value is "FCCS_Managed Data." You can change the data source member to your custom data source member as needed.
Purge Data File	When a file-based data load to Essbase is successful, specify whether to delete the data file from the application *outbox* directory. Select **Yes** to delete the file, or **No** to retain the file.
Member name may contain comma	If the member name contains a comma, and you are loading data to one of the following services, set this option to **Yes**, and then load the data: • Planning Modules • Planning • Financial Consolidation and Close • Tax Reporting

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 9-5 (Cont.) Financial Consolidation and Close Application Options and Descriptions

Option	Description
Workflow	Select the data workflow method. Available options: • Full—Data is processed in the TDATASEG_T table and then copied to the TDATASEG table. All four Workbench processes are supported (Import, Validate, Export, and Check) and data can be viewed in the Workbench. Drill-down is supported. The Full Workflow mode is the default mode. • Full No Archive—Data is processed in the TDATASEG_T table and then copied to TDATASEG table. All four Workbench processes are supported (Import. Validate, Export, and Check). Data can be viewed in the Workbench but only after the import step has been completed. Data is deleted from TDATASEG at the end of the workflow process. Drill-down is not supported • Simple— Data is processed in the TDATASEG_T table and then exported directly from the TDATASEG_T. table. All data loads include both the import and export steps. Data is not validated and any unmapped data result in load failure. Maps are not archived in TDATAMAPSEG. Data cannot be viewed in the Workbench. Drill-down is not supported.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 9-5 (Cont.) Financial Consolidation and Close Application Options and Descriptions

Option	Description
Drill View from Smart View	Specify the custom view of columns from the Workbench when displaying customized attribute dimension member names in Oracle Smart View for Office drill-through reports. When drilling into Smart View, Data Integration uses the last used view on the Drill landing page. If no last used view is found, Data Integration uses the default view selection in this setting Custom views are created and defined in the Workbench option in Data Integration. When the custom view has been defined and then specified in the Drill View from Smart View field, in Smart View you can click the drill-through cell and select Open as New Sheet, and the drill-through report opens based on the view defined in the Workbench. If no views are defined on the Application Options page, the default view is used, meaning that attribute dimensions do not display customized member names in Smart View. For more information about defining a custom view, see Defining a Custom View in the Workbench.

Defining Target Options for an Account Reconciliation Application

At the integration definition level for the target application, you can override the **Data format** and **Use comma as decimal separator** options defined at the application detail level.

To set Account Reconciliation target options:

1. From the **Data Integration** home page, click ••• to the right of the integration, and then select **Options**.
2. Select the **Options** tab.
3. From **Date format**, select the format to use when entering adjustment into reconciliations:

   Available options:

   - MM/dd/yyyy
   - MM-dd-yyyy
   - MMM d, yyyy
   - d-M-yyyy
   - dd-MMM-yy
   - dd/MM/yyyy

   By default, the date format is blank, which enables the system to use all date formats to parse date values.
4. From **Use commas as decimal separator**, select **Yes** to use a comma as the decimal separator.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Select **No** to use the user-locale for number parsing.

By default this option is set to **No** (the decimal separator is based on the user-locale and not a comma.)

5. Click **Save**.

Defining Filters

Filters let you qualify results from the source system. For example, when you specify filters for a Planning source system, select a dimension, and then enter filter criteria for each dimension.

> **Note**
>
> Dynamic Time Series (DTS) members (for example, Y-T-D, H-T-D, Q-T-D, etc.,) that are surfaced for forms and reports are not supported for source filters when using Quick Mode, or anywhere that the member selector for the period dimension is supported.

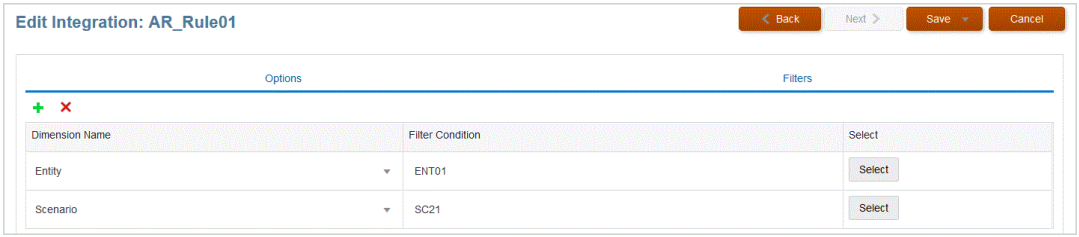
Defining Planning Filters

You can define filter options to specify a subset of data to extract from your Planning application.

To add a Planning filter:

1. From the **Data Integration** home page, click ••• to the right of the Planning integration, and then select **Options**.
2. Click the **Filters** tab.
3. Click ➕.

   Optionally, you can simply select another dimension already assigned to a filter and assign another dimension from the **Dimension Name** drop-down or just change the filter condition.



4. From the **Dimension Name** drop-down, specify the name of the dimension to which to add as a filter.
5. Click **Save**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Defining Oracle NetSuite Filters

Search criteria defined in the NSPB Sync SuiteApp Saved Search are registered automatically as application filters. You edit the filters if needed. For example, you assign default values or hide the filters from end users.

You apply filter conditions to the NSPB Sync SuiteApp Saved Search results so that only those records that meet selected conditions are returned. You can select a single filter condition or multiple filters conditions, and additionally specify the exact values.

Search criteria defined in the NSPB Sync SuiteApp Saved Search are registered automatically as application filters. You edit the filters if needed. For example, you can assign default values or hide the filters from end users.

> **Note**
>
> If you specify a value and operator for a Oracle NetSuite filter in the Oracle Fusion Cloud Enterprise Performance Management, Oracle NetSuite uses the value from the Cloud EPM, but applies the operator for that filter defined in Oracle NetSuite. Any operator defined for the filter in Cloud EPM is ignored by Oracle NetSuite. So if the operator for the filter in Oracle NetSuite is "On or after " and the operator in Cloud EPM is "Equals", Oracle NetSuite use the value from Cloud EPM, but the operator of "on or after ",which was defined in Oracle NetSuite.

To apply NSPB Sync SuiteApp filters:

1. From the **Data Integration** home page, click ••• to the right of the Oracle NetSuite integration, and then select **Options**.
2. Click the **Filters** tab.
3. Select the name of the filter condition.

   Available filters are:

   - **Postingperiod**

     > **Note**
     >
     > If you need to load data by a specific period, for example, by a specific month, use an NSPB Sync SuiteApp posting period filter that contains a period ID. The period ID is created in the posting period filter and returned by the NSPB Sync SuiteApp Saved Search used for filtering data.
     >
     > Oracle NetSuite provides several types of date filters: a named time period, for example, last fiscal year, a custom date range defined by a specific start date and end date, and a relative date range defined by a beginning number of days, weeks, months, quarters, or years ago, or from now, to an ending number of same. See the NetSuite "Search Guide" for more information on setting up filters.

   - **Mainline**
   - **Posting**

Copyright © 2017, 2026, Oracle and/or its affiliates.

4. From the **Condition** drop-down, select the applicable condition.

   The applicable conditions by filter are derived from the NSPB Sync SuiteApp Saved Search.

5. From the **Value** drop-down, select the value to which to apply the filter.

6. Click **Save**.

Defining Budgetary Control Filters

When an integration from a Budgetary Control data source is created, filters are defined automatically. You can modify the filters as needed but cannot delete them. (If the filters are deleted, the default value is recreated.)

For data integrations used to import data from the Budgetary Control, use filters to limit the results.

When an integration from a Budgetary Control data source is created, filters are defined automatically. You can modify the filters as needed but cannot delete them. (If the filters are deleted, the default value is recreated.)

When working with Budgetary Control data load filters, note the following:

- Control Budget— Add this dimension and select the control budget from which to load consumption balances.
- Amount Type budgetary control dimension—Select "PTD" assuming the line items to be loaded from Budgetary Control are typically income statement type of accounts.
- Balance Amounts budgetary control dimension—Depending on your business requirements for the type of consumption amounts you want to bring into the Planning application, select any balance amount(s), such as Commitment, Obligation, Other Anticipated Expenditures, and Expenditures.
- For the other budgetary control dimensions, select members from the appropriate level to which to load data in the Planning application.

To add a Budgetary Control filter:

1. From the **Data Integration** home page, click ••• to the right of the Budgetary Control integration, and then select **Options**.
2. Click the **Filters** tab.
3. Click ➕.

   Optionally, you can simply select another dimension already assigned to a filter and assign another dimension from the **Dimension Name** drop-down or just change the filter condition.
4. From the **Dimension Name** drop-down, specify the name of the dimension to which to add as a filter.
5. In **Filter Condition**, specify the filter.
6. Click **Save**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Defining Oracle HCM Cloud Filters

Any Oracle Human Capital Management Cloud source filters associated with the data source are created automatically during the integration. You can select any specific criteria on the Edit Integrations page to filter the results.

For data integrations used to import data from the Oracle HCM Cloud to Planning Modules or Workforce, use filters to limit the results.

Any source filters associated with the data source are created automatically during the integration. You can select any specific criteria on the Edit Integrations page to filter the results.

Depending on the Oracle HCM Cloud metadata category, the following source filters are populated automatically:

- **Effective Date**—Select the date on which you want the trees to be effective.
- **Legislative Data Group**—Legislative data groups are a means of partitioning payroll and related data. At least one legislative data group is required for each country where the enterprise operates. Each legislative data group is associated with one or more payroll statutory units.
- **Tree Code**— Tree code for hierarchy in Oracle HCM Cloud (for objects with hierarchy, for example: Org, Position)
- **Tree Version**—Tree Version for hierarchy in Oracle HCM Cloud
- **Changes Only**—Controls the extract mode. Valid options are **N** or **Y**.
  The following table describes the different extract modes, their lookup values and descriptions:

Mode	Lookup Value	Description
N	All attributes	Includes all data in the extract. A full extract is run which produces the full data output at that point of time. The archived data is utilized as a baseline.
Y	Changed attributes	Compares this extract run with the previous extract runs and by comparing against the baseline (to identify the incremental data), displays the data that has changed only.

To edit an Oracle HCM Cloud filter:

1. From the **Data Integration** home page, click ••• to the right of the Oracle HCM Cloud integration, and then select **Options**.
2. Click the **Filters** tab.
3. Select the name of the filter to edit.
4. In **Value**, specify the filter condition.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Options		Filters
Name	Condition	Value
Changes Only	==	
Legislative Data Group	==	
Effective Date	==	2017-01-01

5. Click **Save**.

Defining Custom Filters to a MDX Query

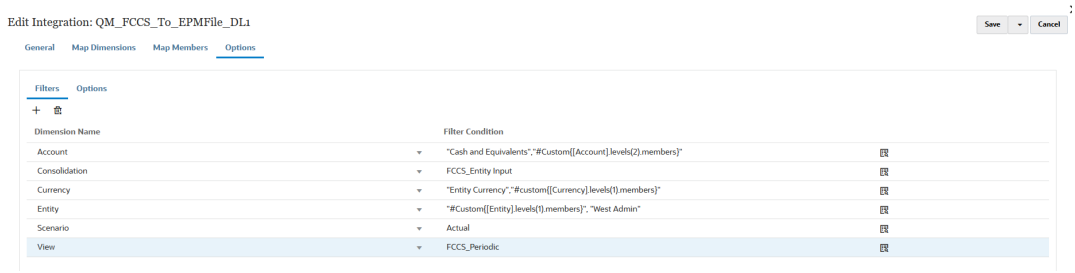
You can specify a custom filter to qualify valid MDX member selections on the Filters tab in Source Options. Custom filters enable you to select members not available on the Member Selection page.

To add a custom filter to a MDX Query:

1. From the **Data Integration** home page, click ••• to the right of an integration, and then select **Options**.
2. Click the **Filters** tab.
3. Select the dimension to which to apply the custom filter.
4. In **Filter Condition**, use MDX syntax to define the custom filter by enclosing members with braces ({ and }) characters and then click **Save**.

   The syntax for the custom filters is: `"#Custom{<Valid MDX Member Selection Function>}".`

   The following example shows custom filters applied to the "Account," "Currency," and "Entity" dimensions:



   In Process Details, you can view the details of the custom filters applied to the MDX query:

```
2023-06-26 18:21:34,149 INFO  [AIF]: -------EXECUTION LOAD PARAMETERS-------
2023-06-26 18:21:34,157 INFO  [AIF]: Account (SOURCE_FILTERS)                 : "Cash and Equivalents","#Custom{[Account].levels(2).members}"
2023-06-26 18:21:34,157 INFO  [AIF]: Consolidation (SOURCE_FILTERS)           : FCCS_Entity Input
2023-06-26 18:21:34,157 INFO  [AIF]: Currency (SOURCE_FILTERS)                : "Entity Currency","#custom{[Currency].levels(1).members}"
2023-06-26 18:21:34,157 INFO  [AIF]: Entity (SOURCE_FILTERS)                  : "#Custom{[Entity].levels(1).members}", "West Admin"
2023-06-26 18:21:34,157 INFO  [AIF]: Scenario (SOURCE_FILTERS)                : Actual
2023-06-26 18:21:34,158 INFO  [AIF]: View (SOURCE_FILTERS)                    : FCCS_Periodic
2023-06-26 18:21:34,158 INFO  [AIF]: DataExportDynamicCalc (SOURCE_OPTIONS)   : 2
2023-06-26 18:21:34,169 INFO  [AIF]: Source Application Type is:HPL
2023-06-26 18:21:34,174 INFO  [AIF]: Source Application Sub Type is:FCCS
2023-06-26 18:21:34,174 INFO  [AIF]: Target Application Sub Type is:DIRECTFILEEXPORT
```

Defining A Clear Region

Select a slice of data to clear before loading data in Replace mode to a Planning module application. Additionally, you can also clear prior budgets before loading a new budget to the Oracle General Ledger (available for customers using Oracle Financials Cloud 20C or higher.)

The Clear Region option in Data Integration lets you select the slice of data to be cleared before loading data in Replace mode to a Planning module application. In addition, when

Copyright © 2017, 2026, Oracle and/or its affiliates.

loading budgets to the Oracle General Ledger, you can optionally clear prior budgets before loading a new budget. This feature can only be used with Oracle Financials Cloud 20C or higher.

You define the clear region by specifying member selection criteria for dimensions such as a static list of members, member function selections, and members derived from the data. By default, the system defines a default clear script based on Entity, Scenario, Version, Year, and Period values included in the data load. For the Oracle Financials Cloud, you cannot specify member functions. You can select parent member and system clear data for all lowest level children automatically. When the integration or data load rule is executed, the combination of dimensions defined on the Clear Region are cleared before loading data.

For Planning, the Period, Year, and Scenario dimensions are derived automatically based on the data and you do not have to specify any filters for these dimensions. If you specify member for these dimensions, they are ignored.

For Oracle General Ledger, the Ledger, Scenario, and Accounting Period dimensions are derived automatically based on the data and you do not have to specify any filters for these dimensions. If you specify member for these dimensions, they are ignored.

> **Note**
>
> The **Replace for Non Admin** setting in Application Options determines the Replace method available for non-administrator users. These methods include: none, numeric data only, and All Load Methods. For more information, see Defining Direct Integration Options.

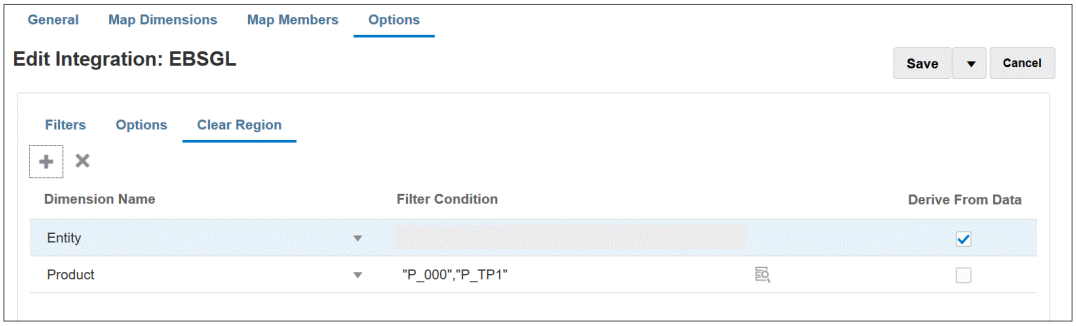
> **Note**
>
> The Clear Region feature is not available for Financial Consolidation and Close, Tax Reporting, Account Reconciliation, and Profitability and Cost Management business processes. It is also not available for
> Budgetary Control (XCC prefixed applications.)

> **Note**
>
> When running an integration in Replace mode to an ASO cube, if the scenario member is a shared member, then only a Numeric data load is performed. Be sure to specify the member name with fully qualified name including complete hierarchy. The All Data Type load methods does not work when the scenario member is a shared member.

To define a Clear Region:

1. From the **Data Integration** home page, click ••• to the right of the Planning application, and then select **Options**.
2. On the **Options** page, click the **Clear Region** tab.

Copyright © 2017, 2026, Oracle and/or its affiliates.



3. From **Dimension Name**, click + to add a dimension from which to add a clear region.

   You can also click ▼ to the right of a dimension and select a dimension from the drop-down.

4. Select the method for clearing a region.

   To clear a region by building a member list based on the data available to export to the target application, go to **step 5**.

   To clear a region by select members, for example by Descendants of Revenue, go to **step 6**.

5. Select the dimension and click **Derive from Data** to clear data for a selected dimension and build the member list based on the data available to export to target.

   If you select **Derive from Data** method for clearing a region, do not specify the any filters in the **Filter Condition** field.

   This option is useful when you want to clear data for a Company or Cost Center to which you are posting data, but you don't want to change the values for each run.

6. To clear data based on a filter or use the member selector, specify the filter or member selector in the **Filter Condition** field.

> **Note**
>
> Functions are not available for General Ledger. You can select a Parent member and system clears values for all children.

7. To view the detail of the clear region feature, after running the integration of data load rule, go to Process Details for job and open the log.

```
2020-06-04 07:22:53,890 INFO  [AIF]: EssbaseRuleFile.getReplaceDataScript - START
2020-06-04 07:22:53,914 DEBUG [AIF]: dimName: Entity, dimClassName: Entity, balColName: ENTITY, filter: DERIVE_FROM_DATA
2020-06-04 07:22:54,118 DEBUG [AIF]: dimName: Period, dimClassName: Period, balColName: null, filter: DERIVE_FROM_DATA
2020-06-04 07:22:54,121 DEBUG [AIF]: dimName: Product, dimClassName: Generic, balColName: UD3, filter: "P_000"
2020-06-04 07:22:54,121 DEBUG [AIF]: dimName: Scenario, dimClassName: Scenario, balColName: null, filter: DERIVE_FROM_DATA
2020-06-04 07:22:54,143 DEBUG [AIF]: dimName: Version, dimClassName: Version, balColName: UD1, filter: DERIVE_FROM_DATA
2020-06-04 07:22:54,344 DEBUG [AIF]: dimName: Year, dimClassName: Year, balColName: null, filter: DERIVE_FROM_DATA
2020-06-04 07:22:54,348 INFO  [AIF]: Replace data script:
```


Copyright © 2017, 2026, Oracle and/or its affiliates.

Using Business Rules

Launch and manage sophisticated rules when loading data to Planning applications by invoking rule logic without programmer assistance and without a need for an additional tool like EPM Automate.

Business rules enable you to launch and manage sophisticated rules when loading data to Planning applications. This feature empowers you to invoke rule logic without programmer assistance and without a need for an additional tool like EPM Automate. You can run business rules for Numeric Data Only and All data type load methods. In addition, you can run an individual business rule or rule set. A rule set enables you to combine multiple rules to be executed simultaneously or sequentially.

Examples of business rules include:

- clear data before a data load
- perform calculations
- aggregate data
- copy data to one slice of a cube

> **Note**
>
> Business rules are registered only for a Planning target application

You register business rules at the application or integration level. Business Rules registered at the application level can be executed for all integrations loading data to the application. At the integration level, you register business rules to be executed for a selected integration on the Options tab.

To begin using a business rule, you go to a new Business Rules page in Applications and then register the business rule on either the Independent or Embedded mode.

Use the Independent mode to register rules that can be run in response to system events in the Data Integration workflow. The registration includes the system Event, rule or rule set name, any runtime parameters, and run sequences.

The Embedded mode is used to register business rules that are executed after the data load process. This mode is embedded as part of the data load process and provides access to the data intersections loaded. This mode is available only when using the All Data Type load method. Embedded rules are not triggered by an Event and do not support runtime parameters.

You can execute a business rule that runs a calculation script. For more information on designing and creating business rules using calculation script, see Designing Business Rules in *Designing with Calculation Manager* .

You can execute a business rule that runs a Groovy script. For more information on designing and creating business rules using Groovy script, see Creating a Groovy Business Rule in *Designing with Calculation Manager* .

Registering a Business Rule in Independent Mode

Register business rules in independent mode to run a rule in response to a system event.

Copyright © 2017, 2026, Oracle and/or its affiliates.

To register a business rule in independent mode:

1. To register a business rule for all integrations loading to an application, from the **Application** page, click ••• to the right of the application, and then select **Business Rules**.

   or

   To register a business rule for given integration, from the **General** page, click ••• to the right of the integration, then select **Options**, and then from the **Edit Integration** page, select the **Business Rules** tab.

   > **Note**
   >
   > If a given Event has a business rule registered at the integration and application level, only the business rule on the integration level is triggered.

2. From the **Business Rules** page, click the **Independent** mode.
3. - Before Extract
   - Before Import
   - After Import
   - Before Load
   - After Load
   - Before Check
   - After Check

   For a description of the system events that can be used to trigger business rules in Data Integration, see Business Rules Supported Events.
4. From **Type**, select if the business rule to be run is an individual rule or belongs to a ruleset for any given Event.

   Available types:

   - Rule—Identifies a specific business rule that can run for any given Event.
   - Ruleset—Identifies a business rule that belongs to a ruleset. A ruleset provides a unit of execution when multiple rules are included in a set of rules to be run. In rulesets, the priority of rules applies to specify the order of firing of the rules in the ruleset. Rulesets are used to launch rules sequentially or simultaneously depending on your logic requirements.
5. From **Name**, select the business rule to run.

   The business rules available in this field are prebuilt in Calculation Manager and then made available to Data Integration when loading data to your Planning target applications.

   If a business rule name is changed in Planning, the new business rule name is not automatically renamed in Data Integration, which does not validate business rules names and logic.

6. In **Run Time Prompts**, click to add or edit any run time prompts.

   For more about working with run time prompts, see Adding Run Time Prompts to Business Rules.
7. In **Sequence**, specify the order in which to run the rule when multiple rules or rulesets have been associated with a single event.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Since multiple scripts can be executed for an event, this sequence value provides the numeric order in which each business rule is executed. You can enter any number, but the number must be unique within an event.

Business Rules Supported Events

The following system events are available to trigger business rules in independent mode. System events are stored in the `bin/scripts/event` folders.

Event	Description
Before Extract	This event is triggered before data is extracted from the source application. For example, you might use this event to run a calculation such as aggregating data or copying from a BSO cube to an ASO cube, use the Before Extract event before extracting data for data synchronization or data export.
Before Import	This event is triggered before the data is imported into the staging table. For example, you might use this event to run a business rule to generate a data file for processing.
After Import	This event is triggered after the data is imported into the staging table.
Before Load	This event is triggered before the data is loaded to the Oracle Fusion Cloud Enterprise Performance Management application. For example, you might use this event to run a custom clear script to clear the data before loading the new data.
After Load	This event is triggered after the data is loaded to the Cloud EPM application. For example, you might use this event to perform any calculations after the data is loaded.
Before Check	Called before the Check Rule.
After Check	This is event is called after the Check step and is the final event. For example, you might use this event to trigger a business rule that copies data to another version or to trigger another business rule to perform other downstream tasks.

Adding Run Time Prompts to Business Rules

You can enter or edit values for run time prompt variables when deploying business rules. Run time prompts enable you to parameterize a rule by identifying and defining the list of values used in the rule. If a business rule includes run time prompts, users enter the required information before the business rule is launched. Business rules can prompt users for such variables as members, text, dates, or numbers. Prompts should be specific and tell users what type of data is expected. For example, you could add a run time prompt that enables a user to select a specific "Entity" when running a rule.

You can select a run time prompt value from a predefined list of values, such as the POV Period.

You can also specify custom parameters, in which case, you can explicitly state the variable or enter any Oracle Essbase filter syntax.

Additionally, you can reference values stored in the Integration Option 1-4 fields on the Options page to drive calculation logic.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> Run time prompts are not validated by Data Integration.

To add a run time prompt for a business rule:

1. From the **Business Rules** page, click **Independent**.
2. Click .
3. Select the business rule to which to add a run time prompt.
4. In **Run Time Prompts**, click to add or edit any run time prompts.
5. From the **Run Time Prompt** page, click .
6. In **Name**, type the name of the run time prompt as it has been defined in the business rule (in Planning).

   For example, type **Entity** as the run time prompt name.

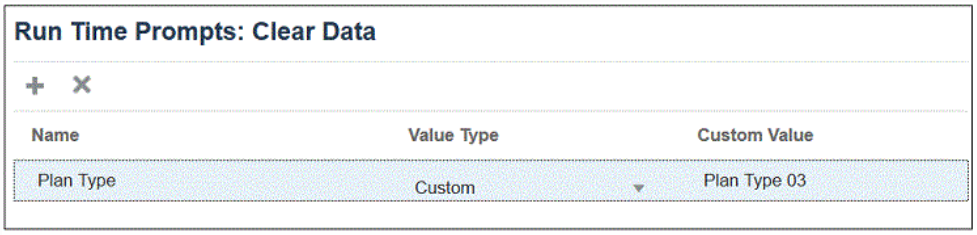
   A LOV (list of values) is unavailable at this time.
7. From **Value Type**, select the type of value expected for the run time prompt:

   Available value types used to construct the run time prompt based on the following:

   - Custom—a custom parameter specified in the **Custom Value** field. You can also use a custom value based on Essbase filter syntax. For more information, see Working with Essbase Calc Script.
   - POV Period—a predefined "POV Period" value based on the Period the integration is running
   - POV Year—a predefined "POV Year" value based on the Period the integration is running
   - POV Category—a predefined "Scenario" value based on the POV Category from the integration (data rule) definition.
   - All Data Load Entity—a list of entities from the data load
   - All Data Load Account—a list of accounts from the data load
   - Integration Options 1 - 4—any free form text or value saved to the Integration Option 1 - 4 fields in General Options. For more information, see Defining Direct Integration Options.

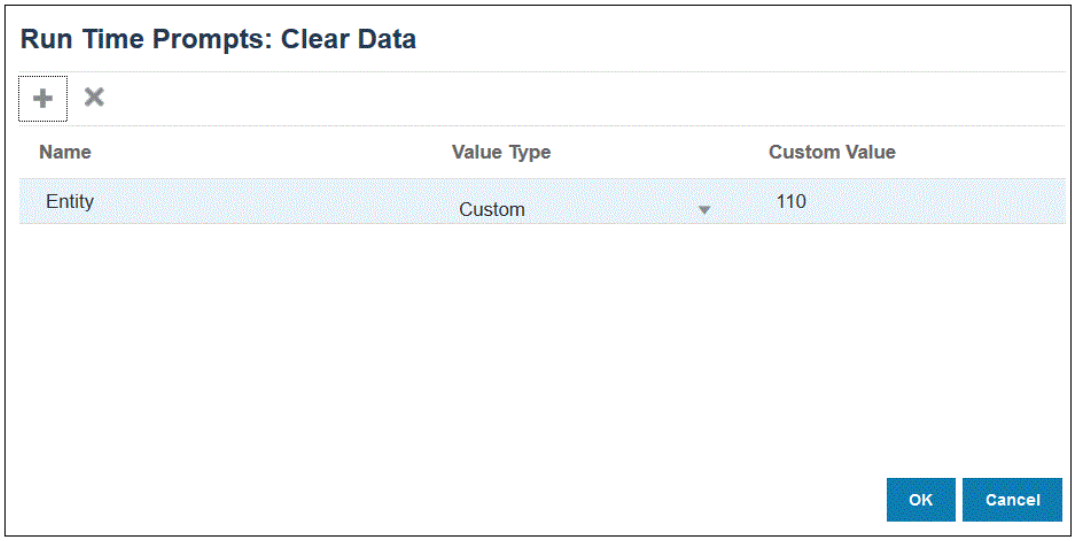
Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> By default, business rules are executed for the Target Plan Type specified in the integration. If you want to execute the business rule for a different Plan Type, specify a run time prompt named "Plan Type," specify a "custom value type," and then provide the name of the desired Plan Type as the custom value.
> The following example shows how to specify a Plan Type other than default Plan Type associated with the integration for a Clear Data business rule.



8. If you selected a custom value type for a run time prompt, specify the actual value in the **Custom Value** field.
9. Click **OK**.

   In this example, a custom run time prompt has been defined for a Clear Data business rule that prompts for an Entity.



You can view the details of the business rule in Process Details.

```
2020-09-24 19:06:47,108 DEBUG [AIF]: DEBUG:Parameters:{"jobName":"Clear Data","jobType":"RULES","parameters":{"Entity":"\"110\"","planType":"VisASO"}}
2020-09-24 19:06:47,108 INFO  [AIF]: Calling Rest service to submit the job for event:BefImport, rule:Clear Data, type:RULES
2020-09-24 19:06:48,241 INFO  [AIF]: Submitted Rest service job, process Id: 4
2020-09-24 19:06:48,454 INFO  [AIF]: Executing rule:Clear Data completed.
```

Registering a Business Rule in Embedded Mode

The Embedded mode is used to register business rules that are executed after the data load process. This mode is embedded as part of the data load process and provides access to the data intersections loaded. This mode is available only when using the All Data Type load method. Embedded rules are not triggered by an Event and do not support runtime parameters.

To register a business rule in embedded mode:

Copyright © 2017, 2026, Oracle and/or its affiliates.

1. To register a business rule for all integrations loading to an application, from the **Application** page, click ••• to the right of the application, and then select **Business Rules**.

   or

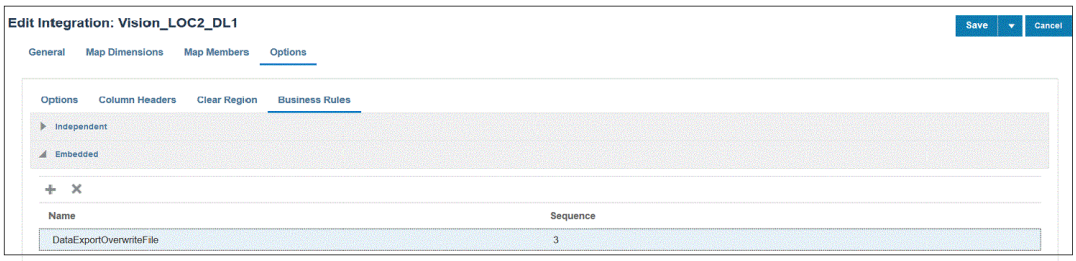
   To register a business rule for a given integration, from the **General** page, click ••• to the right of the integration, then select **Options**, and then from the **Edit Integration** page, select the **Business Rules** tab.

   > **Note**
   >
   > If a given Event has a business rule registered at the integration and application level, only the business rule on the integration level is triggered.

2. From the **Business Rules** page, click the **Embedded** mode.
3. Click ➕.
4. From **Name**, specify the business rule to run.

   The business rules available in this field are prebuilt in Calculation Manager and then made available to Data Integration when loading data to your Planning target applications.

   If a business rule name is changed in Planning, the new business rule name is not automatically renamed in Data Integration, which does not validate business rules.
5. In **Sequence**, specify the order in which to run the rule when multiple rules or rulesets have been associated with an integration or application.



6. Click **Save**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

10

Defining Data Maps

Create data mappings between a source application and a target reporting application.

Related Topics

- Understanding Data Maps
  Move data, comments, attachments, and supporting detail from source cubes and smart lists to a target application (either locally or remotely).
- Creating Data Maps
  Shows how to create, manage, refresh, move mappings between source and target applictions.
- Refreshing Data Maps
  Shows how to refresh data maps.
- Managing Data Maps
- Moving Data
- Checking Real Time Progress Status for Data Maps and Smart Push
  View comprehensive information about each job's operations, including the completion status and duration in Jobs.

Understanding Data Maps

Move data, comments, attachments, and supporting detail from source cubes and smart lists to a target application (either locally or remotely).

Use Data Maps to move data, comments, attachments, and supporting detail from source cubes and smart lists to a target application in a local or remote target application. You can map dimensions between a source cube and target cube to enable to push data:

- within Planning or a FreeForm application
- between Planning applications
- **Note**

  Smart lists are not supported when you push from Planning to Enterprise Profitability and Cost Management, Financial Consolidation and Close and Tax Reporting.
- between Strategic Modeling and Planning
- between an Enterprise Profitability and Cost Management source and Enterprise Profitability and Cost Management target in the same environment.

A cross application data map job is not allowed when the target application is either Financial Consolidation and Close or Tax Reporting.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> Data Maps support an aggregate storage cube (ASO cube) as a source. Data Maps and Smart Push ignore rows when an entire record consists of zeros, missing values, or a combination of missing and zero values.

Videos

Your Goal	Watch This Video
Learn how to move data using data maps.	Instantly Push Data for Reporting in Oracle Planning and Budgeting Cloud

Related Topics

- About Applications
- About Reporting Cubes
- Unsupported Features with Data Maps

About Applications

An application is a related set of dimensions and dimension members used to meet a set of data needs. Each application has its own accounts, entities, scenarios, and other data elements.

For more information about Planning applications see Managing Applications in *Administering Planning*.

For more information about Enterprise Profitability and Cost Management applications, see Creating an Enterprise Profitability and Cost Management Application in *Administering and Working with Enterprise Profitability and Cost Management*.

About Reporting Cubes

A reporting cube is a cube on which you can report on and aggregate Planning data. The primary use cases:

- The reporting cube contains data from any source, such as a data warehouse. You want to move new Planning data to it and report on the data.
- You want to report on Smart Lists in Planning, which you can't do in Planning.

Unsupported Features with Data Maps

Mapping a Planning application to a reporting application doesn't support:

- Aggregate storage database outlines with the "Duplicate Members Allowed" option selected
- Attribute dimensions
- Attribute member selections

Copyright © 2017, 2026, Oracle and/or its affiliates.

Creating Data Maps

Shows how to create, manage, refresh, move mappings between source and target applictions.

To create a data map that maps an application to a reporting application:

1. Click **Application,** then **Data Exchange**, and then click **Data Maps**.
2. Click **Create**.
3. Enter a name and description for the data map.
4. Make your **Source** and **Target** selections as follows:
   - Under **Source**, select a cube from the available cubes for the current application.
   - Under **Target**, select an application and a cube from the available applications and cubes for the target reporting application.

   > **Note**
   >
   > Tax Reporting is currently not supported as a **Target**.

   The current mappings for the selected source and target cubes are displayed under **Source** and **Target**, and any unmapped dimensions are displayed under **Unmapped Dimensions**.

   For information about creating and moving cross application data map job, see: Moving Data Using a Cross Application Data Map.

   > **Note**
   >
   > If you are creating a data map to push data between Strategic Modeling and Planning, you see additional **Source** and **Target** options:
   >
   > - For **Source**, the options are **Select Model** and **Select Scenario Rollup**. Select a model or consolidation, and then click **Select**.
   > - For **Target**, if a model is your **Source**, then select a target Planning cube or target **Models**. If a scenario rollup is your **Source**, then select a target Planning cube or the scenario rollup.
   >
   > If you don't see your Strategic Modeling model in the list, from the **Actions** menu, select **Synchronize**.

5. **Optional**. Change the current mappings and map any unmapped dimensions.

   To change the current mappings, see Changing Mapping Information for a Dimension.

   To map unmapped dimensions, see Moving Unmapped Dimensions.
6. Define data map options.

   See Defining Data Map Options.
7. Click **Save and Close**.

   The new data map is added to the end of the data maps list.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Refreshing Data Maps

Shows how to refresh data maps.

To refresh data maps:

1. Click **Application,** then **Data Exchange**, and then click **Data Maps**.
2. Click **Refresh**.

Managing Data Maps

Related Topics

- Viewing Data Maps
- Changing Mapping Information for a Dimension
- Using Detailed Member-Level Mapping
  Use detail member-level mapping to include selected members of a dimension while moving data from one cube to another cube when working with exceptions.
- Moving Unmapped Dimensions
- Populating Detail Mappings
  Shows how to populate all members for all dimensions in the source that do not exist in the target.
- Excluding Source Members from a Data Map
  Shows how to exclude source members of a dimension while moving data from one cube to another cube.
- Defining Data Map Options
  Shows how to define data map settings and options.
- Editing Details for a Data Map
- Duplicating a Data Map
- Exporting a Data Map to Excel
  Shows how to export a data map to a dynamic workbook in Excel.
- Importing a Data Map
  Shows how to import an external data map in Excel format to a data map.
- Deleting a Data Map
  Shows how to delete a data map.
- Validation Rules for Selected Members
  Explains validation rules for selected members.

Viewing Data Maps

To view data maps:

1. Click **Application**, then **Data Exchange**, and then click **Data Maps**.
2. Search, filter, or sort the listing by performing a task:
   - To search for a map by name, enter text in **Search**, and then click .

Copyright © 2017, 2026, Oracle and/or its affiliates.

- To filter maps by source and target, click **All Sources** and **All Targets** and select filter criteria.
- To sort items in each column by ascending or descending order, hover your cursor over a column heading, and then select **Sort Ascending** or **Sort Descending** .

3. Perform a task:

- Create a data map. See Creating Data Maps.
- Refresh the data map listing. See Refreshing Data Maps.
- Change mapping information. See Changing Mapping Information for a Dimension.
- Map unmapped dimensions. See Moving Unmapped Dimensions.
- Define data map options. See Defining Data Map Options.
- Edit details for a data map. See Editing Details for a Data Map.
- Duplicate a data map. See Duplicating a Data Map.
- Delete a data map. See Deleting a Data Map.
- Synchronize Smart Lists in reporting applications. See Synchronizing Smart Lists in Reporting Applications in *Administering Planning*.
- Set relational data merge options. See Defining Data Map Options.
- Review validation rules for default members. See Validation Rules for Selected Members.

Changing Mapping Information for a Dimension

To change the mapping information for the dimensions in a data map:

1. View the **Data Maps** listing page.

   See Viewing Data Maps.

2. Click an existing data map.

3. Select a dimension under **Source**, and then click to select a new source dimension or Smart List to use for the mapping.

   If all dimensions are mapped, you can only map an existing source dimension to a Smart List (only "Smart List" is displayed when you click). To unmap a dimension so that it can be mapped to another dimension, select the dimension under **Source**, and then click . The dimension is moved to **Unmapped Dimensions**.

4. **Optional**: If needed, exclude source members from the data map.

   For more information, see Excluding Source Members from a Data Map.

5. Select a dimension under **Target**, and then click to select a member for mappings.

   For most data maps the target selection does not need any changes.

6. **Optional**: If there are more members and the requirement is to restrict the members in the Target Selection from member mapping, then modify the Target Selection.

   For more information, see Clearing and Restricting Target Data Selections for Mappings.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Considerations for Target Selections

Note the following behavior of target selections:

- Data push and smart push are driven based on the source selections.
- The target selection for mapped dimensions is used for populating the target members for detailed member to member mappings.
- Based on the source selection and member mappings, data can be moved to target members outside the scope of the target section. For example, if the source selection in ILvl0Descendants(YearTotal) and target selection is ILvl0Descendants(YearTotal) with a target exclusion as Dec, the detailed mapping page populates Dec in the source with no member in the target. When you save the data map, the systems warns that Dec is not mapped and the data map may not work correctly. If Dec from the source is mapped to some other member in the target, then data is moved to that member instead on Dec.
- If you don't want to move data to a certain member in the target, then you must define this by either using an exclude member on the source selection or member mappings.
- The target clear region on the target is derived from the source selection and mapping.
- If you want to restrict selections in the target region to specific members and you can't achieve this by way of source selection, use the **Use target selection to clear and restrict data** option. This option enables you to exclude members in the target region.
- If needed, you can revert the target selections to the default option by using the **Use target selection only for mapping** option. In this case, the exclusion selection defined on the target selection for the mapping group is removed.
- When there are multiple dimensions in the target, then the data map can only be run with the clear option if the **Use target selection to clear and restrict data** option is selected for multiple dimensions.
- When the **Use target selection to clear and restrict data** is selected, then the selection for each of the target dimensions with this option is used "as is " to define the target region. Source selection and mappings are not considered for the clear.
- As the target selection is used "as is" without any changes, then the target selection has to be defined correctly to exclude dynamic calc members for an ASO cube.
- If the target selection is a dimension name, then the data of all its descendants is included. For example, when all months must participate in the clear, then the target selection should be ILvl0Descendants(YearTotal) or ILvl0Descendants(Period) based on the business use case, instead of Period.

Clearing and Restricting Target Data Selections for Mappings

In the Target mapped dimensions of a data map, a **Use target selection for clear and restrict data** condition enables you to clear and restrict data to selected member(s) in the target region when moving data.

You might use this option in cases where the outline of source is different from that of target and there is a requirement to control the target region. You simply pick the target selection to clear and then select the **Use target selection for clear and restrict data** condition. When the push is executed, the system does not allow data to be written outside the target region.

Additionally, you can enable an **Add Exclusion** condition to exclude a member on the Target when the **Use target selection for clear and restrict data** condition is selected. In this case, the system does not allow data to be written to the excluded member.

Copyright © 2017, 2026, Oracle and/or its affiliates.

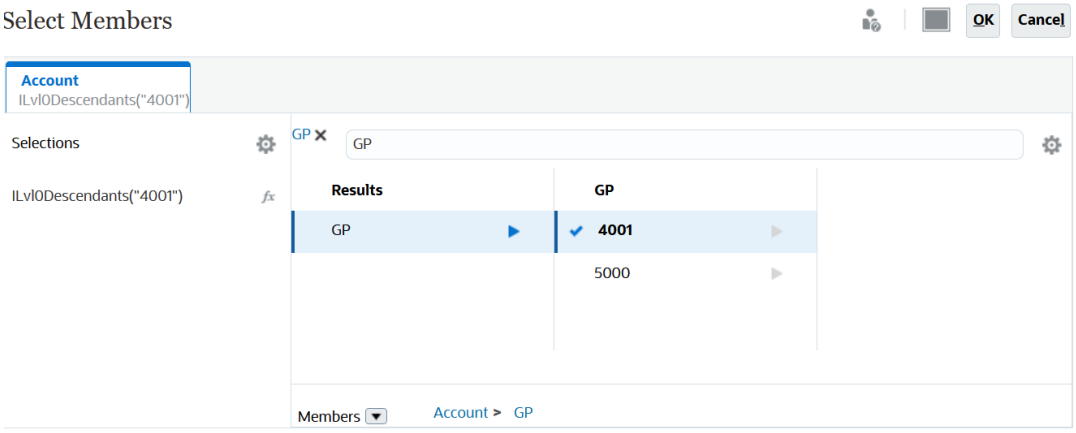
> **Note**
>
> Use this option with caution as the target selection is used as-is for performing the clear option when the data push is run. The clear option occurs regardless of the source selection which determines the region for which the data is moved.
> When a source dimension is mapped to multiple dimensions in the target, then the data push with the clear option is supported only when this option is used for all mappings with multiple dimensions in the target. For example, if the Years dimension in the source is mapped to Years and the Period dimension in the target; the source selections are FY24, FY25; the target selection for Period is Dec,Jan; and the target selection for year is FY24,FY25, then the clear option clears the regions: FY24 Dec, FY24 Jan, FY25 Dec and FY25 Jan.

> **Note**
>
> If a data map uses the **Use target selection for clear and restrict data** condition for a target dimensions, then the corresponding dimension in the Smart Push definition should not check "Use Form Context" or specify a selection in the Overwrite Selection.

To clear and restrict a target region:

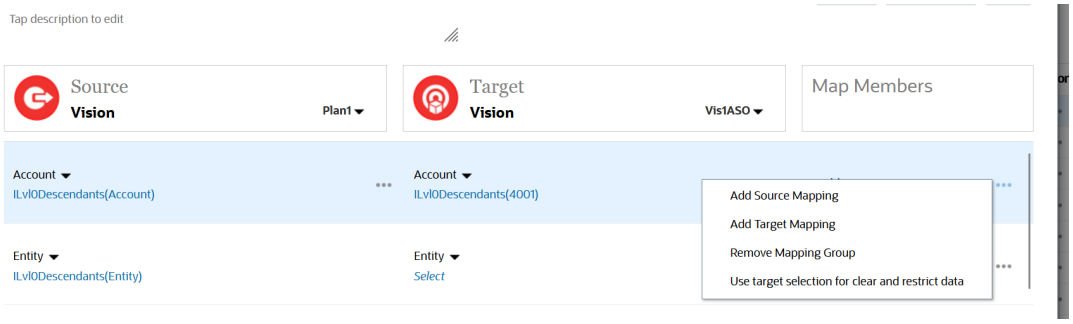
1. Open **Data Exchange**, and then click **Data Maps**.
2. From the **Data Maps** listing page, select the data map in which to clear and restrict the target region.
3. In the Target Mapped Dimension(s), click **Select** under the dimension to which to apply the clear and restrict condition. Then from the **Select Members** page, select the member(s) to use for the clear and restrict option and then click **OK**.

   For example, you might select Gross Profit (GP) for the clear and restrict condition on the 4001 level 0 descendants as shown below:

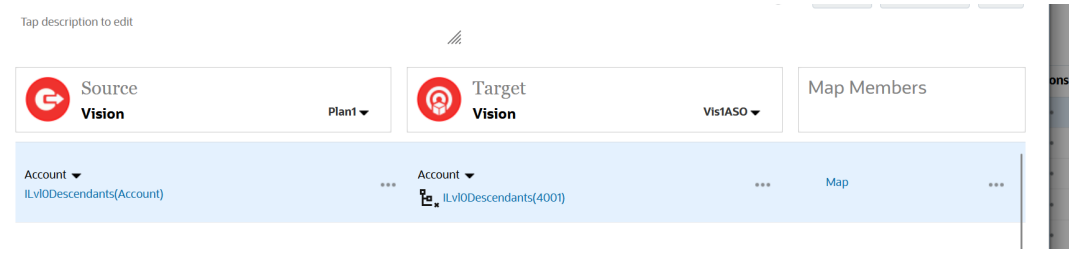


4. Click **Mapping Actions** (•••) next to the target member and then select **Use target selection for clear and restrict data**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



When a target selection has been flagged for a target selection clear and restrict, a icon appears to the left of the flagged member as shown below:



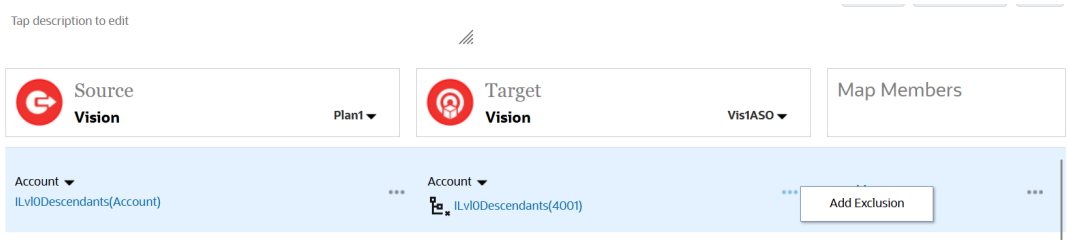
In the example above, the data map clears only the level 0 descendants of 4001.

5. **Optional**: To remove a targeted selection clear and restrict condition, selected the dimension and from **Mapping Actions** (**…**), select **Use target selection only for mapping**.

   If you remove the **Use target selection for clear and restrict data** condition where an exclude condition is also used, both the **Use target selection for clear and restrict data** and **Add Exclusion** conditions are removed.

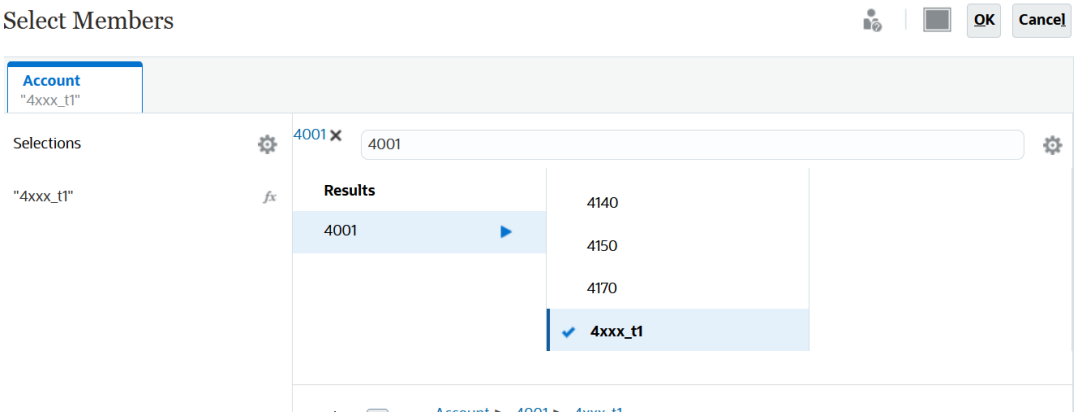
6. To exclude a member from a target that has a clear and restrict condition, click **Target Actions** (•••) and then select **Add Exclusion**.

   The **Add Exclusion** condition on the target is only available when the target has the option clear and restrict data enabled.



7. On the **Select Members** page, select the member to exclude from the push and then click **OK**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



When a target member has been excluded from mapping, a icon appears to the left of the flagged member.

To remove an exclusion, select the member, and then click **Target Actions** ••• , and then select **Remove Exclusion**.

Using Detailed Member-Level Mapping

Use detail member-level mapping to include selected members of a dimension while moving data from one cube to another cube when working with exceptions.

You use detail member-level mapping to include selected members of a dimension while moving data from one cube to another cube when working with exceptions. That is, member-level mappings do not need to be defined when source and target members are the same or do not require any roll-up member mappings or transformations. Including detail member-level mapping while moving data is useful when you need to define complex mappings to or from multiple dimensions or from a single dimension between members with different names. Mapping is required when the members in the source cube do not match the members in the Target cube.

Detail member-level mappings include:

- Simple Mappings: One to One mappings between source and target members
- Roll-up Mappings: Multiple members on a source to a single target member
- Multi-Dimension Mapping: Multiple dimensions mapping enables you to map two source dimensions to one target dimension or one source dimension to two target dimensions.
- Substitution Variable Mapping: Select a Substitution Variable so that it is referred and picked-up during the Data Push/Smart Push instead of explicitly mapping a dimension member in the source.

To perform a detailed member-level mapping:

1. From the **Data Maps** home page, click the name of the data map to which to define a detailed mapping.
2. From the **Detail** page, select the dimension to which to add the detailed mappings.
3. Click **Map** for the dimension to add the mappings.
4. On the **Edit Data Map** page, select the source member(s) to map to the target member.

   If the source members are different than the target members, the source members may be shown. If they are not shown, use the Populate Detailed Mapping feature, or select each member to map by clicking the arrow drop-down in Source and selecting each member.

Copyright © 2017, 2026, Oracle and/or its affiliates.

a. To add a source member, double-click the Source selection arrow ( ), and select the source member to map.

b. To add a target member, double-click the Target selection arrow ( ), and select the target member to map.

   If you are using a different (cross) application, under **Target**, and then from the **Cube** drop-down, select **Select Remote Cube**. For more information, see Using a Cross Application Smart Push or .

Additional options include:

- To add a mapping definition (row) between a source and target members, click .
- To delete a mapping row, select the row and click .
- To copy a mapping row, select the row to be copied and click .
- To paste a mapping row, click .
- To delete invalid entries, click .
- To populate detailed mappings, click .

  For information, see (see Populating Detail Mappings.

5. **Optional**: Specify an operator if needed.

   The operator for a row defines the operation to use for the current row.

   If you want to subtract some of the members, then change the operator to a -.

   By default, the + operator for each row defines the aggregation.

6. Click **Done**.

Defining a Simple Member-Level Mappings

Simple member-level mappings enables you to perform one to one mappings between source and target members.

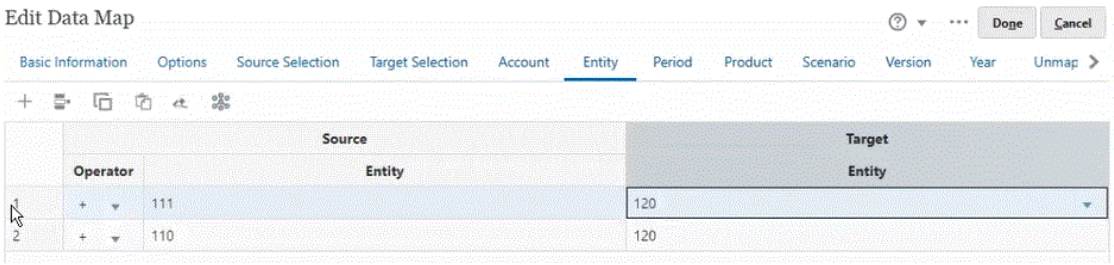
For information on defining a simple member-level mappings, see: Using Detailed Member-Level Mapping.

Defining a Roll-up Mapping

When you need to map multiple source members to a single target member, use roll up mappings to aggregate the selected source members to the target member. For example, to have multiple accounts from the source aggregate and map to a single account in the target, add a row for each member in the source, and map each to the same target member.

In the following example, source members "110" and "111" are rolled-up mappings to the target member "120."

Copyright © 2017, 2026, Oracle and/or its affiliates.



Defining a Multi-Dimension Mapping

Multi-Dimensions mapping enable you to map two source dimensions to one target dimension or one source dimension to two target dimensions. For example, map the source dimension "Year" to a combination of "Year" and "Period" dimensions in the target. Alternately, for the source a combination of "Year" and "Period" dimensions can be mapped to a single target dimension "Year."

To use multi-dimension mappings:

1. From the **Data Maps** home page, click the name of the data map to which to define a multi-dimension mapping.
2. From the **Detail** page, select the dimension to which to define a multi-dimension mapping and then select one of the following:

   To add a target mapping, click ••• to the right of the dimension and select **Add Target Mapping**. You can add one source dimension to two unmapped target dimensions.

   To add a source mapping, click ••• to the right of the dimension and select **Add Source Mapping**. You can add two unmapped source dimensions to one target dimension.
3. Click **Select** from the dimension where you either chose **Add Source Mapping** or **Add Target Mapping**, and then select the dimension.

   If you selected **Add Source Mapping**, select the member(s) for this dimension.

   If you selected **Add Target Mapping**, leave the member(s) as the default or select the target members to use in the mapping.

   If you leave the member as the default, then you can select any member from the dimension.

   If a member(s) has been selected, only the selected member(s) is available to use in the mapping.

4. Click **Map** in the dimension to map the members.
5. On the **Edit Data Map** page, complete one of the following:

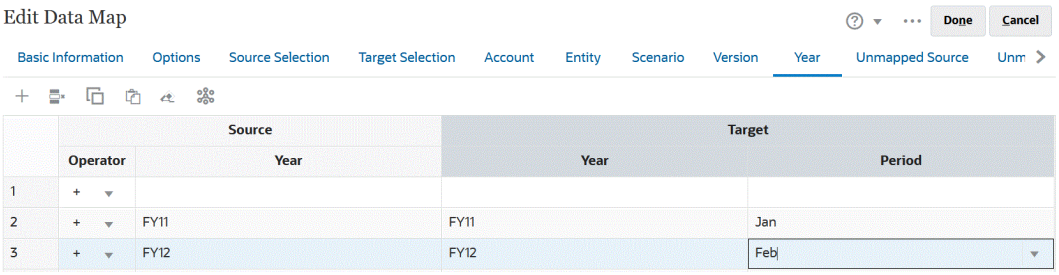
   If you are mapping two source dimensions to one target dimension, select the two source members and map them to the target member.

   If you are mapping one source member to two target members, select the source member and map it to the two target members.
6. **Optional**: Select an operator if required. The default operator is +.

Copyright © 2017, 2026, Oracle and/or its affiliates.

The operator for a row defines the operation to use for the current row.

In the following example the two target dimension have been mapped to the one source dimension.



7. Click **Done**.

Defining a Substitution Variable Mapping

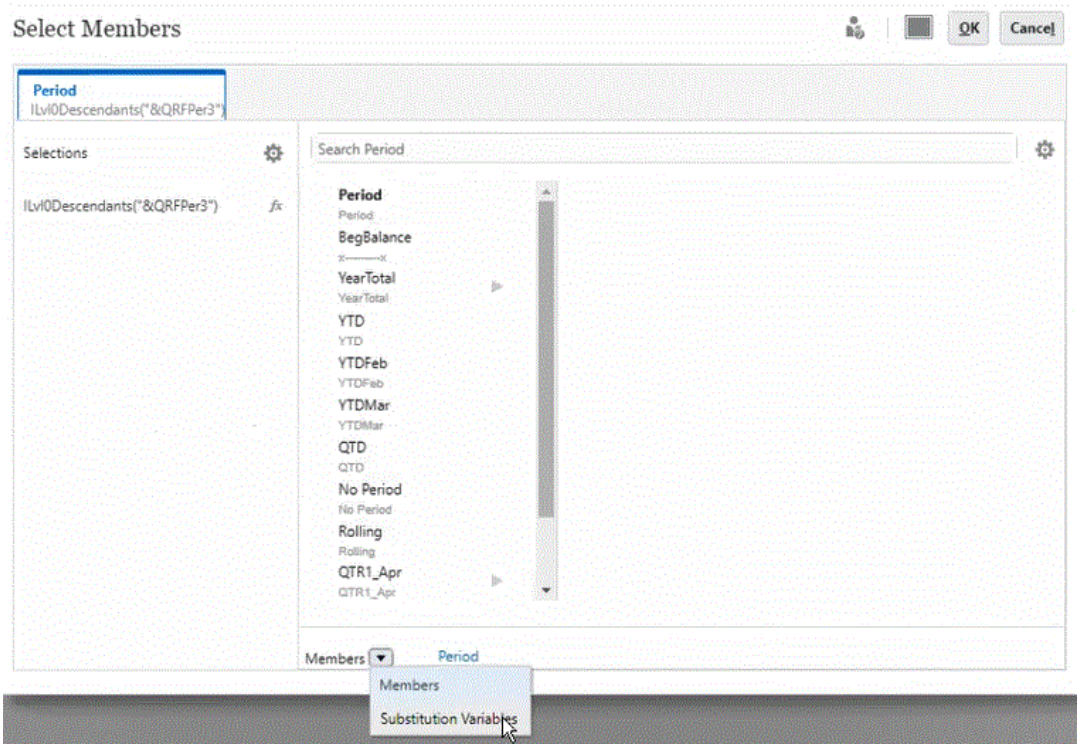
Use a Substitution Variable Mapping to pick up substitution variable during the Data Push/ Smart Push instead of using explicitly mapped dimensions in the source.

Note the following when using substitution variables mappings:

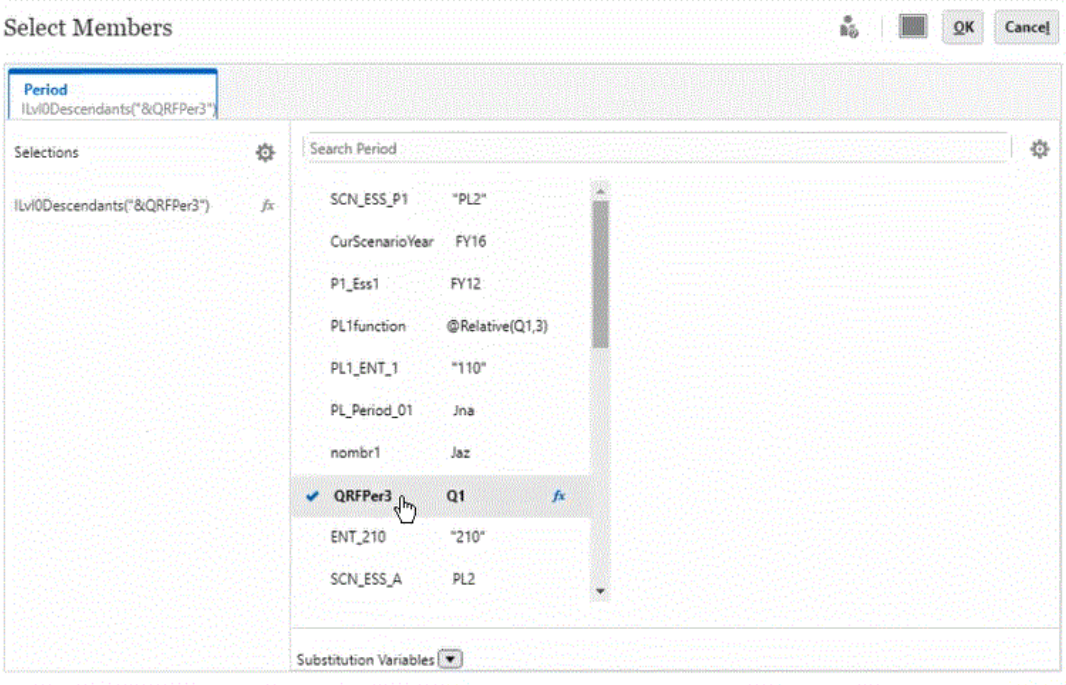
- Substitution variables are checked when you move data.
- The member selector displays the substitution variables that were defined for the Planning application.
- For dimension-to-dimension mappings, substitution variables reside in the source application and the value of the substitution variable are used to push to the target application.
- For unmapped dimensions, the member selector displays the substitution variables defined only for this application.
- Substitution variables are not supported for the target on cross application data maps.

1. From the **Data Maps** home page, click the name of the data map for which to define a substitution variable mapping.
2. From the **Detail** page, click the dimension with substitution variables.
3. From the **Select Members** page, and then from the **Members** drop-down, select **Substitution Variables**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

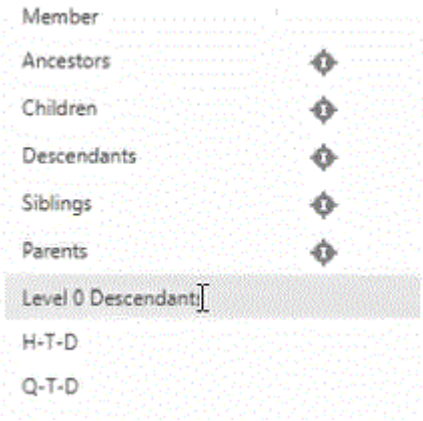


4. Select the substitution variable to map.



5. **Optional**: To use a function for a substitution variable, click (Function Selector icon) and select the function from the list.

Copyright © 2017, 2026, Oracle and/or its affiliates.



6. On the **Details** page, click **Map** in the dimension to map the substitution variable in the source to the target.
7. On the **Edit Data Map** page, map the substitution variables in the source to the members in the target.

> **Note**
>
> When a substitution variable is used in a data map, and if detailed mapping must be performed for this substitution variable, it is recommended that you specify all possible values for the substitution variables in the mappings to avoid changes to the data map after the value of the substitution variable is updated. This is only applicable if detailed mapping must be defined for that dimension.

8. Click **Done**.

Moving Unmapped Dimensions

To move unmapped dimensions in a data map:

1. View the **Data Maps** listing page.

   See Viewing Data Maps.
2. Do one of the following:
   - Click **Create** to create a new data map.
   - Click an existing data map to edit the mapping in an existing data map.
3. In the **Target** for **Unmapped Dimensions**, click ⬆ next to the dimension.

   This moves the dimension out of **Unmapped Dimensions** and under the **Target** dimensions above.
4. Find the dimension that you just moved under **Target**.

   The corresponding dimension under Source will say **Select**.
5. Click ▼ next to **Select** under **Source**, and then select the Source Dimension or Smart List that you want to map to the unmapped Target dimension.
6. Click **Save and Close**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

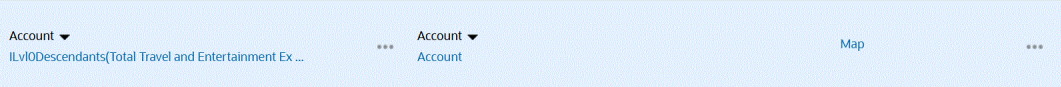
Populating Detail Mappings

Shows how to populate all members for all dimensions in the source that do not exist in the target.

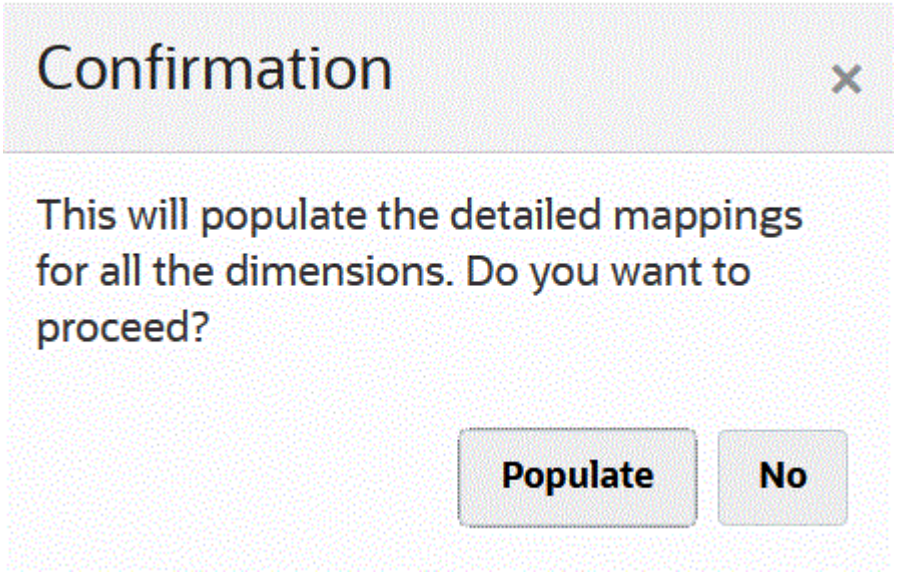
Use the Populate Detailed Mappings feature to populate all members for all dimensions in the source that do not exist in the target. You can also use this feature to populate a detail.

To populate detail mappings for all dimensions in the source that do not exist in the target:

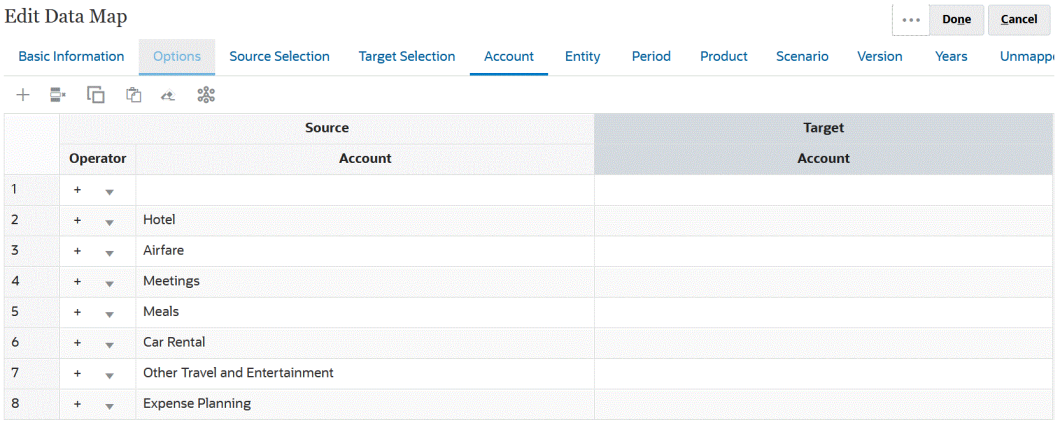
1. From the **Data Maps** home page, click the name of the data map.
2. From the **Detail** page, select the dimension to which to populate detailed mappings and then click **Map**.



3. From the **Edit Mapping** page, click , and then **Populate Detailed Mappings**.
4. On the **Confirmation** page, click **Populate**.



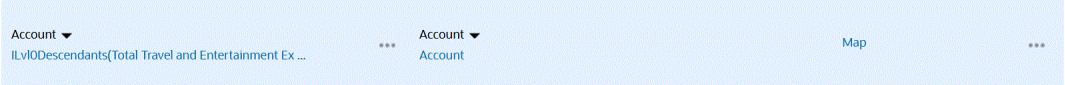
All dimensions are populated for the Source as shown below:



Copyright © 2017, 2026, Oracle and/or its affiliates.

To populate a detail mapping for a single dimension:

1. From the **Data Maps** home page, select the name of the data map to which to edit for the detailed mappings.
2. From the **Detail** page, click the selected members to edit and then click **Map**.



3. From the **Select Members** page, select the member(s) to use for the data map and click **OK**.
4. If you have existing mappings and you want to keep these, on the Warning page, select **No**.

   If you want to remove all mappings for this dimension, click **Yes**.
5. From the **Detail** page, click **Mapped** for the dimension to edit the mapping.
6. From the **Edit Mapping** page, click and then click **Done**.

Excluding Source Members from a Data Map

Shows how to exclude source members of a dimension while moving data from one cube to another cube.

Defining exclusions in data maps enables you to exclude source members of a dimension while moving data from one cube to another cube. Excluding source members while moving data is useful if you don't want to overwrite certain data at the target location or if you want to avoid getting warnings if data at the target location is not present. Also, exclusions can ignore specific source member data while data is required for all other source members. You can define exclusions for source mapped dimensions and unmapped dimensions and you can define multiple exclusions for a dimension.

Excluded source members for each dimension are logged in the Job Details for the Data Map or Smart Push operation.

Example:

In the Account dimension you want to bring in data for all accounts except for the Cash/Bank Account. This can be achieved by including All Accounts in the Source mapping and creating an exclusion for the Cash/Bank Account. Smart Push will honor the defined exclusion and ignore Cash/Bank Account, and then will load the data for all other accounts.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> - Strategic Modeling data map exclusions are also honored.
> - Exclusion of a base member also excludes any shared members. The same is true if you exclude a shared member; the base member is also excluded.
> - The excluded members must be a subset of the source members.
> - When excluding source members from a dimension, the member list for the dimension must contain at least one member that is not on the excluded member list for the data map to successfully save and run. For example, if the only Account member defined in a mapping is Cash/Bank Account, you can't exclude Cash/Bank Account from the mapping. You must correct the selection so that map results include at least one member. The same is also true for Smart Push definitions.

To exclude members from a data map:

1. View the **Data Maps** listing page.

   See Viewing Data Maps.
2. Click an existing data map.
3. Under **Source**, hover your cursor over the dimension row which contains the member you want to exclude, click ⋯, and then select **Add Exclusion**.

   > **Note**
   >
   > You can also exclude members in unmapped dimensions.
4. Select the member or members you want to exclude, and then click **OK**.
5. If you change your mind and want to remove an exclusion, hover your cursor over the dimension row, click ⋯, and then select **Remove Exclusion**.

   > **Note**
   >
   > **Remove Exclusion** removes the entire exclusion; not just one member.
   > To remove a member from the excluded list, launch the member selector, under **Selections**, highlight the member you want to remove, click
   >
   >
   >
   > and then click **Remove**.
6. Click **Save and Close**.

> **Note**
>
> At any time, excluded members can be appended or removed by opening the member selector of the exclusion list for the dimension.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Tip**
>
> If a member is excluded in one of the segments of the form, the Smart Push process will exclude that member from all form segments. If this behavior is undesired, use the **Overwrite Selection** option in Smart Push to refine the dimension member selection for the Smart Push process. See Configuring Smart Push for a Form.

Defining Data Map Options

Shows how to define data map settings and options.

To define data map options:

1. View the **Data Maps** listing page.

   See Viewing Data Maps.
2. Click an existing data map.
3. Click **Options**.

Table 10-1 Data Map Options

Data Map Option	Description
Select Items to Copy	Select the items for which you want to copy the corresponding relational data. You can copy: • **Comments and Attachments** If you select **Collate**, then: – If there is one-to-one mapping between the source cells and the target cells, then Comments and Attachments are copied from the source cells to the target cells. – If there isn't a one-to-one mapping between the source cells and the target cells, then the Comments and Attachments from the source cells are combined and saved in the corresponding target cells. If you do *not* select **Collate**, then: – If there is one-to-one mapping between the source cells and the target cells, then Comments and Attachments are copied from the source cells to the target cells. – If there isn't a one-to-one mapping between the source cells and the target cells, then last source cell's Comments and Attachments are copied into the corresponding target cell. "Last source cell" means the last source cell among multiple source cells that has non-empty relational data. • **Supporting Detail** – If there is a one-to-one mapping between the source cells and the target cells, then the Supporting details are copied from the source cells to the target cells. – If there isn't a one-to-one mapping between the source cells and target cells, then the Supporting details are not copied to the target cells. Note that the relational data (**Comments and Attachments** and **Supporting Detail**) selected to be copied in the **Data Map Options** dialog box applies only to moving data using Smart Push. See Moving Data.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 10-1 (Cont.) Data Map Options

Data Map Option	Description
Smart List/Pivot Dimension	Select a Smart List dimension or pivot dimension. Note that only dense dimensions are allowed for Smart List dimension selection. Pivoting a dimension changes the orientation of data on the grid so you can view data from another angle, or slice and dice information. Enterprise Profitability and Cost Management can only select the dimension shown in the Smart List/Pivot Dimension drop-down list. For more information about pivoting dimensions in Enterprise Profitability and Cost Management, see Pivoting Dimensions in *Administering and Working with Enterprise Profitability and Cost Management*. The default value for the Smart List/Pivot dimension is **Period**. If you are creating a new data map, if the Smart List to dimension mapping is required, the Smart List/Pivot Dimension option may need to be modified to the dimension using the Smart List.
Exclude Dynamic Calc Members	Select if you don't want to move the dynamically calculated member data during the data map process. Note that **Exclude Dynamic Calc Members** is supported for data push only and not for Smart Push. By default the **Exclude Dynamic Calc Members** is enabled.
Allow Smart Push Simultaneously	Allows data map and Smart Push processes to run at the same time. Click **Yes** to enable this option. By default this option is set to **No**. See Running Data Maps and Smart Push Processes Simultaneously.
Prioritize Smart List Entry Names When Moving Data	Requires the data map or Smart Push operation to use the Smart List entry name instead of the member alias in cases where the target cube has duplicate aliases. Smart List entry names cannot have spaces or special characters. If the Smart List is defined using a function and if the members have spaces, those spaces are converted to underscore. Due to this restriction, the Smart List entry name might not be used. When this option is selected, the system checks for the member based on the entry name, and if there are no members matching the entry name, the Smart List entry label is used to fetch the member. For intersections where the Smart List value isn't defined, the Smart Push and data map operations will use the missing label defined for the Smart List to push the data. If this option is not enabled and the data push or Smart Push operation detects duplicate aliases in the target cube, the operation will complete with warnings and the job details will prompt you to either remove the duplicate aliases (duplicated aliases are provided in the message) or select this data map option. The Prioritize Smart List Entry Names When Moving Data option is enabled by default.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 10-1 (Cont.) Data Map Options

Data Map Option	Description
Import #missing value as	Import options for missing values (only for data maps where Planning is the source and Strategic Modeling is the target) • **Ignore**—Ignores the imported #missing value and retains the existing value in the target cell. • **Zero As Output** (default)—Imports the #missing values and sets their output value to 0 in the target cells. • **NaN**—Imports the #missing values and sets their output value to NaN (Not a Number) in the target cells. • **Zero As Input**—Sets the input cell to 0, no back calculation required.
Import NaN value as	Import options for a NaN value (only for data maps where Strategic Modeling is the source and Planning is the target): • **Ignore**—Ignores the imported NaN value and retains the existing value in the target cell. • **Zero As Output** (default)—Imports the NaN value and sets the output value to 0 in the target cells. • **NaN**—Imports the NaN values and sets their output value to NaN in the target cells. • **Zero As Input**—Sets the input cell to 0, no back calculation required.
For cells that are not being imported	Import options for cells not imported: (Strategic Modeling only) • **Preserve Input Values**(default)—Preserves the existing input data values in the model during the data import • **Preserve Output Values**—Preserves the existing output data values in the model during the data import.
Suppress Missing Columns for Smart Push	Suppress missing columns from the source grid while performing a Smart Push. Disable this option to include missing columns in the Smart Push.

Editing Details for a Data Map

When you edit data map details, you can change data mappings and map unmapped dimensions.

To edit data map details:

1. View the **Data Maps** listing page.

   See Viewing Data Maps.

2. Do one of the following:
   - Click a data map.
   - Click ••• to the right of the data map that you want to edit, and then select **Edit**:
     - To edit the mapping between dimensions, see Changing Mapping Information for a Dimension.
     - To map an unmapped dimension, see Moving Unmapped Dimensions.
     - To define data map options, see Defining Data Map Options.
     - To delete a data map, to the right of the Source and Target dimension, click ✕.
3. Click **Save and Close**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Duplicating a Data Map

To duplicate a data map:

1. View the **Data Maps** listing page.

   See Viewing Data Maps.

2. Click ••• to the right of the data map that you want to duplicate, and then select **Duplicate**.
3. In the **Save As** dialog box, enter a name for the data map, and then click **OK**.

   The duplicate data map is added to the data map list. It has all the details of the original data map, but is saved under the new name.

Exporting a Data Map to Excel

Shows how to export a data map to a dynamic workbook in Excel.

Data Maps can be exported to a dynamic workbook in Excel enabling you to view the detail of the data map definition in an Excel format. The exported data map includes summarized basic information, data option and the detail for each mapped source and target dimensions.

To export a data map to Excel:

1. Click **Application,** then **Data Exchange**, and then click **Data Maps**.
2. From Data Maps home page, select the data map to export to Excel and from the **Actions** drop-down, select **Edit**.
3. From the details page, then from **Actions** menu, select **Export Data Map**.
4. When prompted, open or save the <*datamap_name*>.*xlsx*.

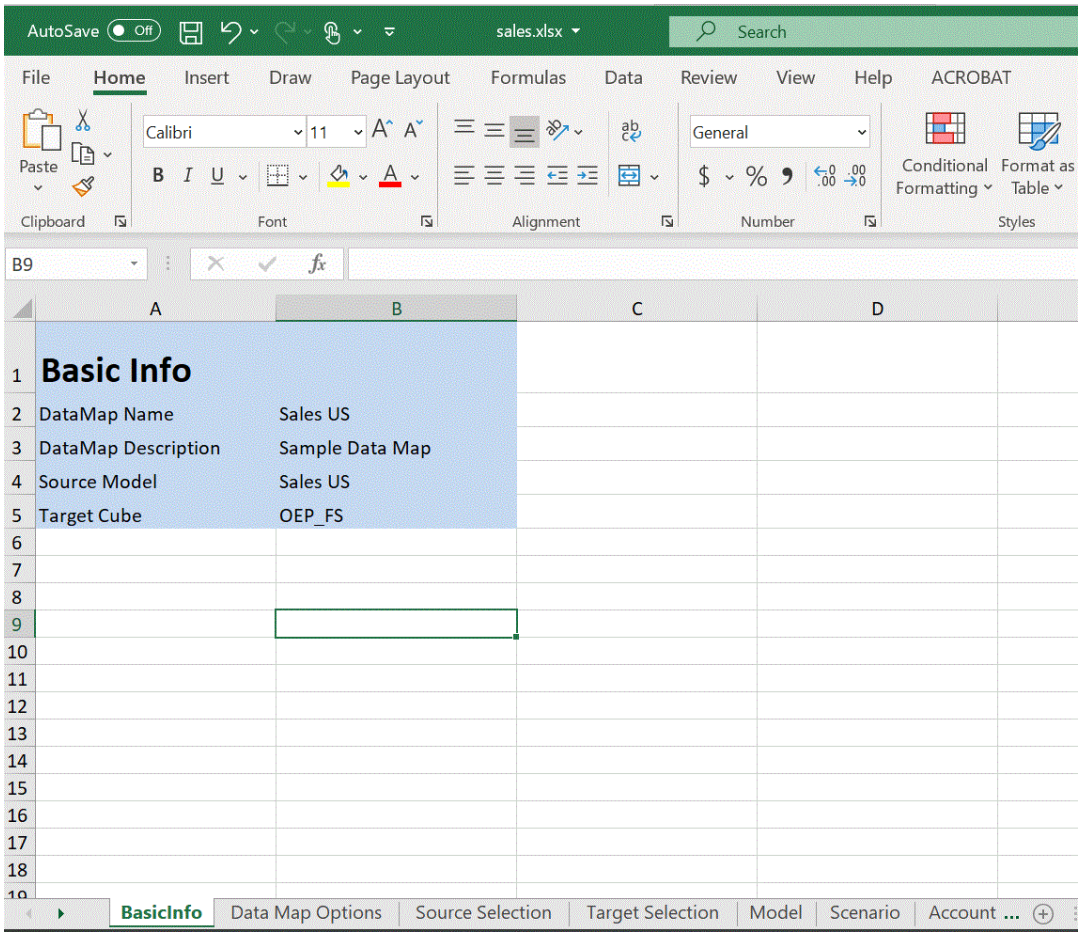
> ⓘ **Note**
>
> Updates can be made in Excel in the respective Tabs and saved. The updated Excel can be imported for updating an existing Data Map or for creating a new Data Map (by specifying a new data map name),

5. Update the Source and Target tabs if needed, and the detailed mappings in each tab of the Excel worksheet, and then save it.

   Updates can be made in Excel in the respective tabs and then saved. For example, customers can take advantage of Excel's copying a group of cells by dragging and dropping them to populate maps as needed. The updated Excel can be imported for updating an existing Data Map or for creating a new Data Map (by specifying a new data map name).

   The following is an example of a data map exported to an Excel workbook.

Copyright © 2017, 2026, Oracle and/or its affiliates.



Importing a Data Map

Shows how to import an external data map in Excel format to a data map.

You can import an external data map in Excel format to Data Maps. When the data map has been imported, you can view and update basic map information, options, source and target selections, and mapped and unmapped dimensions.

Before using this feature, it is recommended that the source and target cubes are selected along with the required smart list dimension for the data map from the user interface before exporting the data map.

During the import of the data map, the options are derived from the data map through which the import operation is performed, so it is recommended to import the data map through the same data map through which the export file was generated. If a new data map must be created through the import, select the source and target cubes along with the smart list dimension in the user interface before importing the Excel data map.

> **Note**
>
> Before importing a data map, make sure it is in an Excel file format (XLSX). The system does not allow you to import a data map in CSV file format.

Considerations for importing a data map:

Copyright © 2017, 2026, Oracle and/or its affiliates.

When importing a data map, keep in mind the following considerations:

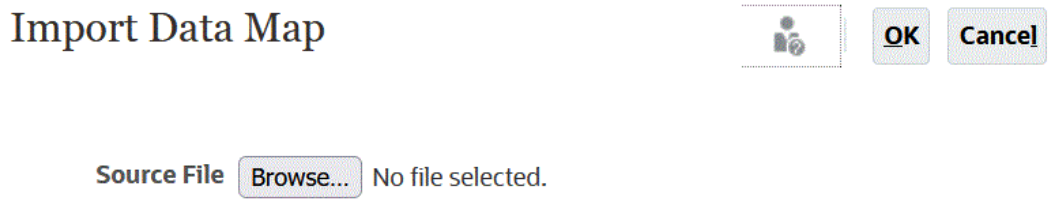
1. A Planning data map to a Planning data map differs from an import from Planning data map to Strategic Modeling data map, and a Strategic Modeling data map to Strategic Modeling data map.

   Options for the import of a new Planning data map to a Planning data map are derived from the data map through which the import is being performed. When the options are not selected in the initial data map, then they are derived from the default settings.

2. A Planning data map to a Planning data map may fail when a member is in source and not in target. For a Strategic Modeling data map, the members are ignored when they exist in the source but not the target.

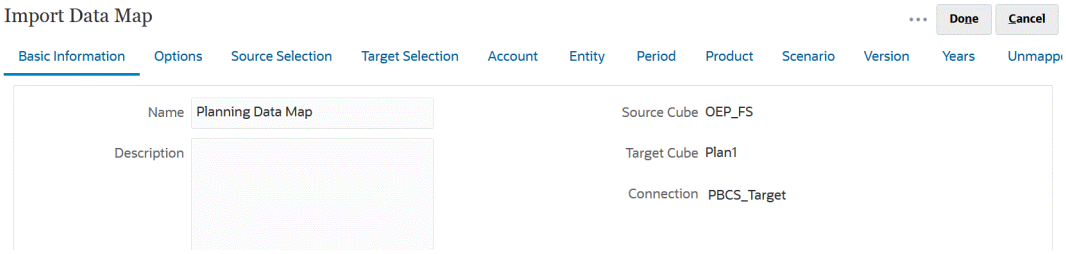
To import a data map:

1. Click **Application,** then **Data Exchange**, and then click **Data Maps**.
2. From the Data Maps home page, click **Create**.
3. From the details page, select the source and target cubes.
4. Select **Actions**, and then **Options**.
5. On the **Data Map Options** page, select the smart list dimensions and then click **OK**.
6. From the details page, then from **Actions** menu, select **Import Data Map**.
7. From the **Import Data Map** selection page, click **Browse** to navigate to the data map source file and then click **OK**.



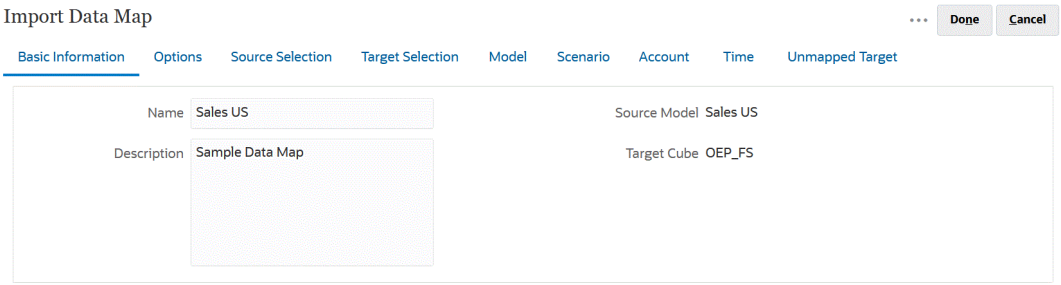
8. When the imported data map is displayed, click **Done**.

   The following example shows an imported Planning data map:



   The following example shows an imported Strategic Modeling data map:

Copyright © 2017, 2026, Oracle and/or its affiliates.



Deleting a Data Map

Shows how to delete a data map.

To delete a data map:

1. View the **Data Maps** listing page.

   See Viewing Data Maps.

2. Click ••• to the right of the data map that you want to remove, and then select **Delete**.

Validation Rules for Selected Members

Explains validation rules for selected members.

Selected members in the reporting cube store the data that is moved from the source Planning application. If any of the following constraints are not met, or if a dimension in either application isn't mapped and has no valid selected member, then a mapping isn't valid and an error message is displayed.

Rules:

- If the reporting cube is an aggregate storage database, then the default members must be level 0 members.
- If the reporting cube is a block storage database, then the default members can be any members with the Store Data property.
- If the Planning application has only dimension to dimension mappings, then the default members can be any level or data storage type.
- If the Planning application has Smart List to dimension mappings, then default members must be only level 0. In addition, the source cube must contain a dense Account dimension with at least one member associated with a Smart List.
- If Descendants (Acct_Default) is selected in a mapping, the Acct_Default member must exist in the reporting cube.

> **Note**
>
> Mappings that were once valid can become invalid if dimensions, members, or Smart Lists are renamed, removed, or added. If a target cube has a change in dimensionality, you must select the corresponding application mapping on the **Map Reporting Application** screen and click to refresh the data.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Moving Data

After setting up data maps, you can:

- move data to a reporting cube
- push data with member mappings or a data map to a remote environment
- move data from one cube to another cube using Smart Push
- run data maps and Smart Push processes simultaneously

Moving Data to a Reporting Cube

Shows how move data to a reporting cube.

After setting up data maps, you can move data to a reporting cube. Planning validates the selected data maps, and then moves the mapped Planning dimension data to the reporting cube dimensions. You can also check the Job for the job status.

> **Note**
>
> If you enabled parent members for dynamic children in both a block storage and aggregate storage cube and added a dynamic member, you can use **Push Data** to successfully move data from the block storage to the aggregate storage cube without refreshing the database. See Synchronizing Smart Lists in Reporting Applications in *Administering Planning*.

To move data to a reporting cube:

1. Create the reporting cube.
2. Create the data map.
3. Click **Application**, then **Data Exchange**, and then click **Data Maps**.
4. To the right of the data map, click •••, and then select **Push Data**.

   Confirm whether you want to clear the data before moving it:

   - Click **No** to add the new data to the existing data in the reporting cube. No data is cleared from the target cube when you use this option.

     If you map Smart Lists to dimensions, Smart List labels must match either member names or aliases in the reporting cube. **Push Data** doesn't work with Smart List entry names.
   - Click **Yes** to clear the data in the target cube before moving data to it.

     When the target is an aggregate storage reporting cube, note the following when clearing and moving data:

     – Members with non-matching names in the target reporting cube are ignored.

     – This option works only with member names, not with member aliases.

     – Use caution when using member relationships (such as Children) when selecting members for the data map, because using this option can cause the calculation script to exceed its length limit.

Copyright © 2017, 2026, Oracle and/or its affiliates.

- If you use member relationships, this option expands the level 0 member list in the source Planning cube. If at least one member name in the source cube matches a member in the reporting cube, this option proceeds without error. If at least one member doesn't match, the option can't proceed.
- When an on-fly member/dynamic member is part of a selection in the data map, you need to synchronize the data map before executing the data push.

When the target is a block storage reporting cube, to proceed successfully, clearing and moving data requires these conditions:

- If you use member relationships, all member names in the source cube must match all member names in the reporting cube.
- If you map Smart Lists to dimensions, all Smart List entries in the source cube must match all member names in the reporting cube.
- If you map Smart Lists to dimensions, the Smart List entry label in the source Planning cube must match the member name in the reporting cube. If a Smart List entry label doesn't match a member name in the reporting cube, then the Smart List entry name must match the reporting cube member name.

If the previous conditions are not met, the clearing and moving data operation can't proceed.

Moving Data Using a Cross Application Data Map

Shows how to transfer data, cell comments, and supporting from a source cube to a remote reporting cube.

For data maps with member(s) mapping and data can be pushed to a remote target environment.

To transfer data, cell comments, and supporting from a source cube to a remote reporting cube:

1. Create a new Oracle Fusion Cloud Enterprise Performance Management connection if it's not available to the cross application where the data has to be moved.

   For more information, see Creating, Editing, and Deleting Connections to Other Cloud EPM Environments in *Administering Planning*.
2. Create the reporting application.

   For information on cubes, see Managing Cubes in *Administering Planning*. For information on adding members, see Editing Dimension Properties in the Simplified Dimension Editor in *Administering Planning*.
3. Click **Application** then **Data Exchange**, and then click **Data Maps**.
4. Click **Create**.
5. Under **Source**, select a cube from the available cubes for the current application.
6. Under **Target** and from the **Cube** drop-down, choose **Select Remote Cube**.
7. From the **Select Remote Cube** page, select the connection and cube for the target application and then click **OK**.
   - Under **Connection**, select the connection to use to connect to the target cube.
   - Under **Cube**, select the cube from the remote application.

     All the available connections in the source application are displayed. The cubes from the remote application are displayed based on the selected connection.

Copyright © 2017, 2026, Oracle and/or its affiliates.

8. Click **OK**.

   The system displays the message: "Loading dimensions in the target cube."

9. Map any source members with their corresponding target members when member names differ in the source and target applications. For more information, see Changing Mapping Information for a Dimension.

10. On the **Detail** page, click **Save and Close**.

11. On the Data Maps listing page, select the data map job to push to the cross application and then from the **Actions** menu, select **Push Data**.

12. When prompted with the message: **Do you want to clear the data before push?**, select either **Yes** or **No** to clear the target cube.

> **Note**
>
> In the case of one to multi-dimension mapping, the data map should be configured with the target selection to clear data. Otherwise the data push fails.

The Cross Application Data Maps job is displayed in the Jobs console. If an error occurs, click the error link to troubleshoot the problem. If the data is moved, the **Run Status** displays either as **Complete** or **Completed with Warnings**. Click the job name to review the log for further details.

> **Note**
>
> You can run Cross Application Data Maps job as a scheduled job.

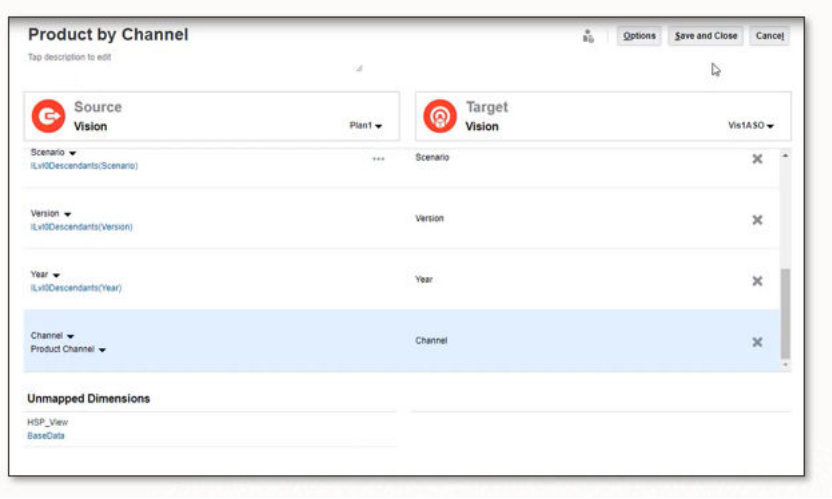
Moving Data from One Cube to Another Cube Using Smart Push

Shows how to move data from one cube to another cube using Smart Push.

About Smart Push

For more meaningful and complete reporting, users can instantly move comments, attachments, and supporting detail from source cubes to a reporting cube while working in forms. Users can then do more analysis on the planning data coming from the different cubes.

Copyright © 2017, 2026, Oracle and/or its affiliates.



For example, assume that you have expense planning in one cube and revenue planning in another cube. Assume further that you use a reporting cube to consolidate your reporting needs. Without Smart Push, the data from your cubes would be moved to the reporting cube through scheduled jobs set up by Service Administrators. Smart Push allows the data to be moved immediately to the reporting cube by users.

You can move application data from:

- Block Storage cubes to aggregate storage cubes
- Block Storage cubes to block storage cubes

Videos

Your Goal	Watch This Video
Learn how to instantly push data for reporting using Smart Push.	Instantly Push Data for Reporting in Planning

Considerations When Using Smart Push

When using Smart Push, keep in mind the following considerations:

- Smart Push honors metadata and approvals security.
- Smart Push requires that at least one of the dense dimensions, account or period, is set as a dimension-to-dimension mapping in the data map definition.
- Numeric data across all members selected for dimensions listed in the unmapped section of the source cube in the data map isn't listed in the Smart Push definition.
- A Data map has an exclusion when the Smart Push is defined:
  1. using a form context
  2. with no exclusion on the form definition

  In this case, it is recommended that you define the Smart Push as "Run after save" on the Web form since no exclusion is picked up by Smart Push.

  When Smart Push is defined through Groovy, then the exclusion is picked by Smart Push from the Data map.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Configuring Smart Push for a Form

Smart Push enables you to use data maps defined in the application to synchronize data automatically when you change data in forms. In this way, Smart Push enables you to access the functionality of a data map for real-time data movement.

When configuring smart push on a form having multiple segments, Oracle recommends you use an overwrite selection and provide the appropriate selection for the overwrite selections instead of using the form context. This limits the data transferred to only the necessary data to move, which promotes better performance.

If the form context option is selected, then the final selection is the cumulative selection from all segments of the selection as well as any excluded selections.

For example, if you choose three segments with selection such as:

```
Segment-1 --> ILvl0Descendants(Q1)


Segment-2 --> ILvl0Descendants(Q2) Exclude Selection: Feb


Segment-3 --> ILvl0Descendants(Q2) Exclude Selection: Mar
```

the final selection using form context is:

```
ILvl0Descendants(Q1) , ILvl0Descendants(Q2) Exclude Selection: Feb,Mar
```

If a segment is hidden in the form and the smart push is configured to use a form context, then the smart push also transfers data from the hidden segments to the target cube.

If the final selection is not a desired selection to move data, then configure the smart push to use the overwrite selection.

To configure smart push for a form:

1. From the Home page, click **Navigator** , and then under **Create and Manage**, click **Forms**.
2. Expand **Forms**, and then select a form.

   The form can also be edited once the run-time view of the form is launched from a Navigation Flow or through the Data Card.
3. With the form selected, select .
4. Click **Smart Push**.

   The defined mappings for the form are displayed. Click to add new mappings if desired.
5. Expand each mapping.
6. For each dimension shown under a mapping, specify the Smart Push region for the dimension:

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> The **Use Form Context** for dimensions on rows and columns uses the form design members to push, not the members on the form. For the POV and Page dimensions, **Use Form Context** uses the member shown on the form at run time.

- Select **Use Form Context** to move data for all the members that were selected (and excluded) for the dimension on **Layout**.

  The Page Dimensions use the member selected on the form.

- Clear **Use Form Context**, and then next to **Overwrite Selection**, click :

  – Select **Edit** to use the member selector to select the dimension members you want to work with.

  – Select **Add Exclusion** to use the member selector to define the dimension members you want to exclude from the Smart Push process. **Remove Exclusion** removes the entire exclusion; not just one member.

  > **Note**
  >
  > The following exclusion considerations apply:
  >
  > * Smart Push exclusions honor any exclusions defined in forms if the **Use Form Context** option is selected for Smart Push.
  > * When excluding members from a dimension, the member list for the dimension must contain at least one member that is not on the excluded member list for the Smart Push process to successfully complete. For example, if the only Account member defined for Smart Push is Cash/Bank Account, you can't exclude Cash/Bank Account from the Smart Push process. You must correct the selection so that Smart Push results include at least one member.

  To remove a member from the excluded list, click

  next to the excluded list of members. Remove the member(s) from the member selector by selecting the member in the right pane, and then clicking or .

  See Excluding Source Members from a Data Map.

  – Select **Clear** to remove selections.

> **Note**
>
> You can override a mapping definition on source cube dimensions only. You can override dimension members if you take the context from a form by selecting **Use Form Context**. You can also override dimension members if you specify members in the **Member Selection** dialog box, or if you leave **Overwrite Selection** blank.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Tip**
>
> Smart Push makes incremental updates to the reporting cube. To achieve this, you can select the **Use Form Context** or **Overwrite Selection** options to limit the amount of data being moved to the current context or to limit the data movement to the section that you are updating or calculating.

If the Smart Push region isn't specified using one of the above options, then member selection is taken from the data map.

7. For each mapping, define whether Smart Push will be automatic or manual:
   - Select **Run After Save** to automatically move the data when the form is saved.
   - Leave **Run After Save** cleared to manually move the data.

     For information on how to manually move the data, see Moving Data After Configuring Smart Push.
8. Select from these additional Smart Push options:
   - **Run Smart Push in Background**—Allows you to continue using the application while Smart Push runs in the background.
   - **Use Database Suppression**—Enables you to apply row suppression at the database level instead of at the application level, which eliminates the impact on query thresholds and improves Smart Push wait times.

     The **Use Database Suppression** option is recommended if you are using Smart Push to load large amounts of data from a block storage cube and while using the missing block and row suppression options.

     If you use Groovy rules to configure Smart Push with the **Use Database Suppression** option, there is an additional parameter to specify this option for Smart Push execution:

     ```
     public void execute(Map<String, String> overrideMembersMap, boolean
     suppressMissingRowsNative)
     ```
9. Click **Save** to save the Smart Push configuration for the form.

Using a Cross Application Smart Push

The Cross Application Smart Push feature enables customers with multiple environments to instantly transfer data, cell comments and supporting detail from a Planning cube in a source environment to a reporting or other Planning cube in a remote environment while working in forms.

> **Note**
>
> The Cross Application Smart Push feature is not available for Strategic Modeling.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> Transfer of attachments using Smart Push is supported only for same environment data maps.

To transfer data, cell comments, and supporting from a source cube to a remote reporting cube:

1. Create a new Oracle Fusion Cloud Enterprise Performance Management connection if it's not available to the cross application where the data has to be moved.

   For more information, see Creating, Editing, and Deleting Connections to Other Cloud EPM Environments in *Administering Planning*.

2. Create the reporting application.

   For information on cubes, see Managing Cubes in *Administering Planning*. For information on adding members, see Editing Dimension Properties in the Simplified Dimension Editor in *Administering Planning*.

3. Click **Application,** then **Data Exchange**, and then click **Data Maps**.
4. Click **Create**.
5. Under **Source**, select a cube from the available cubes for the current application.
6. Under **Target** and from the **Cube** drop-down, choose **Select Remote Cube**.
7. From the **Select Remote Cube** page, select the connection and cube for the target application and then click **OK**.
   - Under **Connection**, select the connection to use to connect to the target cube.
   - Under **Cube**, select the cube from the remote application.

     All the available connections in the source application are displayed. The cubes from the remote application are displayed based on the selected connection.
8. Click **OK**.
9. Map any source members with their corresponding target members when member names differ in the source and target applications. For more information, see Changing Mapping Information for a Dimension.
10. On the **Detail** page, click **Save and Close.**
11. Open a form and add the Smart Push to the form.

    For more information, see Configuring Smart Push for a Form.

Configuring Merge Options

Merge options for comments, attachments, and supporting detail are configured on the **Data Options** tab available when mapping a cube for reporting. See Defining Data Map Options.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> When using Smart Push, relational data such as Supporting Details, Date, Smart List, and Text can't be merged. Smart List, Date, and Text are moved only in cases of a one-to one-mapping between source cells and target cells. Empty cells are not considered for a relational data move.

Moving Data After Configuring Smart Push

See Moving Data for information on how to move data.

How the data is moved depends on whether you selected **Run After Save** when you configured Smart Push for a form.

- If you selected **Run After Save** when you configured Smart Push for a form, the data is moved when you save the form.
- If you left **Run After Save** unchecked:
  1. Click **Data**.
  2. Click a form to open it.
  3. With the form open, select **Actions**, and then click **Smart Push Details**.
  4. Click the link in the Smart Push dialog box to move the data.

> **Note**
>
> Smart Push considerations:
>
> - Smart Push always clears data in the target region before moving new data.
> - If users observe errors in Smart Push execution, the Service Administrator must verify and fix how Smart Push is set up by using the errors listed in the details of the Smart Push job. See Viewing the Smart Push Status.

Data Push and Smart Push Performance

If the data push or Smart Push is taking more time than expected to process, check if there are any dynamic calc members as part of the selection. If the data from these dynamic calc members is not required to be moved to the target, add the dynamic calc members to exclude the selection.

If the dynamic calc member data has to be pushed, check if the formula has cross dimensional members and are included in @ISMBR function These formulas has to be rewritten by avoiding the @ISMBR function to improve the performance.

Running Data Maps and Smart Push Processes Simultaneously

While moving data to a reporting cube, you can minimize the time the system is unavailable to users. This can be achieved by moving only the required data to the reporting cube using runtime prompts in a Groovy rules-based data push. For a data push through EPM Automate, you can use substitution variables and set up substitution variable values before moving data using Data Maps. See these topics for more information:

Copyright © 2017, 2026, Oracle and/or its affiliates.

- To find documentation and examples for Groovy business rules, see **Oracle Enterprise Performance Management Cloud, Groovy Rules Java API Reference**.
- For information about EPM Automate, see *Working with EPM Automate*.

You can also further minimize downtime by allowing users to run the Smart Push and Data Map processes simultaneously by enabling the **Allow Smart Push Simultaneously** option in **Data Map Options**.

Consider the following when running simultaneous Data Map and Smart Push executions:

- Smart Push will wait for data to clear in the reporting cube before pushing data.
- There are limits on the amount of data that can be processed simultaneously. Check the job output and design accordingly.

For information about data map options, see Defining Data Map Options.

Checking Real Time Progress Status for Data Maps and Smart Push

View comprehensive information about each job's operations, including the completion status and duration in Jobs.

When pushing data using data maps and smart push, you can view comprehensive information about each job's operations, including the completion status and duration in Jobs.

A sub-status field provides the detail of the specific operation being performed as part of the submitted job, such as clear data, export data, and so on. For parent-child job relationships, the parent job displays the current ongoing child job, while each child job features the specific operation being performed as part of its execution. The sub-status of a job is shown while the job is executing and it is cleared after a successful completion of the job. If a job encounters an error, the sub-status field remains as-is and it is not cleared.

Sub status for data maps and smart push enables customers to have access to meaningful details instead of just the processing state, without having to open the details of each individual job. Customers can easily identify when a job is blocked. For example, if an exclusive operation like "Clear Data" is being performed through some process on a target ASO cube and the customer tries to perform clear operation through a data map on the same cube, the sub- status indicates that the cube is waiting.

Cross application data maps create a job in both the source and target environments. The target environment has the job details for clear and import options. The job details from the target environment are copied to the job details on the source environment.

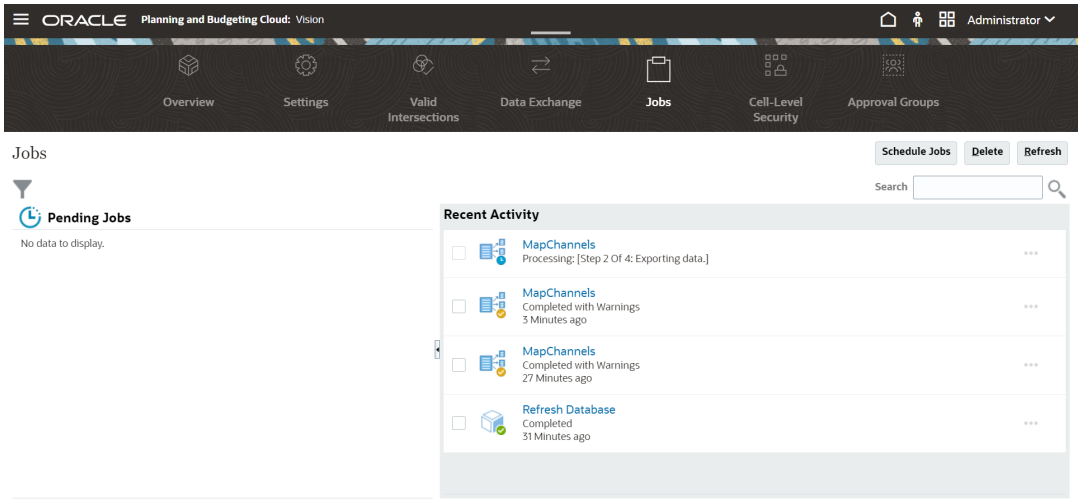
The data map job status is "completed with warning" when any governor of the data map crosses the warning threshold. This helps an administrator to identify any errors and take any corrective actions.

> **Note**
>
> For Enterprise Profitability and Cost Management customers, sub-status details are only available on the Jobs Detail page and not on the Jobs listing page.

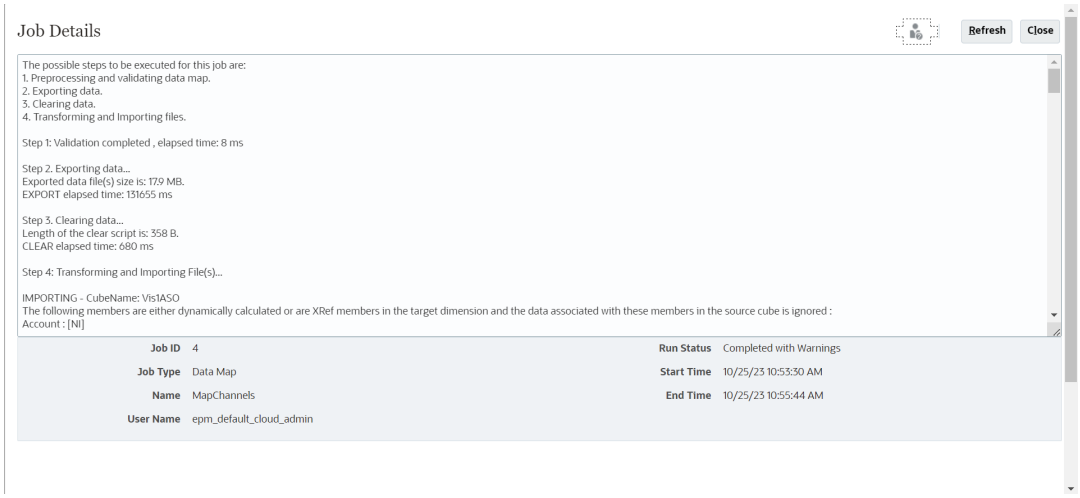
To check the real-time progress status for data maps and smart push

1. From the **Home** page, click **Application**, then click **Jobs**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



2. **Optional**: Perform the following task(s):
   - To filter jobs and activity by criteria such as date or job type, click .
   - To search for job by name, enter text in the **Search**, and then click .
3. **Optional**: To view the details of the job, click the job name.



The sub-status of a job is only updated when you click **Refresh**.

Viewing the Smart Push Status

During the execution of a smart push job, the following sub-statuses are shown for parent and child jobs:

For the parent job, the sub status is the parent job run status followed by the running count of the executing child jobs, for example: `Processing: [Executing Child Job 1 Of 5: Child_Job_Name] --> Parent job is in processing state`. If an error occurs, the system shows an error message for the child job where the error occurred, for example: `Error: [Executing Child Job 2 Of 3: Child_Job_Name] --> Parent job is in Error state as the mentioned child job failed.`

For a child job, the sub-status is the job run status of the job followed by the current step of execution, for example: `Processing: [Step 1 Of 5: Validating]` and `Processing: [Step 5 Of 5: Clearing and pushing data to target]`

Copyright © 2017, 2026, Oracle and/or its affiliates.

When the clear data is selected as NO, the above is shown, except the Clearing data step is not shown.

Identifying Smart Push Intersections with Unrecognized Smart List Values

When Smart Push fails with the message "Unrecognized Smart List Value," you can identify the intersections with the invalid Smart List values by creating a data form.

To identify intersections with the unrecognized Smart List values:

1. Create a data form with a definition the same as the Smart Push definition for all the dimensions except for the smart list/pivot dimension.
2. For the Smart List dimension only, add the Smart List members which are used in the data map to define a Smart List to dimension mapping.
3. Include the Smart List dimension as a column dimension in the data form.
4. All the dimensions which have only a single member as per the Smart List definition need to be part of the POV or page dimensions.

   Include all other dimensions in rows.
5. Save the form definition.
6. Open the data form and look for cells where the value is defined as a number instead of a Smart List value. These are the intersections with unrecognized Smart List values.

Viewing the Status of the Data Maps Job

During the execution of the Data Maps job, the following sub-statuses are introduced when the Clear Data option is set to "Yes":

1. Preprocessing and validating data map.
2. Exporting data.
3. Clearing data.
4. Transforming and Importing[Transforming file [File_Getting_Executed] Of [Total_Number_Of_Files]]/Transforming and Importing[Importing file [File_Getting_Executed] Of [Total_Number_Of_Files]](At step 4 one of the above messages is displayed.)

When the clear data is selected as NO, the steps above are shown, except the Clearing data step is not shown.

During the execution of the Data Maps job, the following sub-statuses may display on the Jobs page:

1. Preprocessing and validating data map.
2. Exporting data.
3. Clearing data.
4. Transforming and Importing files.

To see which statuses were processed on the Job Details page, click the executed Data Maps job on the Jobs page

Copyright © 2017, 2026, Oracle and/or its affiliates.

Viewing the Strategic Modeling Push Status

For more information about the Strategic Modeling push status, see the following in *Administering Planning*:

- Checking Real Time Progress Status for Data Maps and Smart Push in *Administering Planning*
- Viewing the Strategic Modeling Push Status in *Administering Planning*

Copyright © 2017, 2026, Oracle and/or its affiliates.

11

Running an Integration

Run an integration to extract the data from the source system, view and verify the results, and then push the data to the target application.

You run an integration to extract the data from the source system, and then view and verify the results. If the data has been transformed correctly, you can push it to the target system.

The Run Integration page consists of two tabs: Options and Filters.

The Options tab enables you to select general and target options for the selected integration at the transformation stage. Note that the period selections on this page depend on the target system.

The Filters tab enables you to quickly add or modify filters at runtime and execute them rather than modifying the integration definition every time. Filters are used to query the data from the data sources. The changes to the filters are not saved and used for that execution only.

Watch this tutorial for information about running integrations: Setting File-Based Data Integration Options and Running Integrations in Enterprise Planning Cloud.

To run an integration:

1. From the **Data Integration** home page, select an integration, and then click ▶.
2. Select the **Options** tab.
3. Select **Import Source** to import the data from the source system, perform the necessary transformation, such as import, map, and validate the data.

   Select this option only when:

   - you are running an integration for the first-time.
   - the data in the source system has changed. For example, if you reviewed the data in the workbench after the export, and it was necessary to modify data in the source system.

   In many cases, source system data may not change after you import the data from the source the first time. In this case, it is not necessary to keep importing the data if it has not changed.
4. Select **Recalculate** to skip importing the data, but re-process the data with updated mappings.
5. From **Import Mode**, select the method for importing data.

   Available import modes:

   - **Append**—Keep existing rows for the POV but append new rows to the POV. For example, a first-time load has 100 rows and second load has 50 rows. In this case, 50 rows are appended. After this load, the row total for the POV is 150.
   - **Replace**—Clears all data for the POV in the target, and then loads from the source or file. For example, a first-time load has 100 rows, and a second load has 70 rows. In this case, 100 rows are removed, and 70 rows are loaded to TDATASSEG. After this load, the row total is 70.

Copyright © 2017, 2026, Oracle and/or its affiliates.

For a Planning application, Replace clears data for Year, Period, Scenario, Version, and Entity that you are loading, and then loads the data from source or file. Note that when you have a year of data in the Planning application, but are only loading a single month, this option clears the entire year before performing the load.

> **Note**
>
> When running an integration in Replace mode to an ASO cube, if the scenario member is a shared member, then only a Numeric data load is performed. Be sure to specify the member name with fully qualified name including complete hierarchy. The All Data Type load methods does not work when the scenario member is a shared member.

> **Note**
>
> Replace Mode is not supported for the load method "All data types with auto-increment of line item."

- **Merge**—Merge adds existing cells and create new cells if they do not exist. It does not add to the data.

  Merge mode eliminates the need to load an entire data file when only a few balances have changed since the last time data was loaded. If mappings change between two loads, customers must reload the full data set.

  For example, a customer might have 100 rows of existing balances for one number account IDs, each which has an amount of $100.00. If the customer runs the integration in merge mode and the source has one row for one account ID with an amount of $80, then after running integration, there are one hundred rows of balances, 99 each of which have a balance of $100.00, and 1 which has a balance of $80.00.
- **No Import**—Skip the import of data entirely.
- **Map and Validate**—Skip importing the data but reprocess the data with updated mappings.

6. From **Start Period**, select the first period for which data is to be loaded.

   You can filter periods by simply typing the character(s) to filter by. For example, type **J** to filter by months beginning J such as June or July. You can also click the drop down and specify additional filter criteria in the edit box shown below **More results available, please filter further**.

   This period name must be defined in period mapping.

   > **Note**
   >
   > Use a Start Period and End Period that belong to a single Fiscal Year. If a period range crosses fiscal years, you run into the following issues:
   >
   > - When loading data in replace mode, the system clear data for both years
   > - When exporting data, you will get duplicate data
   >
   > The above is applicable for all modes of data load including standard and quick mode.

Copyright © 2017, 2026, Oracle and/or its affiliates.

7. From **End Period**, select the last period for which data is to be loaded.

   This period name must be defined in period mapping.

8. From **Period**, select the period for which to load data.

   If a POV period has been selected on the Home page, then the selected period defaults here. If no POV period is selected, then the system defaults to the last used period based on browser cache.

   You can select another period when an unlock icon ( ) appears next to it.

   When the Global POV mode is enabled in System Settings in Data Integration, then the period defaults to the Global POV Period and a customer is not allowed to select another period. In this case, a lock icon ( ) appears next to the Period drop-down.

9. Select **Export to target** to export the data to the target application.

10. Select **Execute Check** to generate the data, and then run the Check Report.

11. From **Export Mode**, select the method for exporting data to the target application.

    Available options:

    - **Merge**—Overwrite existing date with the new data from the load file. (By default, all data load is processed in the Merge mode.) If data does not exist, create new data.
    - **Replace**—Clears all data for the POV in the target, and then loads from the source or file. For example, a first-time load has 100 rows, and a second load has 70 rows. In this case, 100 rows are removed, and 70 rows are loaded to the staging table. After this load, the row total is 70.

      For a Planning application, Replace clears data for Year, Period, Scenario, Version, and Entity that you are loading, and then loads the data from source or file. Note that when you have a year of data in the Planning application, but are only loading a single month, this option clears the entire year before performing the load.
    - **Accumulate**—Accumulate the data in the application with the data in the load file. For each unique point of view in the data file, the value from the load file is added to the value in the application.
    - **Subtract**—Subtract the value in the source or file from the value in the target application. For example, when you have 300 in the target, and 100 in the source, then the result is 200.
    - **Dry Run**—(Financial Consolidation and Close and Tax Reporting only) Scan a data load file for invalid records without loading data it to the target application. The system validates the data load file and lists any invalid records into a log, which lists 100 errors or less. For each error, the log indicates each record in error with its corresponding error message. Log details are available in Process Details.

      > **Note**
      >
      > Dry Run ignores the Enable Data Security for Admin Users target option and always uses the REST API for the administrator user.

    - **No Export**—Skip the export of data entirely.
    - **Check**—After exporting data to the target system, display the Check report for the current POV. If check report data does not exist for the current POV, a blank page is displayed.

Copyright © 2017, 2026, Oracle and/or its affiliates.

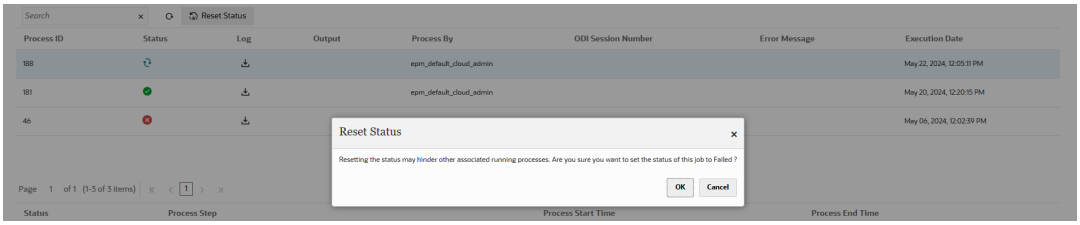
12. Click **Filters**
13. Add or modify the filter(s) to execute at runtime.

    Filters are used to query the data from the data sources. Filter specified on the Run Integration page are not saved and used for that execution only.

    For more information, see Defining Filters.
14. **File-Based Integrations Only**: If no file profile has been selected for the integration, in **File**, specify the file to use at run time.

    If a file has been selected when the integration was created on the General page, the selected file profile is shown in this field and you can't upload a different file at runtime with another file profile.
15. Click **Run**.

    If you need stop an incorrect execution of an integration job without restarting the instance, you can reset the status of a integration to a "Failed" status. To do this, go to **Process Details** for the job and click **Reset Status**. When prompted, click **OK** on the **Reset Status** page.



For more information about Process Details, see Viewing Process Details.

Viewing the Integration Job

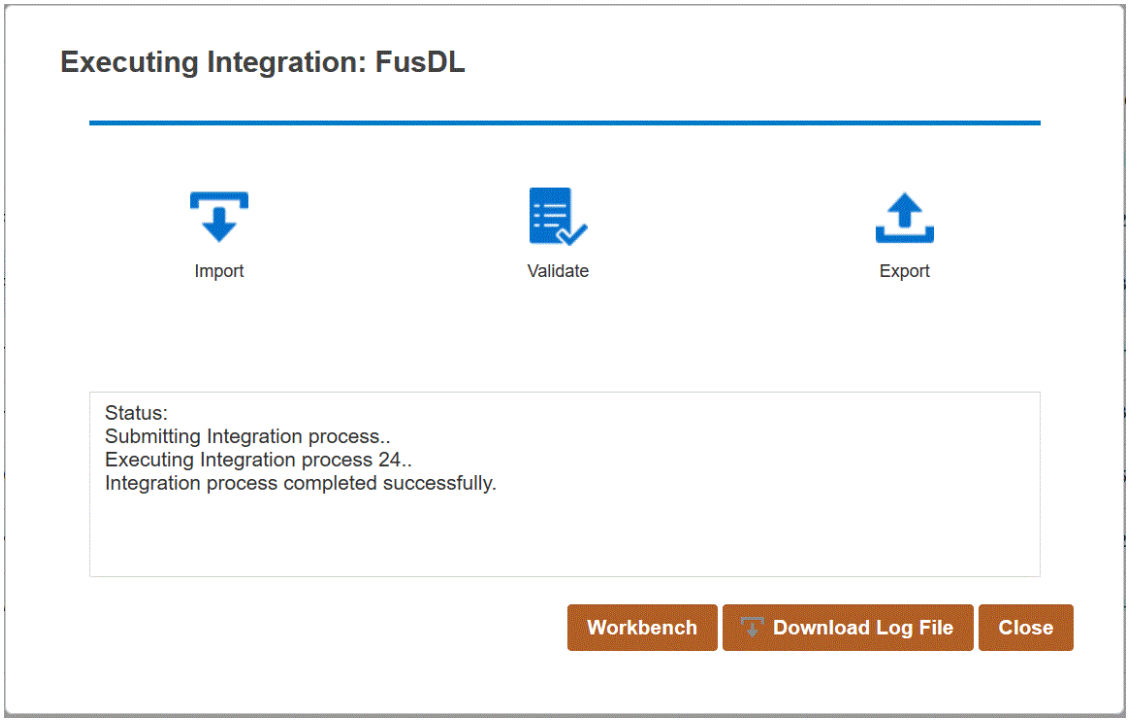
Shows how to launch visual feedback on the status of each submitted process.

When an integration job is submitted on the Run Integration page, the Run Integration Execution pop-up shows the process flow of the four main integration operations: Import, Validate, Export, and Check Data. These icons provide visual feedback on the status of each submitted process. The pop-up also shows the status of the submitted processes.

You can select to execute an integration offline, in which case the system processes the data in the background instead of processing the integration immediately.

When an integration has been run successfully, you can select to go to the Workbench or download the log file.

Copyright © 2017, 2026, Oracle and/or its affiliates.



Copyright © 2017, 2026, Oracle and/or its affiliates.

12

Using the Pipeline

Runs an Oracle Fusion Cloud EPM pipeline (orchestration of a series of jobs as a single process) based on the variables defined for the pipeline.

The Pipeline feature enables you orchestrate a series of jobs as a single process. In addition, you can orchestrate Oracle Fusion Cloud Enterprise Performance Management jobs across instances from one location. Using the Pipeline, you have better control and visibility of the full extended data integration process for preprocessing, data loading and post processing jobs.

Job types supported in the Pipeline include:

- Runs a Data Integration pipeline (orchestration of a series of jobs as a single process) based on the variables defined for the pipeline.

  Runs a Data Integration pipeline (orchestration of a series of jobs as a single process) based on the variables defined for the pipeline.

  Using a Business Rule Job Type
- Using a Business Ruleset Job Type
- Using a Clear Cube Job Type
- Using a Copy from Object Storage Job Type
- Using a Copy from SFTP Job Type
- Using a Copy to Object Storage Job Type
- Using a Copy to SFTP Job Type
- Using a Create Reconciliations Job Type
- Using an EPM Platform Job Job Type for Planning Jobs
- Using an EPM Platform Job Job Type for Enterprise Profitability and Cost Management
- Using an EPM Platform Job Job Type for Financial Consolidation and Close and Tax Reporting Jobs
- Using an Export Dimension by Name (EDMCS) Job Type
- Using an Export Dimension Mapping by Name (EDMCS) Job Type
- Using an Export Mapping Job Type
- Using an Export Metadata Job Type
- Using an Extract Dimension Viewpoint (EDMCS) Job Type
- Using an Extract Package (EDMCS) Job Type
- Using a File Operations Job Type
- Using a Generate Report for Account Reconciliation Job Type
- Using an Import Attribute Values Job Type
- Using an Import Balances Job Type
- Using an Import Mapping Job Type

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Using an Import Metadata Job Type
- Using an Import Pre-Mapped Balances Job Type
- Using an Import Pre-Mapped Transactions (TM) Job Type
- Using Import Rates Job Type
- Using an Integration Job Type
- Using an Integration with Smart Split Job Type
- Using an Open Batch - File Job Type
- Using an Open Batch - Location Job Type
- Using an Open Batch - Name Job Type
- Using a Plan Type Map Job Type
- Using a Run Auto Alert Job Type
- Using a Run Auto Match Job Type
- Using a Set Period Status Job Type
- Using a Set Substitution Variable Job Type

Pipeline Considerations

Note the following Pipeline considerations:

- Only administrators can create and run a Pipeline definition.
- A non-administrator can be set as a proxy user to run a Pipeline definition.
- Pipeline is a replacement for the batch functionality in Data Management and can be migrated automatically to the Pipeline feature in Data Integration.
- For file-based integrations to a remote server in the Pipeline when a file name is specified in the pipeline job parameters, the system copies any files automatically from the local host to the remote server automatically under the same directory.
- **Note**

  The character limit for a sub-directory string name is 2,000 characters.

Pipeline Tasks

This section describes tasks associated with the Pipeline.

Copying a Pipeline

The Copy Pipeline feature enables you to create a Pipeline identical to another Pipeline but with a different name within the same environment or to a remote environment. .

Use the copy Pipeline feature to create a new copy of a Pipeline definition with identical connections and job type information. You might use this feature when you want to copy a Pipeline definition in a test environment or you want to copy a source Pipeline as a model for another Pipeline.

To copy a Pipeline:

Copyright © 2017, 2026, Oracle and/or its affiliates.

1. From the Data Integration home page, click ••• next to the Pipeline to copy and then from the **Actions** drop-down, select **Copy**.
2. From the **Copy Pipeline** page, then in **New Pipeline Name**, specify the new Pipeline name.

   The new name of the copied Pipeline is a concatenation of the source Pipeline name and the suffix:" -1" by default. Optionally, you can assign a unique name to the copied Pipeline.
3. From the **New Pipeline Code**, specify the new Pipeline code.

   The new Pipeline code is a concatenation of the existing Pipeline code and the suffix "1." Optionally, you can assign a unique Pipeline code, which can only include alphanumeric characters.

   The code can contain up to 25 alphanumeric characters with a minimum of 3 characters and a maximum of 25 characters. This code cannot be updated after a Pipeline is created.
4. From **Connections**, select the connection to use when copying the integration.

   Select **Local** to copy the integration in the same instance. You can also select the connection of another instance to copy the integration to the other instance. The connections must be defined in Oracle Fusion Cloud EPM Connections. For more information, see About Connecting Cloud EPM Environments in *Administering Planning* and Creating, Editing, and Deleting Connections to Other Cloud EPM Environments in *Administering Planning*.

   Select a remote connection to copy an Pipeline to a remote environment.

   If you specify a remote connection, all jobs and connections used in your Pipeline must already exist in the remote environment. The system validates connections and integrations only. If connections and integrations don’t exist, then the copied Pipeline fails. Other job parameters are not validated. If any referenced job parameters do not exist, the user interface shows a blank value. For example, when a Business Rule does not exist, the system does not validate the referenced job parameters and the user interface shows a blank value.
5. Click **Save**.

Pipeline Process Description

Shows the workflow for creating and running a Pipeline.

The workflow for creating and running a Pipeline process is as follows:

1. From the Data Integration home page, click ••• before the **Actions** menu, and then click **Create Pipeline**.
2. **Optional**: To enable non-administrators to view Pipelines on the Data Integration home page, complete the following:
   a. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Configure** actions drop-down, select **System Settings**.
   b. From the **System Settings** page, then in the **Enable Pipeline Execution for Non-Admin** setting to enable non-administrator users to view Pipelines on the Data Integration home page, select **Yes**.
   c. Click **Save**.
3. From the **Create Pipeline** page, then **Details**, in the **Pipeline Name**, specify a name for the Pipeline.
4. In **Pipeline Code**, specify a Pipeline code.

Copyright © 2017, 2026, Oracle and/or its affiliates.

The code can contain up to 30 alphanumeric characters with a minimum of 3 characters and a maximum of 30 characters. This code cannot be updated after a Pipeline is created. Use the pipeline code to execute the Pipeline using a REST API.

5. If jobs are run in parallel mode, specify the maximum number of jobs to run in parallel mode in **Maximum Parallel Jobs** for each stage.

   When jobs are run in parallel mode, at runtime, the system runs jobs together in parallel (not sequentially). You can enter between 1 to 25 jobs to run in parallel mode.

   When jobs are run in serial mode, at runtime, the system runs the jobs one after another in a specific sequence.

6. In **Location**, specify or select the location to associate with the Pipeline.

   > **Note**
   >
   > The name of the location selected for the Pipeline cannot be the same name used in an integration.

7. **(Optional)** : To enable non-administrators to run a Pipeline job, complete the following:

   a. In **Proxy User**, enter the username associated with the service administrator role.

      If the user name is different from the current username, then the system prompts the user to enter the password.

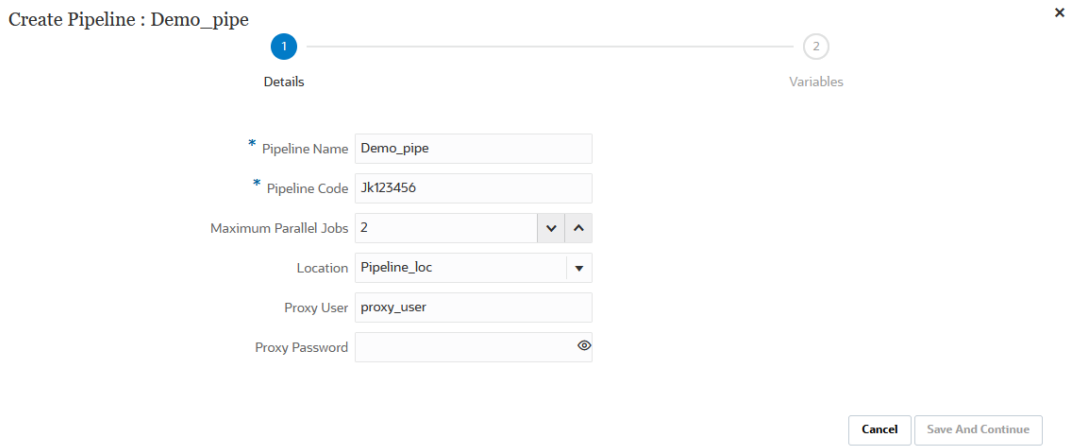
      For information on how roles are defined in the Oracle Fusion Cloud Enterprise Performance Management, see Access Control in *Administering Access Control* .

      If the **Proxy User** name is the same name as the service administrator, then the system does not prompt for a password.

   b. In **Proxy Password**, enter the password for the proxy user.

   For more information about using proxy administrator users, see Allowing Non-Administrators to Execute Jobs in the Pipeline.

8. Click **Save and Continue**.



The new Pipeline is added to the Data Integration homepage. Each Pipeline is identified with a under the **Type** header.

Copyright © 2017, 2026, Oracle and/or its affiliates.

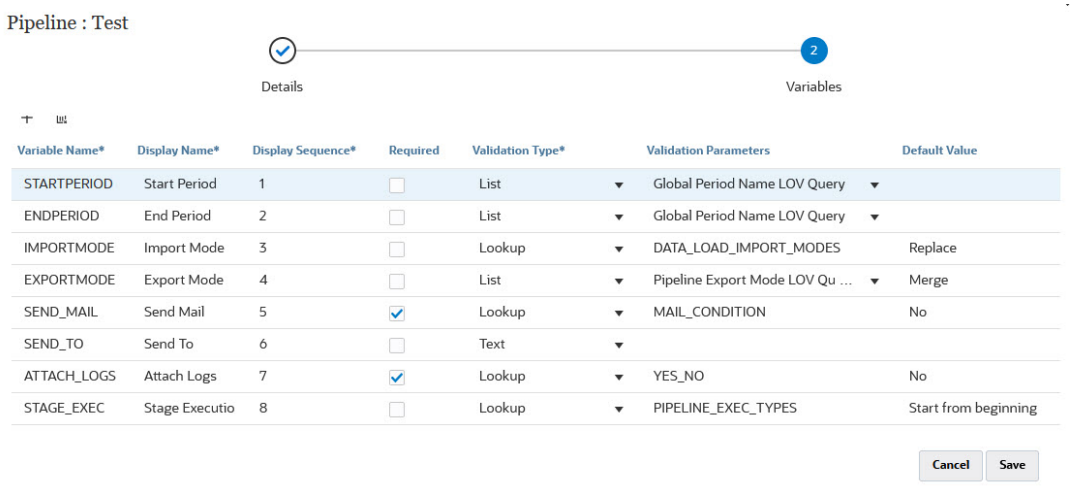
> **Note**
>
> When location security is enabled, non-administrators can view Pipeline jobs in ready-only mode (cannot edit or create a Pipeline) and run individual jobs in the Pipeline based only on the users groups by location to which they have been assigned in Location Security. (The service administrator can create, edit, and run any job type in the Pipeline.)

You can search for Pipeline jobs by searching for the word "pipeline" or a part of the word from Search.

You can view or edit an existing Pipeline by clicking ••• to the right of the Pipeline and then selecting **Pipeline Details**.

9. On the **Variables** page, a set of out-of-box variables (global values) for the Pipeline are available from which you can set parameters at runtime as shown in the following screenshot. Variables can be pre-defined types like: "Period," "Import Mode," "Export Mode," and so on, or they can be custom values used as job parameters.

   For example, you can set a substitution variable (a user variable name preceded by one or two ampersands (&)) for the Start Period.



   For more information on runtime variables, see Editing Runtime Variables.

10. Click **Save**.

11. On the **Pipeline** page, click .

    A new stage card is created.

A stage is a container for jobs that you want to run in the Pipeline and can include jobs of any type and for multiple target applications. Use a stage card as an entry point to add,

Copyright © 2017, 2026, Oracle and/or its affiliates.

manage, and delete job(s) contained in each stage. Each stage can execute jobs in serial or parallel modes. Stages are only executed serially meaning if you have multiple stages, each stage must complete before the system executes the jobs in the next stage.

A sample Pipeline might include the following stages:

Stage 1: Load Metadata (jobs are run in serial mode)

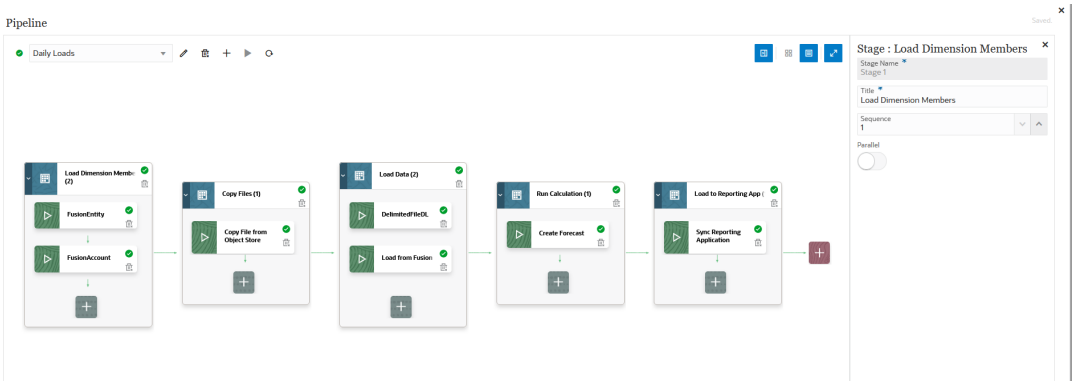
- Load Account Dimension.
- Load Entity Dimension.

Stage 2: (jobs are run in parallel mode)

- Load Data from Source 1.
- Load Data from Source 2.

Stage 3: (jobs are run in serial mode)

- Run Business Rule to perform calculation.
- Run substitution variables.



12. In the **Stage Editor**, specify the stage definition:
    - **Stage Name**—Specify the name of the stage.
    - **Title**—Specify the name of the stage to appear on the stage card.
    - **Sequence**—Specify a number to define the chronological order in which a stage is executed.
    - **Parallel**—Toggle **Parallel** on to run jobs simultaneously.

      The number of parallel jobs executed is determined by the maximum parallel jobs specified in the **Maximum Parallel Jobs** field from the **Create Pipeline** page.

> **Note**
>
> The pipeline automatically moves to the next job when the current job is still running after thirty minutes. In this case, the current job and next job run concurrently.
> If you have set a maximum time in the Batch Timeout in Minutes option (see System Settings), then the pipeline uses the maximum time value to determine how long a job runs. This setting is based on the sync mode (immediate processing) selection. In sync mode, Data Integration waits for the job to complete before returning control.

Copyright © 2017, 2026, Oracle and/or its affiliates.

- **Note**

  The **On Success** and **On Failure** options below control the processing of subsequent stages in the Pipeline. That is, when a stage succeeds or fails, do you want the Pipeline process to stop or to continue or skip to another stage. All jobs within a stage are always executed irrespective of success or failure of other jobs. So, if there is only one stage, then this option is not relevant.
  For this reason, customers should organize logical job types in a stage. For example, you might include all metadata loads in one stage and if it fails, then the data isn't loaded. Or if a data load stage fails, don't perform calculation jobs in a stage.

  On Success—Select how to process a stage when steps in the Pipeline definition are executed successfully.

  Drop-down options include:

  – Continue—Continue processing a stage when steps in the stage of the Pipeline definition execute successfully.

  – Stop—Skip a stage in the case where you want to bypass a stage that is only executed on failure and proceed to the following stage in the Pipeline definition.

  On Failure—Specify how to process a stage when a step within a stage fails.

  Drop-down options include:

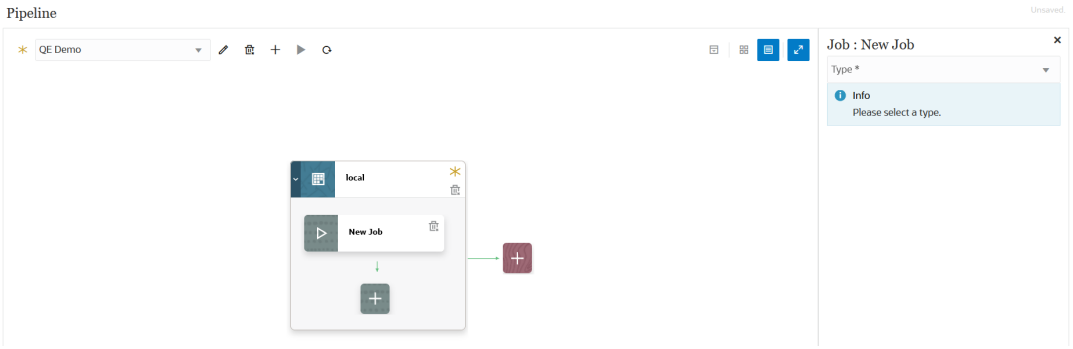
  – Continue—Continue processing subsequent steps in the stage of a Pipeline definition when a step within the stage fails.

  – Stop—Stop processing the current stage of a Pipeline definition when a step fails to process, and skip to perform any cleanup steps.

13. On the stage card, click **>** to add a new job to the stage.

14. On the stage card, click (Create Job icon).

    A new job card is displayed in the stage card.



15. In the **Job Editor**, then from **Type** drop-down, select the type of job to add to the stage card.

    Job types include:

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Business Rule
- Business Ruleset
- Clear Cube
- Copy from Object Storage
- Copy from SFTP
- Copy to Object Storage
- Copy to SFTP
- Create Reconciliation (ARCS only)
- EPM Platform Job for Planning
- EPM Platform Job for Financial Consolidation and Close and Tax Reporting
- EPM Platform Job for Enterprise Profitability and Cost Management
- Export Dimension by Name (EDMCS)
- Export Dimension Mapping by Name (EDMCS)
- Export Mapping
- Export Metadata
- File Operations
- Generate Report for Account Reconciliation (ARCS only)
- Import Attribute Values (ARCS only)
- Import Balances (ARCS only)
- Import Mapping
- Import Metadata
- Import Pre-Mapped Balances (ARCS only)
- Import Pre-Mapped Transactions (TM) (ARCS only)
- Import Rates (ARCS only)
- Integration
- Integration with Smart Split
- Open Batch - File
- Open Batch - Location
- Open Batch - Name
- Plan Type Map
- Run Auto Alert (ARCS only)
- Run Auto Match (ARCS only)
- Set Period Status (ARCS only)
- Set Substitution Variable

16. From the **Connection** drop-down, select the connection name to associate with the job type.

    The connection can be either a "local" connection (the connection is on the host server) or "remote" (the connection is on another server). By default, "Local" is the value for a

Copyright © 2017, 2026, Oracle and/or its affiliates.

connection. If a job type supports a remote operation, (for example, an integration to move data to a remote business process), then you are prompted for the connection name.

Note the following exceptions:

- Copy to Object Storage—The Object Storage requires an *Other Web Services Provider* connection type. You must have access to the Web service to which you're connecting. You must also have the URL for the Web service and any login details, if required. For more information, see Connecting to External Web Services in *Administering Planning*.

  In addition, you must generate an *auth token* to use as the user password for an *Other Web Services Provider* connection type. For information on creating an auth token, see To create an auth token.

- Copy from Object Storage—The Object Storage requires an *Other Web Services Provider* connection type. You must ensure you have access to the Web service you're connecting. You must also have the URL for the Web service and any login details, if required. For more information, see Connecting to External Web Services in *Administering Planning*.

  In addition, you need to generate an *auth token* to use as the user password for an *Other Web Services Provider* connection type. For information on creating an auth token, see To create an auth token.

17. From **Name**, select the name of the job.

    The **Name** job parameter is not applicable for:

    - Set Substitution Variable job type
    - Copy to and from Object Storage job types
    - Open Batch (by file, location, and name) job types

18. In **Title**, specify the title of job name to appear on the job card.
19. In **Sequence**, select the order in which to run the job when jobs are in the stage.
20. From **Job Parameters**, select any job parameters associated with the job.

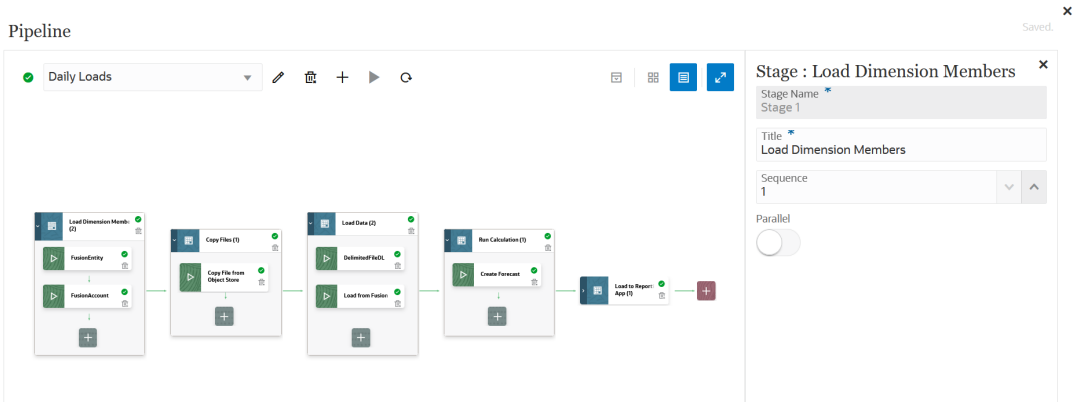
    Job parameter can be a static value assigned to the job or can be assigned from within a Pipeline variable.

    Job parameters are based on the job type. See below for the parameters associated by job type:

    - Using a Business Rule Job Type
    - Using a Business Ruleset Job Type
    - Using a Clear Cube Job Type
    - Using a Copy from Object Storage Job Type
    - Using a Copy from SFTP Job Type
    - Using a Copy to Object Storage Job Type
    - Using a Copy to SFTP Job Type
    - Using a Create Reconciliations Job Type
    - Using an EPM Platform Job Job Type for Planning Jobs
    - Using an EPM Platform Job Job Type for Enterprise Profitability and Cost Management

Copyright © 2017, 2026, Oracle and/or its affiliates.

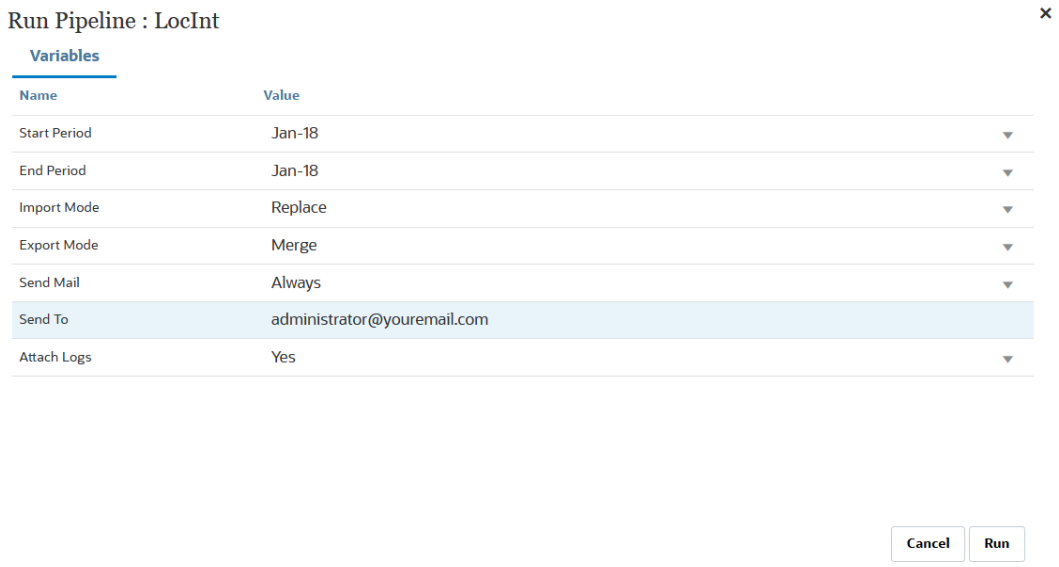
- Using an EPM Platform Job Job Type for Financial Consolidation and Close and Tax Reporting Jobs
- Using an Export Dimension by Name (EDMCS) Job Type
- Using an Export Dimension Mapping by Name (EDMCS) Job Type
- Using an Export Mapping Job Type
- Using an Export Metadata Job Type
- Using an Extract Dimension Viewpoint (EDMCS) Job Type
- Using an Extract Package (EDMCS) Job Type
- Using a File Operations Job Type
- Using a Generate Report for Account Reconciliation Job Type
- Using an Import Attribute Values Job Type
- Using an Import Balances Job Type
- Using an Import Mapping Job Type
- Using an Import Metadata Job Type
- Using an Import Pre-Mapped Balances Job Type
- Using an Import Pre-Mapped Transactions (TM) Job Type
- Using Import Rates Job Type
- Using an Integration Job Type
- Using an Integration with Smart Split Job Type
- Using an Open Batch - File Job Type
- Using an Open Batch - Location Job Type
- Using an Open Batch - Name Job Type
- Using a Plan Type Map Job Type
- Using a Run Auto Alert Job Type
- Using a Run Auto Match Job Type
- Using a Set Period Status Job Type
- Using a Set Substitution Variable Job Type



21. Click to run the Pipeline.

Copyright © 2017, 2026, Oracle and/or its affiliates.

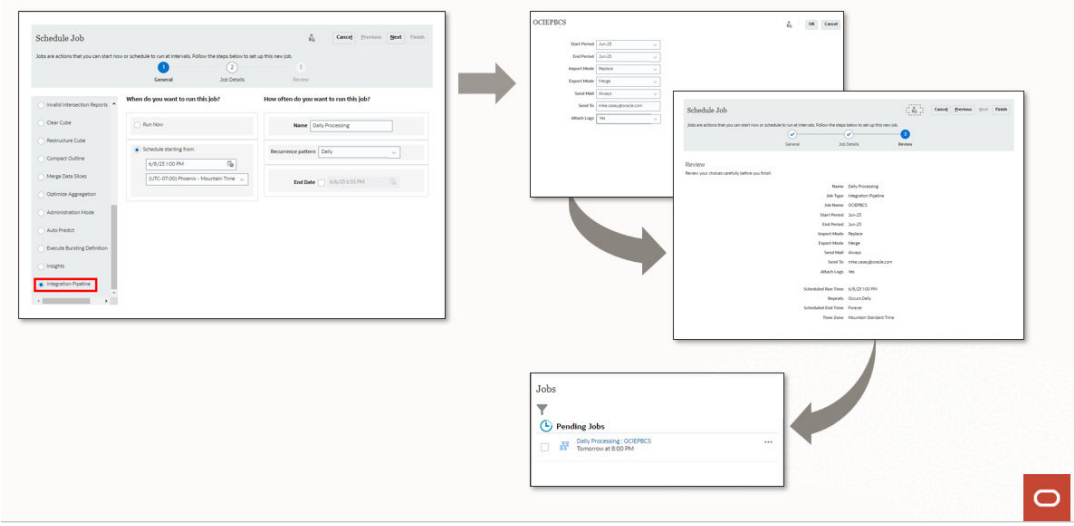
22. On the **Run Pipeline** page, complete any runtime prompts and then click **Run**.



When the Pipeline is running, the system shows the status as: .

You can click the status icon to download the log. Customers can also see the status of the Pipeline in Process Details. Each individual job in the Pipeline is submitted separately and creates a separate job log in Process Details (for more information, seeViewing Process Details).

23. Optionally, you can schedule a Pipeline definition to run based on the parameters and variables that were defined for the Pipeline in the Data Integration user interface. For more information, see: Scheduling Jobs in *Administering Planning*.



24. Click Save.

If there are any unsaved changes in the Pipeline, the message: **Unsaved** is shown beside the Save When the Pipeline is saved, the unsaved message is changed to **Saved**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> To cancel or undo any changes made in the Pipeline without saving changes, click **Cancel** or (Reset icon).

Pipeline Command Reference

The table below shows the command icons used for the Pipeline.

> **Note**
>
> For information on Pipeline accessibility keys, see Data Integration in *Accessibility Guide* .

Table 12-1 Pipeline Command Reference

Command Icon	Description
	Edit Pipeline
	Delete Pipeline
	In Grid View and Container View, click to create a new Pipeline.
	In Table View, use the Add drop-down to create a new: • stage • job • Pipeline
	Run Pipeline
	Refresh Pipeline
	Expand/Collapse All Stages in Container View and Table View. In Grid View, the Expand/Collapse All Stages option is not available.
	Grid View
	Container View
	Table View
	Toggle Editor

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 12-1 (Cont.) Pipeline Command Reference

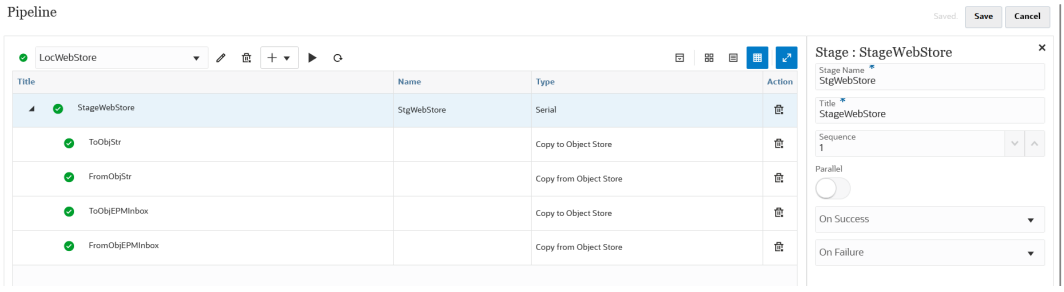
Command Icon	Description
Save	Save If there are any unsaved changes in the Pipeline, the message **Unsaved** is shown beside the **Save** icon. When the Pipeline is saved, the **Unsaved** message is changed to **Saved**.
Cancel	Cancel
	In Table View, use to select and expand/ collapse a stage within the Pipeline.
	Create Stage This command is only available in Grid View and Container View. It is not available in Table View.
	Create Job This command is only available in Grid View and Container View. It is not available in Table View.
	Success (in running the Pipeline)

Using the Table View

You can use the Pipeline table view to render stages and jobs within the Pipeline in a columns and row format. Pipelines shown in table view are an excellent way to present large sets of stages and jobs in a compact and easy-to-understand table format.

To use a Pipeline table view:

1. To go to the table view, click .



2. **(Optional)**: To expand all stages and jobs, click .

   To collapse all stages and jobs, click again.

3. Add a new Pipeline or select an existing Pipeline.

4. To add a new stage, from the drop-down, select **Stage** and then add the stage definition.

Copyright © 2017, 2026, Oracle and/or its affiliates.

5. To add a new job to a stage, click next to the stage to which to add a job, and then from the drop-down, select **Job** and then add the job definition.
6. Click .

> **Note**
>
> To cancel or undo any changes made in the Pipeline without saving changes, click **Cancel** or (Refresh icon).

Pipeline Job Types

Shows the job types available in the Pipeline.

The following job types are available to be added to a Pipeline:

- Using a Business Rule Job Type
- Using a Business Ruleset Job Type
- Using a Clear Cube Job Type
- Using a Copy from Object Storage Job Type
- Using a Copy from SFTP Job Type
- Using a Copy to Object Storage Job Type
- Using a Copy to SFTP Job Type
- Using a Create Reconciliations Job Type
- Using an EPM Platform Job Job Type for Planning Jobs
- Using an EPM Platform Job Job Type for Enterprise Profitability and Cost Management
- Using an EPM Platform Job Job Type for Financial Consolidation and Close and Tax Reporting Jobs
- Using an Export Dimension by Name (EDMCS) Job Type
- Using an Export Dimension Mapping by Name (EDMCS) Job Type
- Using an Export Mapping Job Type
- Using an Export Metadata Job Type
- Using an Extract Dimension Viewpoint (EDMCS) Job Type
- Using an Extract Package (EDMCS) Job Type
- Using a File Operations Job Type
- Using a Generate Report for Account Reconciliation Job Type
- Using an Import Attribute Values Job Type
- Using an Import Balances Job Type
- Using an Import Mapping Job Type
- Using an Import Metadata Job Type

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Using an Import Pre-Mapped Balances Job Type
- Using an Import Pre-Mapped Transactions (TM) Job Type
- Using Import Rates Job Type
- Using an Integration Job Type
- Using an Integration with Smart Split Job Type
- Using an Open Batch - File Job Type
- Using an Open Batch - Location Job Type
- Using an Open Batch - Name Job Type
- Using a Plan Type Map Job Type
- Using a Run Auto Alert Job Type
- Using a Run Auto Match Job Type
- Using a Set Period Status Job Type
- Using a Set Substitution Variable Job Type

Using a Business Rule Job Type

Use a Business Rule job type to add and run a business rule created in the Calculation Manager from the Pipeline. Business rules enable you to launch and manage sophisticated rules when loading data to Planning applications.

For more information on business rules, see About Business Rules in *Designing with Calculation Manager* .

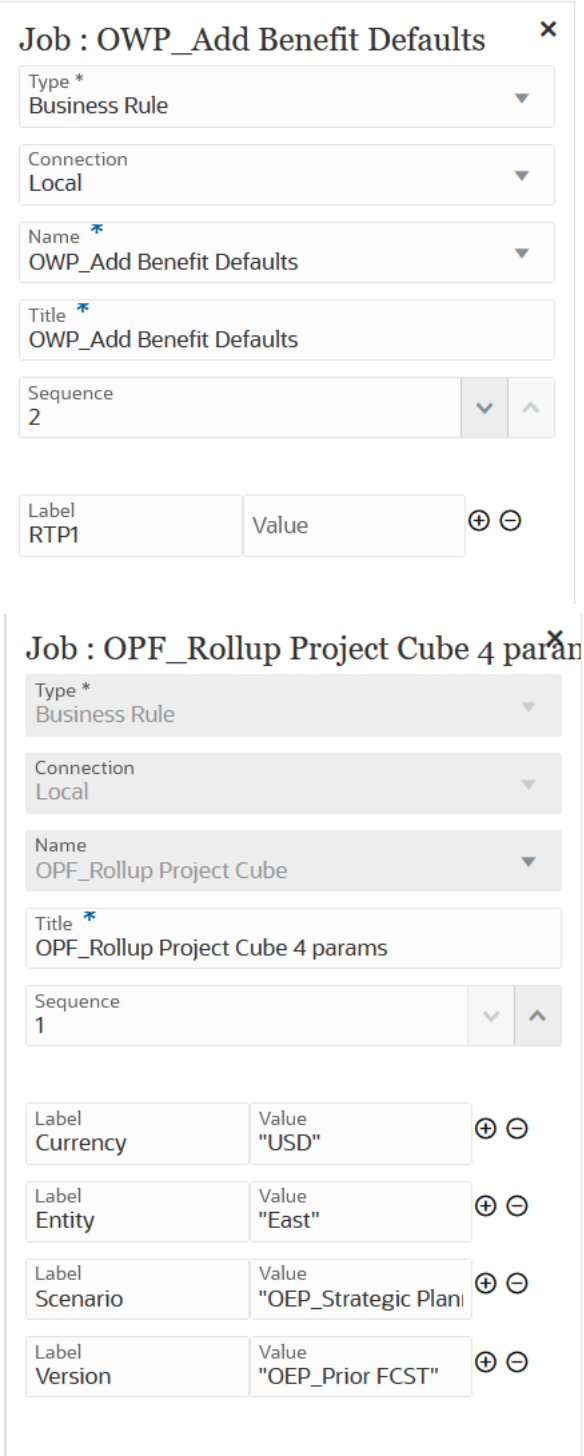
Business Rule job type parameters include:

Table 12-2 Business Rule Job Type Parameters

Business Rule Job Type Parameter	Description
Label	Type the name of the runtime prompt as it has been defined in the Planning business rule.
Value	If you selected a custom value type for a runtime prompt, specify the actual value.

Here are sample job parameters for a Business Rule job type:

Copyright © 2017, 2026, Oracle and/or its affiliates.



Using a Business Ruleset Job Type

Use a Business Ruleset job type to add and run a business ruleset created in the Calculation Manager from the Pipeline. A business ruleset enables you to launch and manage a collection of business rules simultaneously or sequentially when loading data to Planning applications.

For more information on business rulesets, see About Business Rulesets in *Designing with Calculation Manager* .

Copyright © 2017, 2026, Oracle and/or its affiliates.

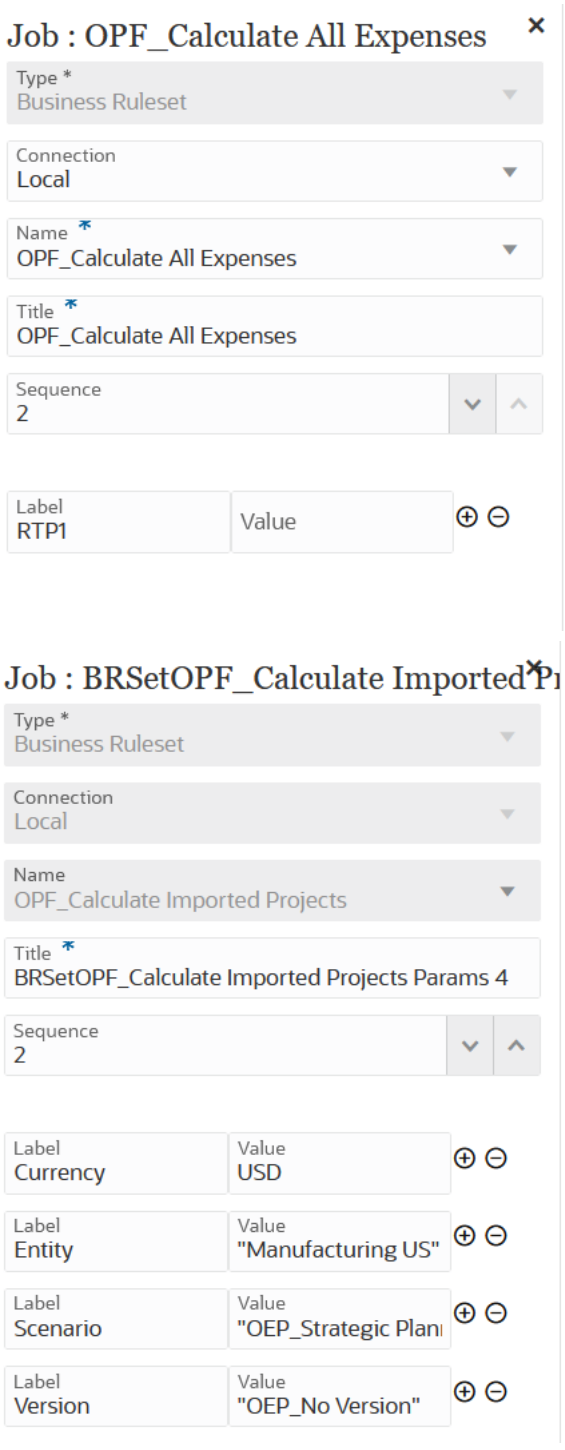
Business Ruleset job type parameters include:

Table 12-3 Business Rule Job Type Parameters

Business Ruleset Job Type Parameter	Description
Label	Type the name of the runtime prompt as it has been defined in the Planning business rule.
Value	If you selected a custom value type for a runtime prompt, specify the actual value.

Here are sample job parameters for a Business Ruleset job type:

Copyright © 2017, 2026, Oracle and/or its affiliates.



Using a Clear Cube Job Type

Use the Clear Cube job type to clear specific data within input and reporting cubes.

You can clear the data using a member selection or a valid MDX query using member selection. Optionally, you can clear related supporting details, comments, and attachments. You can also elect to do a physical or logical clear of data.

When this job type is executed, it does not delete the application definition in the application's relational tables. See Clearing Cubes in *Administering Planning*.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Optional runtime parameters are added as Label and Value pairs (key value pairs) where **Label** is the name of an attribute, and **Value** is an assigned value for this attribute.

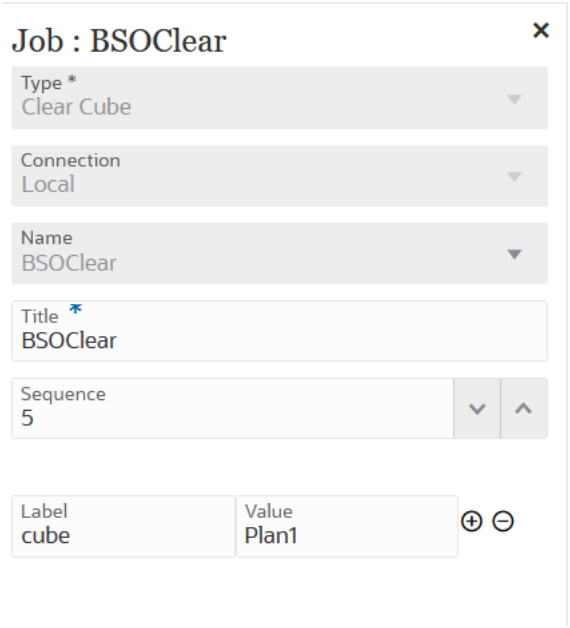
Clear Cube Job Type parameters include:

Table 12-4 Clear Cube Job Type Parameters

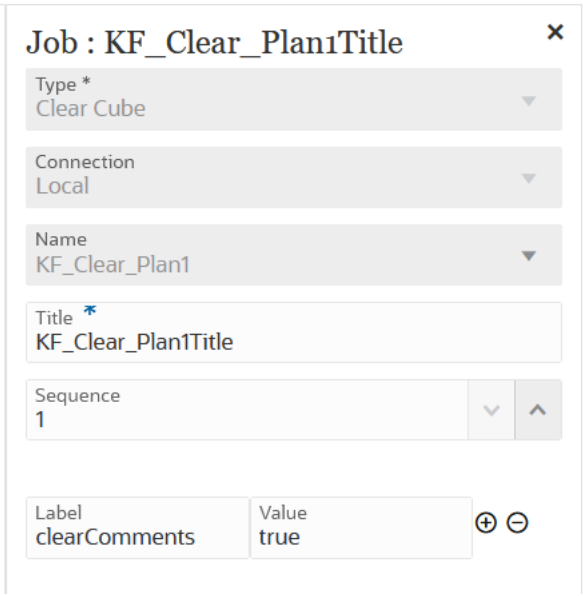
Clear Cube Job Type Parameters	Description
Name	From the **Name** drop-down, select the name of the cube from which data to clear data. The list of available all clear cubes is based on the selected connection.
Label/Value	Specify optional runtime labels and their values. The optional labels can include: • cube • members • mdxQuery • clearSupportingDetails • clearComments • clearAttachments • clearPhysicalOnEssbase To add a new Label/Value pair, click ⊕. To delete a Label/Value pair, click ⊖.
`cube`	Specify a valid cube name to clear.
`mdxQuery`	Valid MDX query. Applicable only for Partial Clear Job, for an ASO cube, defined with MDX query support. **Example**: `"Crossjoin({[Apr],[May],[Jun]},{[Expense1]})"`
`clearSupportingDetails`	Specify if supporting details should be cleared. Allowed values: `true` or `false`. Applicable only for Partial Clear Job, for an ASO cube, defined with member selection.
`clearComments`	Specify if comments should be cleared. Allowed values: `true` or `false`. Applicable only for Partial Clear Job, for an ASO cube, defined with member selection.
`clearPhysicalOnEssbase`	Specify if attachments should be cleared. Allowed values: `true` or `false`. Applicable only for Partial Clear Job, for an ASO cube defined with member selection.

Here are sample job type parameters for a BSO cube:

Copyright © 2017, 2026, Oracle and/or its affiliates.

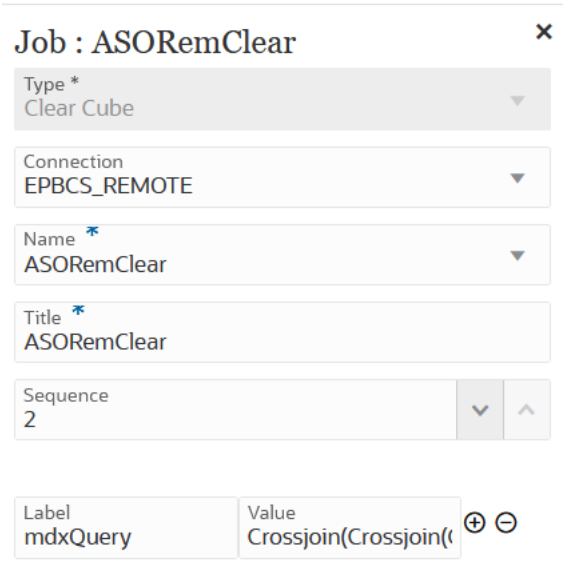


Here are sample job type parameters for a Clear Cube job type with clear comments parameters:

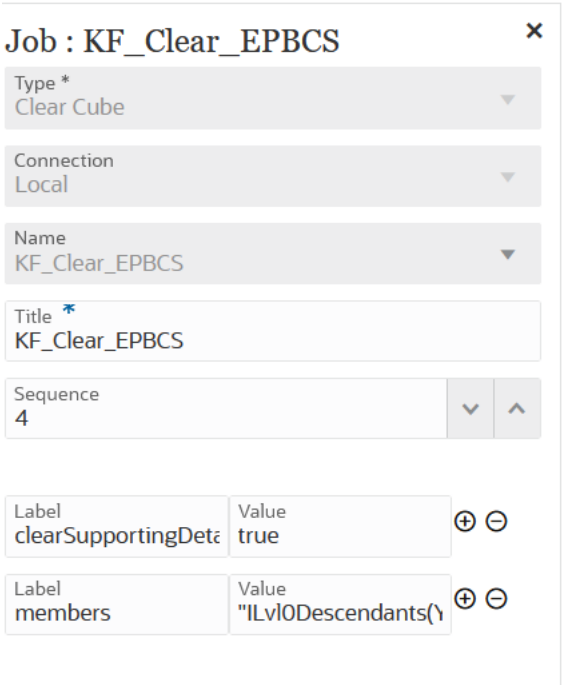


Here are the sample job type parameters for a Clear Cube job type for a MDX Query parameters:

Copyright © 2017, 2026, Oracle and/or its affiliates.



Here are the sample job type parameters for a Clear Cube job type for a MDX Query parameters:



Using a Copy from Object Storage Job Type

You can copy a file from the Object Storage to the `inbox`. The files can be input data source for an integration job. Object Storage is a fully programmable, scalable, and durable cloud storage service.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> Before copying a file from the Object Storage, note the following:
>
> - The Object Storage requires an *Other Web Services Provider* connection type. You must ensure you have access to the Web service to which you're connecting. You must also have URLs for the Web service and any login details, if required. For more information, see Connecting to External Web Services in *Administering Planning*.
> - You need to generate an auth token to use as the user password for an *Other Web Services Provider* connection type. For information on creating an auth token, see To create an auth token.
> - The copy from Object Storage job type requires that you specify the **namespace** name and **bucket** information associated with the file to be copied from Object Storage.
>
> For more information about Oracle Storage, see Oracle Storage.

Copy from Object Storage Rule job type parameters include:

Table 12-5 Copy from Object Storage Job Type Parameters

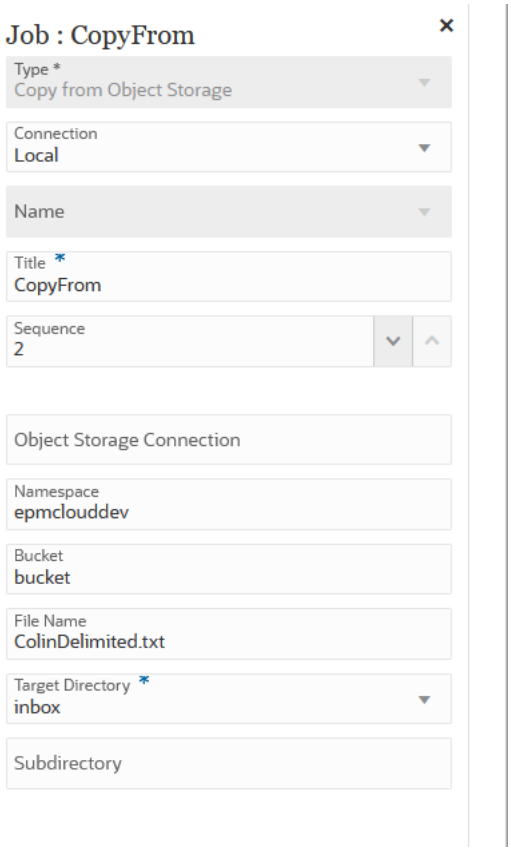
Copy from Object Storage Job Type Parameter	Description
Object Store Connection	Specify the name of the Object Storage connection.
Namespace	Specify the name that serves as the to-level container for all buckets and objects. namespace is the top-level container for all buckets and objects. Each Oracle Cloud Infrastructure tenant is assigned a unique system-generated and immutable Object Storage namespace name at account creation time. Your tenancy's namespace name, for example, `axaxnpcrorw5`, is effective across all regions. The namespace spans all compartments within a region. You control bucket names, but those bucket names must be unique within a namespace. While the namespace is region-specific, the namespace name itself is the same in all regions.
Bucket	Specify the name of the bucket, which is the container for storing objects in a compartment within a namespace. A bucket is the name of a logical container where you store your data and files. Buckets are organized and maintained under compartments. A system generated bucket name, for example, `bucket-20210301-1359` reflects the current year, month, day, and time. To use the Object Storage Service, first create a bucket and then begin adding data files.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 12-5 (Cont.) Copy from Object Storage Job Type Parameters

Copy from Object Storage Job Type Parameter	Description
File Name	Specify the source file name from the Object Storage to copy. This file name must exactly match the full name of the object in Object Storage Cloud. You can use the wild card character * (the asterisk) to copy multiple files. For example, specify the file name GLData* to copy all files starting with GLData.
Target Directory	Specify the target directory to which files are copied. The target directory can be: inbox, openbatch, openbatchml, and epminbox
Subdirectory	Optional: Specify the subdirectory under the Target Directory to which files are copied. The character limit for a sub-directory sting name is 2,000 characters.

Here are sample job parameters for a Copy from Object Storage job type:



Using a Copy from SFTP Job Type

Use the Copy from SFTP job type to copy a file directly from a secure SFTP server to an Oracle Fusion Cloud EPM environment.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> You *cannot* copy from SFTP to the `Outbox`.

If the IP allowlist is enabled for the SFTP server, you must add the outbound IP address of the OCI region that hosts the Cloud EPM environments to the IP allowlist of the SFTP server. See Outbound IP Addresses of Cloud EPM Data Centers and Regions in the *Operations Guide* for the outbound IP addresses of OCI regions.

> **Note**
>
> As a prerequisite to using the Copy from SFTP job type, you need to define the SFTP server connection using EPM Connections. For more information, see Connecting Environments in Cloud EPM in *Administering Planning*.

Copy from SFTP job type parameters include:

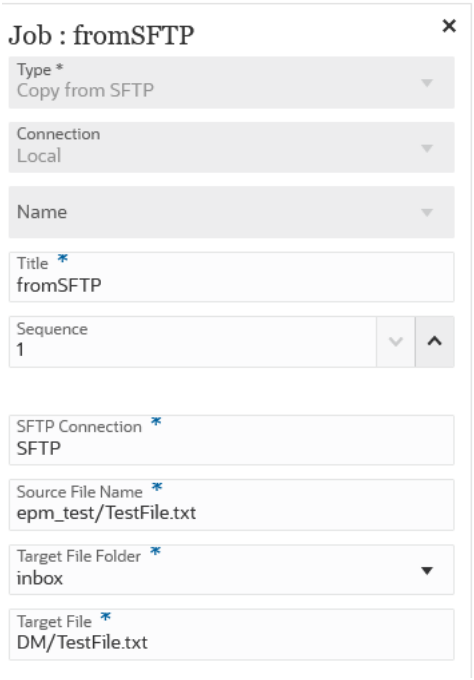
> **Note**
>
> For parameters that are common to all job types, see Pipeline Process Description.

Table 12-6 Copy from SFTP Job Type Parameters and Descriptions

Copy from SFTP Job Type Parameter	Description
SFTP Connection	Specify the connection of the SFTP server from which to copy the file, for example, `sftp://myserver`. This is the Connection name that is created in Connections.
Source File Name	Specify the location and name of the file you want to copy from the Cloud EPM. Specify the path if you are not copying from the default source location. For example, `mydir/File_New.txt`
Target File Folder	Specify the target directory to which files are copied. The target directory can be: Inbox (inbox folder in Data Integration), openbatch, openbatchml and the EPMInbox (Planning inbox folder). You cannot copy from SFTP to the `Outbox`.
Target File	Specify the name of the target file name of the downloaded artifact.

Here are sample job type parameters for a Copy from SFTP job type:

Copyright © 2017, 2026, Oracle and/or its affiliates.



Using a Copy to Object Storage Job Type

You can copy a file used as an input data source to the Object Storage, which is a fully programmable, scalable, and durable cloud storage service.

> **Note**
>
> Before copying a file to the Object Storage, note the following:
>
> - The Object Storage requires an *Other Web Services Provider* connection type. You must ensure you have access to the Web service to which you're connecting. You must also have URLs for the Web service and any login details, if required. For more information, see Connecting to External Web Services in *Administering Planning*.
> - You need to generate an auth token to use as the user password for an *Other Web Services Provider* connection type. For information on creating an auth token, see To create an auth token.
> - The copy from Object Storage job type requires that you specify the **namespace** name and **bucket** information associated with the file to be copied from Object Storage.
>
> For more information, see Oracle Storage.

The Copy to Object Storage job type parameters include:

Table 12-7 Copy to Object Storage Job Type Parameters

Copy to Object Storage Job Type Parameter	Description
Object Storage Connection	Specify the name of the Object Storage connection.

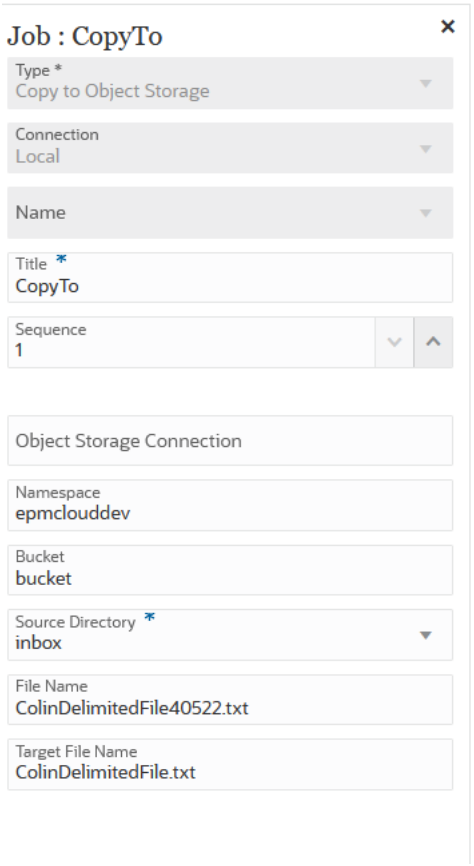
Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 12-7 (Cont.) Copy to Object Storage Job Type Parameters

Copy to Object Storage Job Type Parameter	Description
Namespace	Specify the name that serves as the to-level container for all buckets and objects. namespace is the top-level container for all buckets and objects. Each Oracle Cloud Infrastructure tenant is assigned a unique system-generated and immutable Object Storage namespace name at account creation time. Your tenancy's namespace name, for example, `axaxnpcrorw5`, is effective across all regions. The namespace spans all compartments within a region. You control bucket names, but those bucket names must be unique within a namespace. While the namespace is region-specific, the namespace name itself is the same in all regions.
Bucket	Specify the name of the bucket, which is the container for storing objects in a compartment within a namespace. A bucket is the name of a logical container where you store your data and files. Buckets are organized and maintained under compartments. A system generated bucket name, for example, `bucket-20210301-1359` reflects the current year, month, day, and time
Source Directory	Specify the source directory to which to copy the file to Object Storage. To move or copy an Oracle Fusion Cloud Enterprise Performance Management snapshot to Object storage, the snapshot needs to be in the *inbox/outbox* of the application. The character limit for a sub-directory sting name is 2,000 characters. Files larger than 100 MB are stored in Oracle Object Storage within a logical directory along with the manifest file that identifies its segments.
File Name	Specify the name of the source file to copy to the Object Storage. This file name must exactly match the full name of the object in Object Storage Cloud. You can click to launch the File Browser and navigate to the source file to copy. If you have multiple locations in the inbox, be sure to add the location name before the source file name. You can use the wild card character * (the asterisk) to copy multiple files. For example, specify the file name GLData* to copy all files starting with GLData.
Target File Name	Specify the target file name to copy to the Object Storage. When you copy multiple source file, the system appends the value specified as the file prefix.

Here are sample job parameters for a Copy to Object Storage job type:

Copyright © 2017, 2026, Oracle and/or its affiliates.



Using a Copy to SFTP Job Type

Use the Copy to SFTP job type to copy a file to a location on a Secure FTP server (the target). Using this job type , you can copy a file directly from an environment to an SFTP server without first downloading it to another server using EPM Automate or the REST API and then using an SFTP client to upload it to the SFTP server.

If the IP allowlist is enabled for the SFTP server, you must add the outbound IP address of the OCI region that hosts the Oracle Fusion Cloud EPM environments to the IP allowlist of the SFTP server. See Outbound IP Addresses of Cloud EPM Data Centers and Regions in the *Operations Guide* for the outbound IP addresses of OCI regions.

> **Note**
>
> This job type supports SFTP server running on port 22 only.

> **Note**
>
> As a prerequisite to using the Copy to SFTP job type, you need to define the SFTP server connection using EPM Connections. For more information, see: Connecting Environments in Cloud EPM in *Administering Planning*.

Copy to SFTP job type parameters include:

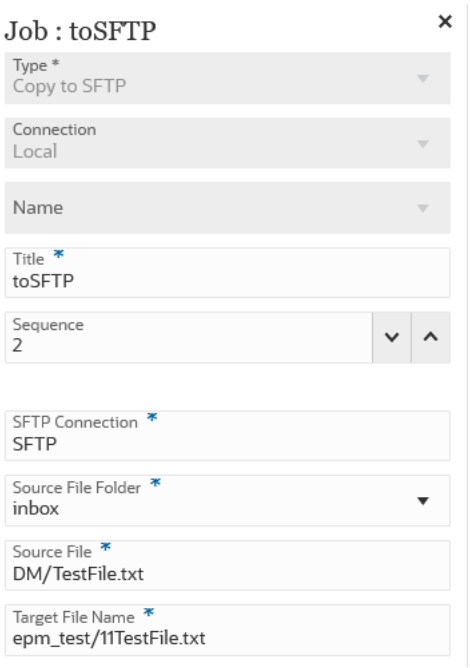
Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> For parameters that are common to all job types, see: Pipeline Process Description.

Table 12-8 Copy to SFTP Job Type Parameters and Descriptions

Copy to SFTP Job Type Parameters	Description
SFTP Connection	Specify the connection of the SFTP server where the file is to be copied and the file name. For example, `sftp://myserver`. This is the Connection name that is created in Connection.
Source File Folder	Specify the name and location of the file you want to copy from the Cloud EPM environment. The source directory can be: Inbox (inbox folder in Data Integration), openbatch, openbatchml and the EPMInbox (Planning inbox folder). For example, `inbox/File_new.txt`
Source File Name	Specify the name of the source file name (with path) to be copied to the SFTP Server.
Target File Name	Specify the name of the target file name (with path) to which to copy to the SFTP Server. For example, `inbox/File_new2.txt`

Here are sample job type parameters for a Copy to SFTP job type:



Copyright © 2017, 2026, Oracle and/or its affiliates.

Using a Create Reconciliations Job Type

Use the Create Reconciliations job type to copy all selected profiles to a period and return a success or failure status.

Create Reconciliations job type parameters include:

> **Note**
>
> For parameters that are common to all job types, see Pipeline Process Description.

Table 12-9 Create Reconciliations Job Type Parameters and Descriptions

Create Reconciliations job type parameters	Description
Period	Select the name of the period to which to copy profiles, such as July**2025**. If the period already contains reconciliations, the reconciliations with Pending status are overwritten with the information in the profile. Reconciliations with Open or Closed status are not updated. After the creation process is complete, a results summary is displayed. The summary displays the number of reconciliations created, the number of profiles for which reconciliation creation failed, and the reason for the failure.
Filters	Select the name of a saved filter to use for filtering profiles, such as **MyFilter**. If you do not specify a saved filter, the system copies all applicable profiles using public filters. If you specify the name of a private filter, the command ignores the filter and displays the `Filter not found` message.

Using a Data Load Essbase Rules File Job Type

Use the Data Load Essbase Rules File job type to leverage Essbase load rules for data loading, enabling customers migrating from on-premises Essbase environments to reuse their existing load rules and source files without converting them into Data Integration definitions, thereby reducing migration effort and preserving familiar workflows. By enabling direct data loads through Essbase load rules, the system also helps maintain comparable performance and avoids the additional processing overhead typically associated with Data Integration, ensuring a more efficient and seamless data loading experience in the cloud.

> **Note**
>
> For parameters that are common to all job types, see Pipeline Process Description

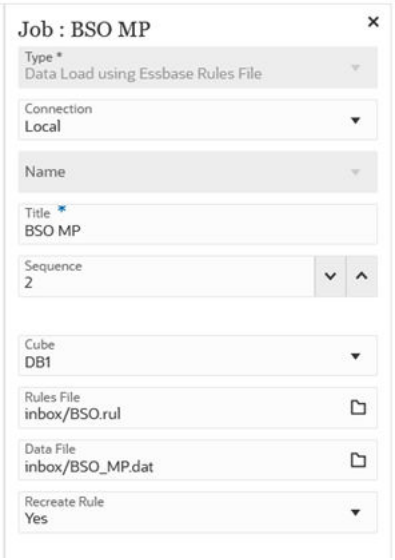
Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 12-10 Data Load Essbase Rules File Job Type Parameters

Data Load Essbase Rules File Job Type Parameters	Description
Cube	Specify a valid cube name to load.
Rules File	Upload rules file to inbox using File browser.
Data File	Upload data file to inbox using File browser
Recreate Rule	If set to **Yes**, system recreates the rules file if it already exists. If there are modifications to rule file, then the user can reload the rule file and set this flag to **Yes**. If set to **No**, system creates the rule file if it doesn't exist in the system. If it exists, it'll reuse the existing rules.

> **Note**
>
> Load rules that use SQL to pull data directly from relational sources and load rules that include SQL statements to load source data aren't supported. You must remove the SQL from the load rule and provide the load rule and data in separate files.

Here's a sample job parameter for Wait job type:



Using an EPM Platform Job Job Type for Planning Jobs

Use the EPM Platform Job job type to execute Planning jobs by providing the job name, job type, and parameters. Some jobs are expected to be defined in the corresponding Planning application, with some or all the required parameters saved within the job definition. For these job types, the parameters can be either provided within the job definition or overwritten at runtime. The remaining jobs can be triggered at runtime without a saved job definition.

Supported EPM Platform Job jobs types for Planning jobs include:

- Administration Mode
- Auto Predict

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Clear Cube
- Compact Cube
- Cube Refresh
- Execute Bursting Definition
- Export Audit
- Export Consolidation Journals
- Export Data
- Export Job Console
- Export Metadata
- Export Security
- Import Consolidation Journals
- Import Data
- Import Exchange Rates
- Import Metadata
- Import Security
- Merge Data Slices
- Optimize Aggregation
- Plan Type Map
- Restructure Cube
- Rules
- Ruleset

> **Note**
>
> To understand how jobs are defined in Planning, see Managing Jobs in *Administering Planning*.

Job type parameters are added as Label and Value pairs (key value pairs) where **Label** is the name of an attribute, and **Value** is an assigned value for this attribute.

The job type parameters name and parameters below should match the ones supported by the corresponding REST APIs (see Execute a Job in *REST APIs* .

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 12-11 EPM Platform Job Job Type Parameters for Planning

EPM Platform Job Job Type Parameters for Planning	Description
Job Type	From the **Job Type** drop-down, select the job type value or variable to execute in the EPM Platform Job. You can qualify the job type by specifying the individual label and value parameters in the Label/Value fields.
Job Name	Specify the job name to be used for this job execution.
Label/Value	Job type parameters are added as Label and Value pairs (key value pairs) where **Label** is the name of an attribute, and **Value** is an assigned value for this attribute. To add a new Label/Value pair, click ⊕. To delete a Label/Value pair, click ⊖.
Administration Mode	Changes the login level for an Planning application. If you set login level to **Administrators**, all Interactive Users and Planners are logged off of the application upon completion of the job. For details on the administration mode, see Scheduling Jobs. For detailed information on the job type parameters and values that can be passed, see Administration Mode in *REST API for Enterprise Performance Management Cloud*.
Auto Predict	Schedule predictions using the Auto Predict job. With Auto Predict, administrators can define a prediction to predict future performance based on historical data and schedule a job to run that prediction definition, automating the prediction process. For details about Auto Predict, see Setting Up Predictions to Run Automatically in *Administering Planning*. For detailed information on the job type parameters and values that can be passed, see Auto Predict in *REST API for Enterprise Performance Management Cloud.*
Clear Cube	Use the Clear Cube job type to clear specific data within input and reporting cubes. You can clear the data using a member selection or a valid MDX query using member selection, you can also optionally clear related supporting details, comments, and attachments. You can also elect to do a physical or logical clear of data. For detailed information on the job type parameters and values that can be passed, see Using a Clear Cube Job Type.
Compact Cube	Compacts the outline file of an ASO cube. Compaction helps keep the outline file at an optimal size. Compacting the outline does not clear the data. For detailed information on the job type parameters and values that can be passed, see Compact Cube in *REST API for Enterprise Performance Management Cloud.*
Cube Refresh	Refreshes the Planning application cube. Typically, you refresh the cube after importing metadata into the application. For detailed information on the job type parameters and values that can be passed, see Cube Refresh in *REST APIs* .

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 12-11 (Cont.) EPM Platform Job Job Type Parameters for Planning

EPM Platform Job Job Type Parameters for Planning	Description
Execute Bursting Definition	Execute bursting for a single report or book for more than one member of a single dimension, and publish a PDF or Excel output for each member. For information on bursting, see About Bursting in *Designing with Reports* . The bursting definition must be present in the folder that you specify with the `burstingDefinitionName` parameter. For detailed information on the job type parameters and values that can be passed, see Execute a Report Bursting Definition in *REST API for Enterprise Performance Management Cloud*.
Export Audit	Exports the audit records to a Comma Separated Values (CSV) file. The output CSV file contains the first character as a Byte Order Mark (BOM) character `\ufeff`. The API writes an encrypted application identifier following the BOM character. This application identifier is written between double-quotes. Headers for the CSV file follow the application identifier. For more information, see Auditing Tasks and Data. You can use an optional `excludeApplicationId` parameter to not write the application identifier in the export file. Exported audit reports without the application identifier cannot be imported back into the application. The generated CSV file is compressed and the output is a ZIP file. For detailed information on the job type parameters and values that can be passed, see Export Audit in *REST API for Enterprise Performance Management Cloud*.
Export Consolidation Journals	See Using an EPM Platform Job Job Type for Financial Consolidation and Close and Tax Reporting Jobs.
Export Data	Exports application data into a file using the export data settings, including file name, specified in a Planning job of type export data. The file containing the exported data is stored in the Planning repository. You can also override some of the parameters of the job definition while executing this job with a REST API. Exporting data supports substitution variables. You can use substitution variables while overriding the `rowMembers`, `columnMembers`, and `povMembers` definition. See Creating and Assigning Values to Substitution Variables in *Administering Planning*. For detailed information on the job type parameters and values that can be passed, see Export Data in *REST APIs* .
Export Job Console	Exports the job console records to a Comma Separated Values (CSV) file. The output CSV file contains the first character as a Byte Order Mark (BOM) character, `\ufeff`. The API writes an encrypted application identifier following the BOM character. This application identifier is written between double quotes. Headers for the CSV file follow the application identifier. You can use an optional `excludeApplicationId` parameter to not write the application identifier in the export file. Exported job console data files without the application identifier cannot be imported back into the application. The generated CSV file is compressed and the output is a ZIP file. For detailed information on the job type parameters and values that can be passed, see Export Job Console in *REST API for Enterprise Performance Management Cloud*.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 12-11 (Cont.) EPM Platform Job Job Type Parameters for Planning

EPM Platform Job Job Type Parameters for Planning	Description
Export Metadata	Exports metadata into a file using the settings specified in a Planning job of type `export metadata`. The file containing the exported metadata is stored in the Planning repository. For detailed information on the job type parameters and values that can be passed, see Export Metadata in *REST APIs* .
Export Security	Exports the security records or access control list (ACL) records for specified users or groups to a Comma Separated Values (CSV) file. For information about access permissions to application artifacts, see Setting Up Access Permissions. For detailed information on the job type parameters and values that can be passed, see Export Security in *REST API for Enterprise Performance Management Cloud*.
Import Consolidation Journals	See Using an EPM Platform Job Job Type for Financial Consolidation and Close and Tax Reporting Jobs.
Import Data	Imports data from a file in the Planning repository into the application using the import data settings specified in a Planning job of type `Import Data`. For Planning, Financial Consolidation and Close, and Tax Reporting, you can review the rejected data records that have errors. To do this, specify an error file that captures the data records that are not imported for each dimension. If an error file is specified, the ZIP file is stored in the Outbox where you can download the file using Inbox/Outbox Explorer or tools like EPM Automate or REST APIs, for example, with the Download API. For detailed information on the job type parameters and values that can be passed, see Import Data in *REST APIs* .
Import Exchange Rates	Import the Exchange Rates template in the Planning repository and change the rates if required. You can then import the rates into the application using the Import Exchange Rates settings specified in a Planning job of type import exchange rates. For more information, see Job Types. For detailed information on the job type parameters and values that can be passed, see Import Exchange Rates in *REST API for Enterprise Performance Management Cloud*.
Import Metadata	Imports metadata from a file in the Planning repository into the application using the import metadata settings specified in a Planning job of type `import metadata`. For Planning, Financial Consolidation and Close, and Tax Reporting, you can review the rejected data records that have errors. For detailed information on the job type parameters and values that can be passed, see Import Metadata in *REST APIs* .

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 12-11 (Cont.) EPM Platform Job Job Type Parameters for Planning

EPM Platform Job Job Type Parameters for Planning	Description
Import Security	Imports the security records or access control list (ACL) records from a Comma Separated Values (CSV) file. For information about access permissions to application artifacts, see Setting Up Access Permissions. Excel File Format: • Object Name: The name of the object for which the ACL is defined • Name: The name of the user or group for which the ACL is defined • Parent: The name of the parent of the object • Is User: The flag (Y or N) that determines if the ACL is defined for a user or for a group • Object Type: The type of object, for example, `Forms folder` • Access Type: The type of privilege, such as READ or READWRITE • Access Mode: Additional information, such as if the ACL is applicable on the descendants • Remove: To remove a particular ACL, set this to Y All possible values: Object Type: • SL_FORM - Form • SL_COMPOSITE - Composite Form • SL_TASKLIST - Tasklist • SL_CALCRULE - Rule • SL_FORMFOLDER - Form Folder • SL_CALCFOLDER - Rule Folder • SL_DIMENSION - Dimension • SL_CALCTEMPLATE - Template • SL_REPORT - Management Report • SL_REPORTSSHOT - Management Report Snapshot Access Type: • NONE - None • READ - Read • WRITE - Write • READWRITE - Read Write • LAUNCH - Launch Access Mode: • MEMBER • CHILDREN • @ICHILDREN • @DESCENDANTS • @IDESCENDANTS

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 12-11 (Cont.) EPM Platform Job Job Type Parameters for Planning

EPM Platform Job Job Type Parameters for Planning	Description

	CSV File Example:
	
	`Object Name,Name,Parent,Is User,Object Type,Access Type,Access Mode,Remove`
	
	`Object Name,Name,Parent,Is User,Object Type,Access Type,Access Mode,Remove`
	
	`"Exchange Rates to USD","ats_user3","Y","SL_FORM","READWRITE","MEMBER","N"`
	
	`"Exchange Rates to USD","ats_user4","Y","SL_FORM","READWRITE","MEMBER","N"`
	
	`"Exchange Rates to USD","ats_user15","Y","SL_FORM","READ","MEMBER","N"`
	
	`"Exchange Rates to USD","ats_user10","Y","SL_FORM","NONE","MEMBER","N"`
	
	`"Calculate Benefits","group_1","N","SL_COMPOSITE","READWRITE","MEMBER","N"`
	
	For detailed information on the job type parameters and values that can be passed, see Import Security in *REST API for Enterprise Performance Management Cloud*.

Merge Data Slices	Merges incremental data slices of an ASO cube. Fewer slices improve a cube's performance. You can merge all incremental data slices into the main database slice or merge all incremental data slices into a single data slice without changing the main database slice. You can optionally remove cells that have a value of zero. For detailed information on the job type parameters and values that can be passed, see Merge Data Slices in *REST API for Enterprise Performance Management Cloud*.

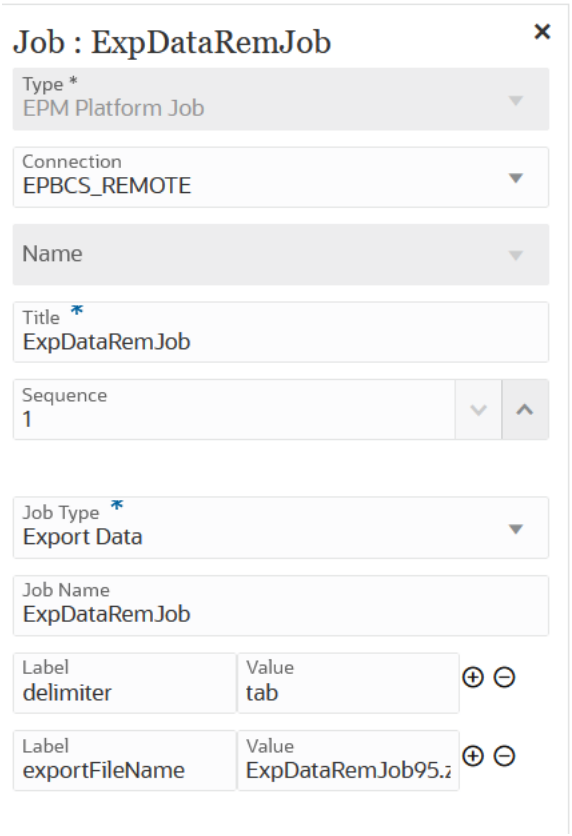
Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 12-11 (Cont.) EPM Platform Job Job Type Parameters for Planning

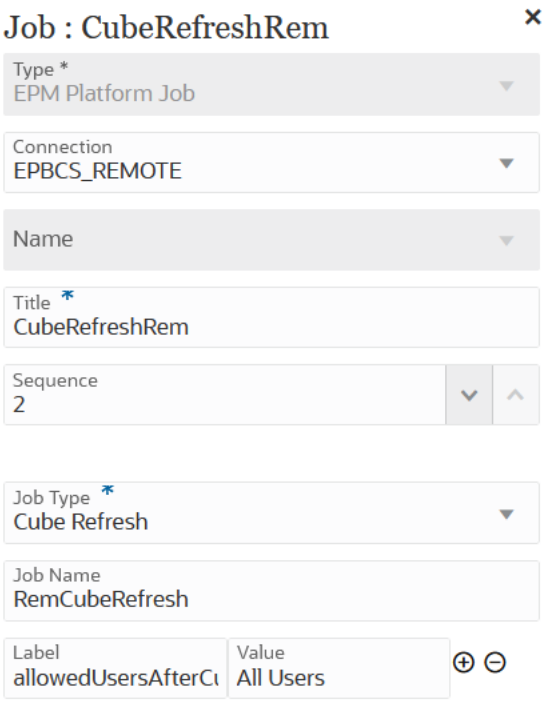
EPM Platform Job Job Type Parameters for Planning	Description
Optimize Aggregation	Improves the performance of ASO cubes. This job has two actions: Enable query tracking and Execute aggregation process. It performs an aggregation, optionally specifying the maximum disk space for the resulting files, and optionally basing the view selection on user querying patterns. This job type is only applicable to aggregate storage databases. Before using this job type, you must first enable query tracking to capture tracking statistics on the aggregate storage cube. Then, after you enable query tracking, you must allow sufficient time to collect user data-retrieval patterns before you execute the aggregation process based on query data. The execute aggregation process deletes existing aggregated views and generates optimized views based on the collected query tracking data. For detailed information on the job type parameters and values that can be passed, see Optimize Aggregation in *REST API for Enterprise Performance Management Cloud*.
Plan Type Map	Copies data from a block storage cube to an aggregate storage cube or from one to another based on the settings specified in a Planning job of type `plan type map`. For detailed information on the job type parameters and values that can be passed, see Plan Type Map in *REST APIs* .
Restructure Cube	Performs a full restructure of a BSO cube to eliminate or reduce fragmentation. For detailed information on the job type parameters and values that can be passed, see Restructure Cube in *REST API for Enterprise Performance Management Cloud*.
Rules	Launches a business rule. For detailed information on the job type parameters and values that can be passed, see Rules in *REST APIs* .
Ruleset	Launches a business ruleset. Supports rulesets with no runtime prompts or runtime prompts with default values. You can add parameters to rulesets for greater flexibility. For details about rulesets, see Designing Business Rulesets. For detailed information on the job type parameters and values that can be passed, see Ruleset in *REST APIs* .

Here are the sample Export Data parameters for an EPM Platform Job job type:

Copyright © 2017, 2026, Oracle and/or its affiliates.

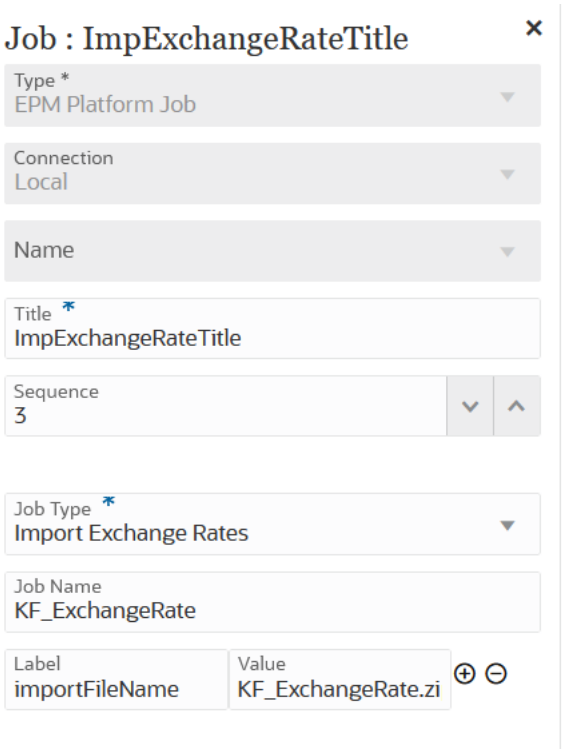


Here are the sample Cube Refresh parameters for an EPM Platform Job job type:

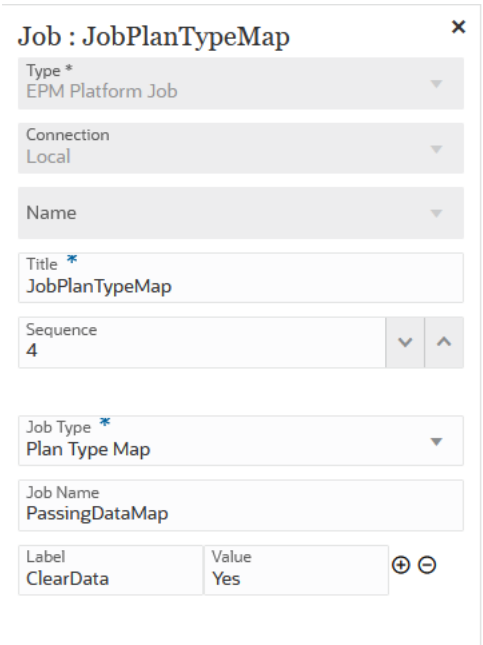


Here are the sample Import Exchange Rate job parameters for an EPM Platform Job job type:

Copyright © 2017, 2026, Oracle and/or its affiliates.



Here are the sample Plan Type Map job parameters for an EPM Platform Job job type:



Using an EPM Platform Job Job Type for Financial Consolidation and Close and Tax Reporting Jobs

Use the EPM Platform Job job type to execute Financial Consolidation and Close and Tax Reporting jobs by providing the job name, job type, and parameters. Some jobs are expected to be defined in the corresponding Financial Consolidation and Close and Tax Reporting applications, with some or all the required parameters saved within the job definition. For these

Copyright © 2017, 2026, Oracle and/or its affiliates.

job types, the parameters can be either provided within the job definition or overwritten at runtime. The remaining jobs can be triggered at runtime without a saved job definition.

The Supported EPM Platform Job job types for Financial Consolidation and Close and Tax Reporting are Clear Data and Copy Data.

> **Note**
>
> The Clear Job is available in the Pipeline, but it's not available from Platform on Financial Consolidation and Close.

Job type parameters are added as Label and Value pairs (key value pairs) where **Label** is the name of an attribute, and **Value** is an assigned value for this attribute.

The job type parameters name and parameters below should match the ones supported by the corresponding REST APIs:

- For Financial Consolidation and Close REST APIs, see Financial Consolidation and Close REST APIs in *REST API for Enterprise Performance Management Cloud*.
- For Tax Reporting REST APIs, see Tax Reporting REST APIs in *REST API for Enterprise Performance Management Cloud*).

Table 12-12 EPM Platform Job Job Type Parameters for Financial Consolidation and Close and Tax Reporting

EPM Platform Job Job Type Parameters for Financial Consolidation and Close and Tax Reporting	Description
Job Type	From the **Job Type** drop-down, select the job type value or variable to execute in the EPM Platform Job. You can qualify the job type by specifying the individual label and value parameters in the Label/Value fields.
Job Name	Specify the job name to be used for this job execution.
Label/Value	Job type parameters are added as Label and Value pairs (key value pairs) where **Label** is the name of an attribute, and **Value** is an assigned value for this attribute. To add a new Label/Value pair, click ⊕. To delete a Label/Value pair, click ⊖.
Clear Data	Execute a Clear Data job using the profile name. Before executing this job, you should create a Clear Data profile in Financial Consolidation and Close or Tax Reporting. For details on this task, see Clearing Data in *Administering Financial Consolidation and Close*. For detailed information on the job type parameters and values that can be passed in Financial Consolidation and Close, see Clear Data in *REST API for Enterprise Performance Management Cloud*. For detailed information on the job type parameters and values that can be passed in Tax Reporting, see Clear Data in *REST API for Enterprise Performance Management Cloud*.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 12-12 (Cont.) EPM Platform Job Job Type Parameters for Financial Consolidation and Close and Tax Reporting

EPM Platform Job Job Type Parameters for Financial Consolidation and Close and Tax Reporting	Description
Copy Data	Execute a Copy Data job using the profile name. Before executing this job, you should create a Copy Data profile in Financial Consolidation and Close or Tax Reporting. For details on this task in Financial Consolidation and Close, see Using Copy Data Profiles. For details on this task in Tax Reporting, see Using Copy Data Jobs in *Administering Tax Reporting* . For detailed information on the job type parameters and values that can be passed in Financial Consolidation and Close, see Copy Data in *REST APIs* . For detailed information on the job type parameters and values that can be passed in Tax Reporting, see Copy Data in *REST API for Enterprise Performance Management Cloud*.
Export Consolidation Journals	Execute an Export Consolidation Journals job using the job name. Before executing this job, you should create an Export Consolidation Journals job in Financial Consolidation and Close. For detailed information on the job type parameters and values that can be passed, see Export Consolidation Journals in *REST APIs* .
Import Consolidation Journals	Execute an Import Consolidation Journals job using the job name. Before executing this job, you should create an Import Consolidation Journals job in Financial Consolidation and Close. For detailed information on the job type parameters and values that can be passed, see Importing Consolidation Journals in *Working with Financial Consolidation and Close* . Import Consolidation Journals in *REST API for Enterprise Performance Management Cloud*.

Using an EPM Platform Job Job Type for Enterprise Profitability and Cost Management

Use the EPM Platform Job job type to execute Enterprise Profitability and Cost Management jobs by providing the job name, job type, and parameters. Some jobs are expected to be defined in the corresponding Enterprise Profitability and Cost Management application, with some or all the required parameters saved within the job definition. For these job types, the parameters can be either provided within the job definition or overwritten at runtime. The remaining jobs can be triggered at runtime without a saved job definition.

Supported EPM Platform Job jobs types for Enterprise Profitability and Cost Management include:

- Calculate Model
- Clear Data by POV
- Copy Data by POV

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Delete POV

Job type parameters are added as Label and Value pairs (key value pairs) where **Label** is the name of an attribute, and **Value** is an assigned value for this attribute.

Table 12-13 EPM Platform Job Job Type Parameters for Enterprise Profitability and Cost Management

EPM Platform Job Job Types for Enterprise Profitability and Cost Management	Description
Job Type	From the **Job Type** drop-down, select the job type value or variable to execute in the EPM Platform Job. You can qualify the job type by specifying the individual label and value parameters in the Label/Value fields.
Job Name	Specify the job name to be used for this job execution.
Label/Value	Job type parameters are added as Label and Value pairs (key value pairs) where **Label** is the name of an attribute, and **Value** is an assigned value for this attribute. To add a new Label/Value pair, click ⊕. To delete a Label/Value pair, click ⊖.
Calculate Model	Runs the calculation on a given point of view in a selected cube. This job type supports batch calculation with multiple POVs. This is an asynchronous call, so use the job status URI to determine whether the operation is complete. For detailed information on the job type parameters and values that can be passed, see Calculate Model in *REST API for Enterprise Performance Management Cloud*. You can only call Calculation Manager based business rules by way of the Pipeline. Enterprise Profitability and Cost Management based rules and rulesets is not supported.
Clear Data by POV	Clears the data for a given point of view in a selected cube. For detailed information on the job type parameters and values that can be passed, see Clear Data By Point of View in *REST API for Enterprise Performance Management Cloud*.
Copy Data by POV	Copies data from a source to a destination point of view in a selected cube. For detailed information on the job type parameters and values that can be passed, see Copy Data by Point of View in *REST API for Enterprise Performance Management Cloud*.
Delete POV	Deletes the data associated with a point of view from the calculation cube. For detailed information on the job type parameters and values that can be passed, see Delete Point of View in *REST API for Enterprise Performance Management Cloud*.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Using an Export Dimension by Name (EDMCS) Job Type

The Export Dimension by Name (EDMCS) job type enables you to export a dimension from an Oracle Enterprise Data Management Cloud application to a file in the staging area or a target environment defined in a connection.

The Export Dimension by Name (EDMCS) job type is useful when you need to export from a viewpoint that is bound to an Oracle Enterprise Data Management Cloud dimension back to one of the following types of Oracle Fusion Cloud Enterprise Performance Management applications:

- Planning and Planning modules
- FreeForm
- Financial Consolidation and Close
- Enterprise Profitability and Cost Management

> **Note**
>
> The Export Dimension by Name (EDMCS) job type supports other application types from the Oracle Enterprise Data Management Cloud in the Pipeline, but the application types listed above are the most commonly used by Cloud EPM customers. For information about other application types, see Understanding Application Types in *Administrating and Working with Enterprise Data Management Cloud*.

> **Note**
>
> If you need to extract and not export Oracle Enterprise Data Management Cloud dimensions into FreeForm or Planning business process, use the Extract a dimension viewpoint - REST API as described as a Business Rules (Groovy Script). For more information, see **Extract a dimension viewpoint**. Then call an internal REST API to load product volumes. For more information, see **Calling an internal REST API to load product volumes**

Export Dimension by Name (EDMCS) job type parameters include:

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 12-14 Export Dimension by Name (EDMCS) Job Type Parameters

Export Dimension by Name (EDMCS) Job Type Parameter	Description
Connection	Select the Oracle Enterprise Data Management Cloud connection to be used for this job type in the Pipeline. This job type is only supported for a remote operation (a non-local connection). The connection must be defined as an "Other Web Services Provider " connection type. The URL inside the connection must be in the format: `https://hostname.com/epm`. A URL ending with any other suffix than "epm" does not function properly. To create a new Oracle Enterprise Data Management Cloud connection, see Adding Pre-built Integrations within the EPM Cloud in *Administering Financial Consolidation and Close*.
Name	Select the application for the Oracle Enterprise Data Management Cloud connection. The list of applications populated in the drop-down is based on the available Oracle Enterprise Data Management Cloud applications (for more information, see Creating Application Connections in *Administrating and Working with Enterprise Data Management Cloud*). Each application in the drop-down list is concatenated with the application type for easy discovery.
Dimension Name	Specify the name of the Oracle Enterprise Data Management Cloud dimension to export.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 12-14 (Cont.) Export Dimension by Name (EDMCS) Job Type Parameters

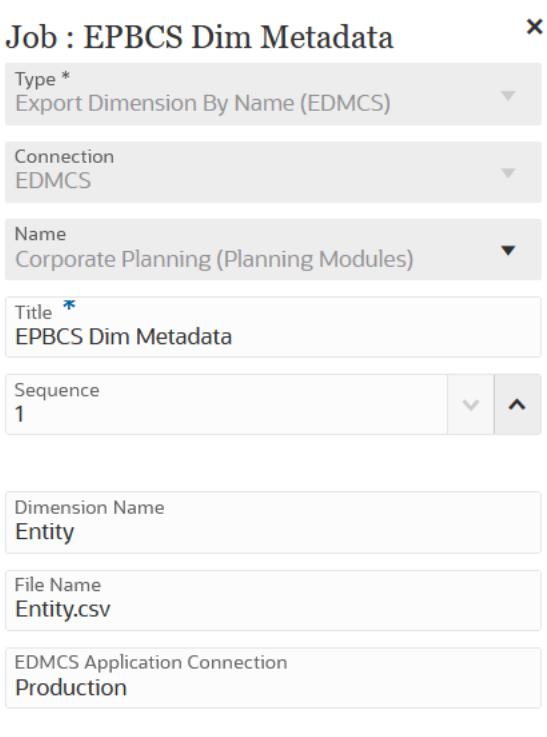
Export Dimension by Name (EDMCS) Job Type Parameter	Description
File Name	Specify the target file name to which the Oracle Enterprise Data Management Cloud dimension must be exported. You can export dimensions to a local CSV file or the target environment based on the application connection for each dimension for the following application types: • Planning • FreeForm • Financial Consolidation and Close • Enterprise Profitability and Cost Management When you specify an application connection on the **EDMCS Application Connection** page, the system exports the file to the *inbox/outbox* folder of the target Cloud EPM environment. When the file is exported to the *inbox/outbox* folder of the relevant Cloud EPM business process, it can be consumed by the application using the Import Metadata job type. See Using an Import Metadata Job Type for more information. The file name specified for the "Export Dimension by Name (EDMCS)" must satisfy one of the following two conditions: 1. The file name specified for the "Export Dimension by Name (EDMCS)" job type must match the file name specified in the saved Import Metadata job (see Using an Import Metadata Job Type for more information) in the Cloud EPM application. In this case, provide a blank file name in the "Import Metadata " job type. 2. If the file name specified in the "Export Dimension by Name (EDMCS)" job is different than the one specified in the saved "Import Metadata" job in the Cloud EPM application, the metadata import still works as long as this file name ends with `<dimensionname>.csv`. The same file name should also be provided to the Import Metadata job in the Pipeline, for example, `myprefix_Account.csv` or `prefix2_Entity.csv`. Here `<dimensionname>` refers to the Cloud EPM application dimension name and not the Cloud EPM application dimension.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 12-14 (Cont.) Export Dimension by Name (EDMCS) Job Type Parameters

Export Dimension by Name (EDMCS) Job Type Parameter	Description
EDMCS Application Connection	**Optional** Specify the application connection name defined in the Oracle Enterprise Data Management Cloud application. (This is the connection defined within the Oracle Enterprise Data Management Cloud application, and this is not a Global Connection.) Application connections are not supported or applicable for Universal application types. If specified, the export file is exported to the target environment (for example, the *inbox/outbox* of the Oracle Enterprise Data Management Cloud application.) If the Oracle Enterprise Data Management Cloud application connection is not supplied, the export file is written to the staging area of the Oracle Enterprise Data Management Cloud application. Files are always exported to the staging area for Universal application types. From here, files from the staging area can be downloaded using the File Operations job type (see Using a File Operations Job Type for more information.)

Here are sample Export Dimension by Name (EDMCS) job type parameters where an EPBCS metadata dimension is exported as a CSV file to a Corporate Planning application.



Copyright © 2017, 2026, Oracle and/or its affiliates.

Using an Export Dimension Mapping by Name (EDMCS) Job Type

The Export Dimension Mapping by Name (EDMCS) job type enables you to export mapping rules for a specific Oracle Enterprise Data Management Cloud dimension and location to a file in the staging area or a target environment defined in a connection.

You can export a dimension mapping by name only to a comma delimited text file (CSV). For a list of application types to which you can export a dimension mapping, see Understanding Application Types.

When the mapping is exported to an Oracle Fusion Cloud Enterprise Performance Management application, it can be imported as an Explicit mapping within Data Integration because the file is exported to the Data Exchange *inbox* folder. You can use the Import Mapping job type in the Pipeline to perform the import. See Using an Import Mapping Job Type for more information.

Export Dimension Mapping by Name (EDMCS) job type parameters include:

Table 12-15 Export Dimension Mapping by Name (EDMCS) Job Type Parameters

Export Dimension Mapping by Name (EDMCS) Job Type Parameters	Description
Connection	Select the Oracle Enterprise Data Management Cloud connection to be used for this job type in the Pipeline. This job type is only supported for a remote operation (a non-local connection). The connection must be defined as an "Other Web Services Provider" connection type. The URL inside the connection must be in the format: `https://hostname.com/epm`. A URL ending with any other suffix than "epm" does not function properly. To create a new Oracle Enterprise Data Management Cloud connection, see Adding Pre-built Integrations within the Cloud EPM in *Administering Financial Consolidation and Close*.
Name	Select the application for the Oracle Enterprise Data Management Cloud connection. The list of applications populated in the drop-down is based on the available Oracle Enterprise Data Management Cloud applications (for more information, see Creating Application Connections in *Administrating and Working with Enterprise Data Management Cloud*). Each application in the drop-down list is concatenated with the application type for easy discovery.
Dimension Name	Specify the name of the Oracle Enterprise Data Management Cloud dimension with mapping to export. You can export dimensions to a CSV file for each dimension.
File Name	Specify the target file name with a CSV extension to which to export the dimension mapping, for example: `mappings.csv`. In this case, you can download the file from the Oracle Enterprise Data Management Cloud staging area to a local *inbox* folder using the Download File Operation. See Using a File Operations Job Type.

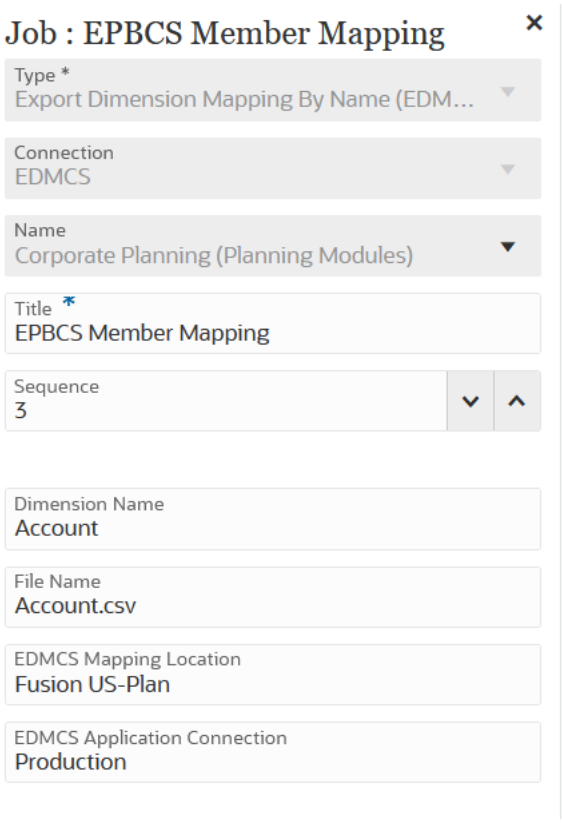
Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 12-15 (Cont.) Export Dimension Mapping by Name (EDMCS) Job Type Parameters

Export Dimension Mapping by Name (EDMCS) Job Type Parameters	Description
EDMCS Mapping Location	Specify the Application dimension bindings mapping location in Oracle Enterprise Data Management Cloud for which mapping rules should be exported. To find the location, navigate to the application, then select **Inspect**, then select the **Dimensions** tab, then select the dimension, then select the **Bindings** tab, then pick the specific dimension, then pick the binding, and then select the **Mappings** tab where you can see the location name.
EDMCS Application Connection	**Optional** Specify the application connection name defined in the Oracle Enterprise Data Management Cloud application. (This is the connection defined within the Oracle Enterprise Data Management Cloud application, and this is not a Global Connection.) Application connections are not supported or applicable for Universal application types. If specified, the export file is exported to the target environment (for example, the *inbox* of the Cloud EPM application. If the Oracle Enterprise Data Management Cloud application connection is not supplied, the export file is written to the staging area of the Oracle Enterprise Data Management Cloud application. Files are always exported to the staging area for Universal application types. In this case, files from the staging area can be downloaded using the File Operations job type (see Using a File Operations Job Type for more information.)

Here are sample Export Dimension Mapping by Name (EDMCS) job type parameters where an Account dimension name is exported to an Account.csv file to a Fusion US-Plan location.

Copyright © 2017, 2026, Oracle and/or its affiliates.



Using an Export Mapping Job Type

The Export Mapping job type in the Pipeline enables you to export member mappings to a .CSV or .TXT file format.

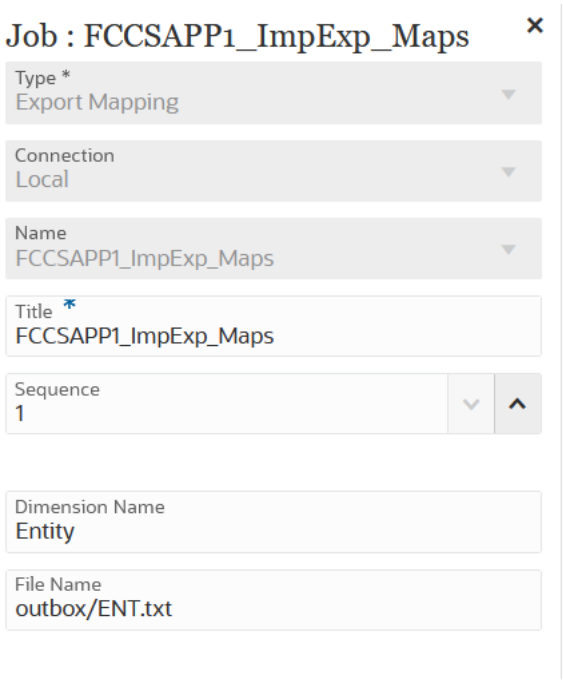
Export Mapping job type parameters include:

Table 12-16 Export Mapping job type parameters

Export Mapping Job Type Parameters	Description
Name	Select the name of the Data Integration location where the member mappings are to be exported. Member mappings are specific to the location in Data Integration and are based on the select connection ("local" or "remote").
Dimension Name	Specify the dimension name for a specific dimension to export, such as `ACCOUNT`, or `ALL` to export all dimensions.
File Name	Specify the file and path to which to export mappings. The file format can be: • .CSV • .TXT • Excel (.XLS, or .XLSX) When specifying the file name, include the outbox in the file path, for example, `outbox/BESSAPPJan-06.csv`

Here are sample Export Mapping job type parameters where the "Entity" dimension at the FCCSAPPI_ImpExp_Maps location will be loaded as a text file to the `outbox`.

Copyright © 2017, 2026, Oracle and/or its affiliates.



Using an Export Metadata Job Type

Use the Export Metadata job type to export metadata to a flat file in a .csv (comma-delimited) or .txt (tab-delimited or other delimiter character) format.

The system creates an export file for each artifact (.csv or .txt depending on the file type), and all export files are consolidated into one ZIP file.

> **Note**
>
> You must extract the .csv or .txt files from the ZIP file if you want to use the files as import files (for example, when importing metadata into another application).

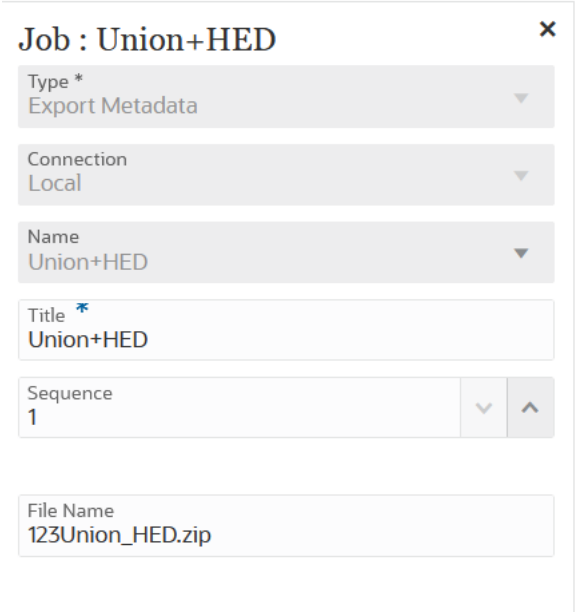
Export Metadata job type parameters include:

Table 12-17 Export Metadata Job Type Parameters and Descriptions

Export Metadata Job Type Parameters	Description
File Name	Specify the name of the file to which metadata is to be exported. If you do not specify a file name, then the default name is the Planning job name with a ZIP extension. The file output is always in ZIP format regardless of the file name extension provided in this parameter to the default download location from where you can download it to a local computer.

Here are sample job parameters for an Export Metadata job type:

Copyright © 2017, 2026, Oracle and/or its affiliates.



Using an Extract Dimension Viewpoint (EDMCS) Job Type

Use the *Extract Dimension Viewpoint (EDMCS)* job type to run a viewpoint extract process in the Oracle Enterprise Data Management Cloud. The extract output file can be written to the Oracle Enterprise Data Management Cloud staging area or transferred to a global connection target.

This job type is only supported for a remote operation (a non-local connection). To create a new Oracle Enterprise Data Management Cloud connection, see: Adding Pre-built Integrations within the Cloud EPM in *Administering Financial Consolidation and Close*. The connection must be defined as an "Other Web Services Provider" connection type. The URL inside the connection must be in the format: `https://hostname.com/epm`. A URL ending with any suffix other than "epm" does not function properly.

For more information about extracts, see: Working With Extracts in *Administrating and Working with Enterprise Data Management Cloud*.

Extract Dimension Viewpoint (EDMCS) job type parameters include:

Table 12-18 Extract Dimension Viewpoint (EDMCS) Job Type Parameters and Descriptions

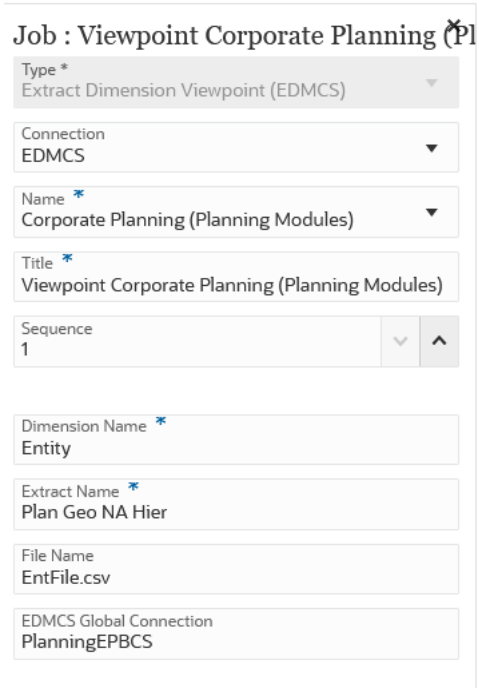
Extract Dimension Viewpoint (EDMCS) Job Type Parameters	Description
Name	Select the name of the application in Oracle Enterprise Data Management Cloud that the extract belongs to. Required.
Title	Specify the title of the job in the pipeline. This parameters defaults to the name of the application.
Dimension Name	Specify the name of the dimension in Oracle Enterprise Data Management Cloud to extract. Required.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 12-18 (Cont.) Extract Dimension Viewpoint (EDMCS) Job Type Parameters and Descriptions

Extract Dimension Viewpoint (EDMCS) Job Type Parameters	Description
Extract Name	Select the name of the extract. Required.
File Name	The target file name to extract the dimension viewpoint to. If the Zip Extract option is disabled in the extract profile, use a text file name. If the Zip Extract option is enabled, use the Zip file name. For example, you could specify: `dimension.csv`.
(EDMCS) Global Connection	Specify the name of the Oracle Enterprise Data Management Cloud global connection to extract to when extracting to a connection instead of the Oracle Enterprise Data Management Cloud staging area.

Here are sample Extract Dimension Viewpoint (EDMCS) job type parameters where "Entity" is the dimension to extract.



Using an Extract Package (EDMCS) Job Type

Use the Extract Package (EDMCS) job type to run an extract package in Oracle Enterprise Data Management Cloud. The extract package results can be output as individual files or a combined file based on the parameters of the extract package. Extract package files can be written to the Oracle Enterprise Data Management Cloud staging area or transferred to a global connection target.

This job type is only supported for a remote operation (a non-local connection). To create a new Oracle Enterprise Data Management Cloud connection, see: Adding Pre-built Integrations within the Cloud EPM in *Administering Financial Consolidation and Close*. The connection

Copyright © 2017, 2026, Oracle and/or its affiliates.

must be defined as an "Other Web Services Provider" connection type. The URL inside the connection must be in the format: `https://hostname.com/epm`. A URL ending with any suffix other than "epm" does not function properly.

> **Note**
>
> For more information about extract packages, see Working with Extract Packages in *Administrating and Working with Enterprise Data Management Cloud*.

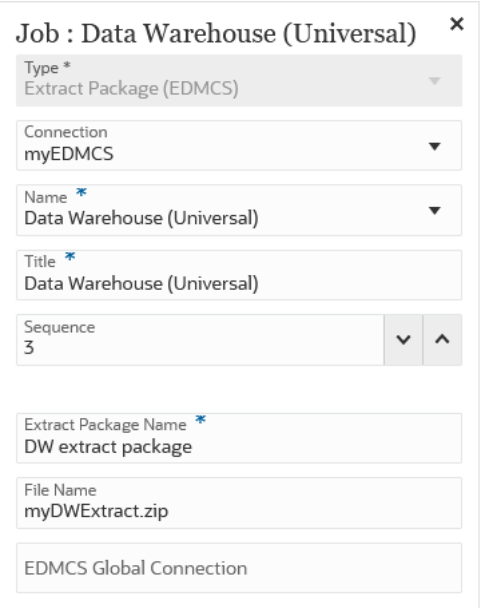
Extract Package (EDMCS) job type parameters include:

Table 12-19 Extract Package (EDMCS) Job Type Parameters and Descriptions

Extract Package (EDMCS) Job Type Parameters	Description
Name	Select the name of the application in Oracle Enterprise Data Management Cloud the extract package belongs to. Required.
Title	Specify the title of the job in the pipeline. This parameter defaults to the name of the application. Required.
Extract Package Name	Specify the name of the extract package. Required.
File Name	Specify the target file name to extract to. The file name must include the .zip extension. If the extract package is configured with a connection and with the Zip Extract parameter disabled, then extract package files will be individually written to the global connection target. Required.
EDMCS Global Connection	Specify the name of the Oracle Enterprise Data Management Cloud global connection to write extract package results to when extracting to a connection instead of the Oracle Enterprise Data Management Cloud staging area. Optional.

Here are sample Extract Package (EDMCS) job type parameters.

Copyright © 2017, 2026, Oracle and/or its affiliates.



Using a File Operations Job Type

Use the File Operations job type to run the following operations at runtime:

- Copy a file (using a local connection)
- Delete a file (using a local connection)
- Move a file (using a local connection)
- Unzip a file (using a local connection)
- Upload a file (using a remote connection)
- Download a file (using a remote connection)

File Operations job type parameters include:

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 12-20 File Operation Job Type Parameters

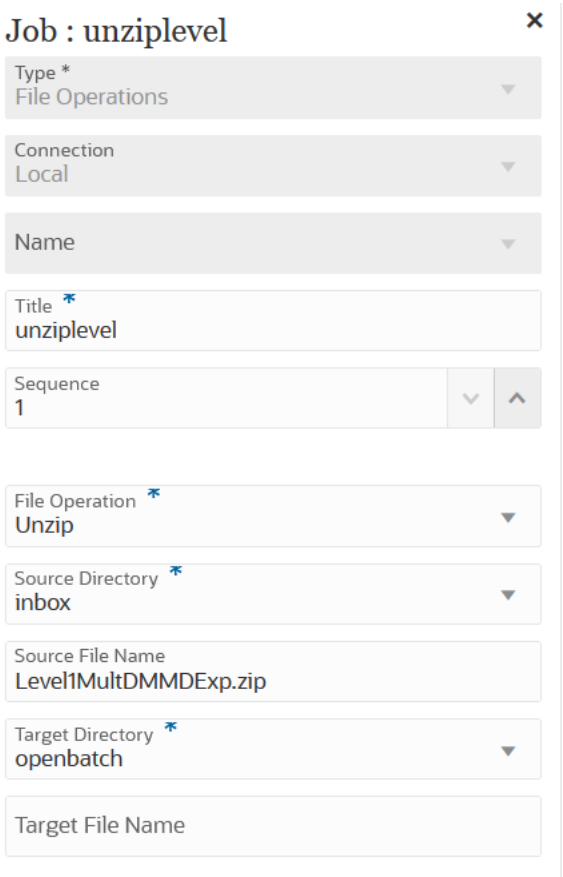
File Operations Job Type Parameters	Description
File Operation	Select the type of file operation to be executed at runtime: Option types: • Copy—The Copy file operation copies the file from a source directory to a target directory and retains the original file in the source directory after the copy operation to a target directory (used for local file operations, that is, within the same environment). Requires a local connection. • Delete—The Delete file operation deletes a file from a source directory. Requires a local connection. • Move—The Move file operation moves the file from a source directory to a target directory, but does not retain the moved file in the source directory after the move operation to a target directory. The Move file operation is a local operation and requires a local connection. • Unzip a file. Requires a local connection. • Upload—Copy a file from a local Oracle Fusion Cloud Enterprise Performance Management server to another Cloud EPM server (a local Cloud EPM server is the server where the Pipeline process is running.) This file operation requires that you specify the target directory and target file name on the remote machine in order for the file to be uploaded remotely from the local server. Upload is a remote operation and requires a remote connection. • Download—Copy a from file from a remote Cloud EPM server to local Cloud EPM (a local Cloud EPM server is the server where the Pipeline process is running.) This file operation requires that you specify the source directory and the source file name on the remote machine in order for the file to be downloaded locally from the remote machine. Requires a remote connection. Note For a *dat* file, Download file works only with a *dat* file generated as part of the integration output or if the integration has a target file name in Options.
Source Directory	Specify the source directory from which to copy, move, unzip, upload, or download the file. A Delete file operation requires that you specify a source directory. When you download a file, the source directory and source file name identify the remote source directory and file name. When you upload a file, the source directory and file identify from where the file is uploaded locally.
Source File Name	Specify the source file name of the file to copy, move, unzip, upload, or download.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 12-20 (Cont.) File Operation Job Type Parameters

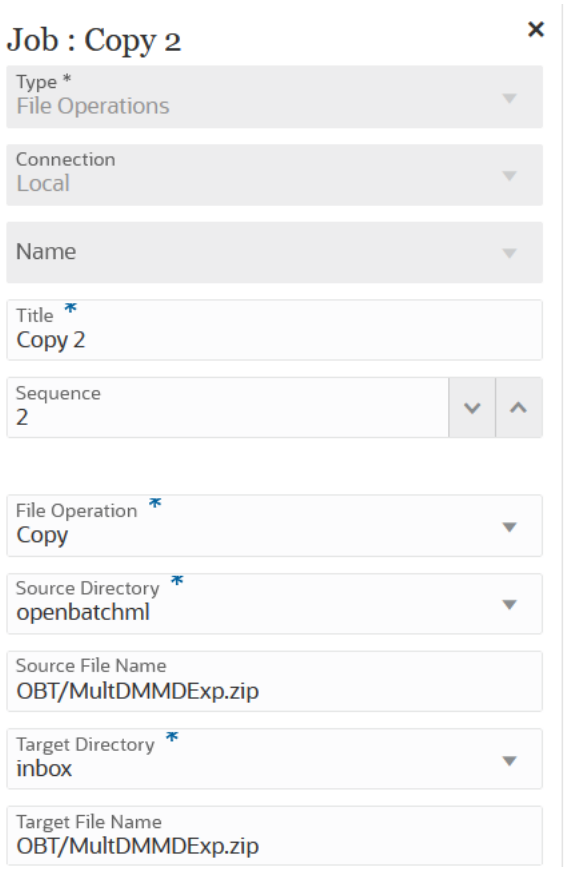
File Operations Job Type Parameters	Description
Target Directory	Specify the target directory to which files are copied. The target directory can be: `inbox` (inbox folder in Data Integration), `openbatch`, `openbatchml`, and `epminbox` (inbox folder in Planning). A zip file can contain only one level of a sub-directory. When you download a file, the target directory and file name identify to where the file is downloaded locally. When you upload a file, the target directory and target file name identify the remote target directory and file name.
Target File Name	Specify the target file name of the file that has been copied, moved, unzipped, uploaded, or downloaded. A target file name is not required for an "Unzip" file operation.

Here are sample File Operations job type parameters used to unzip files from the `inbox` source directory to an `openbatch` target directory:



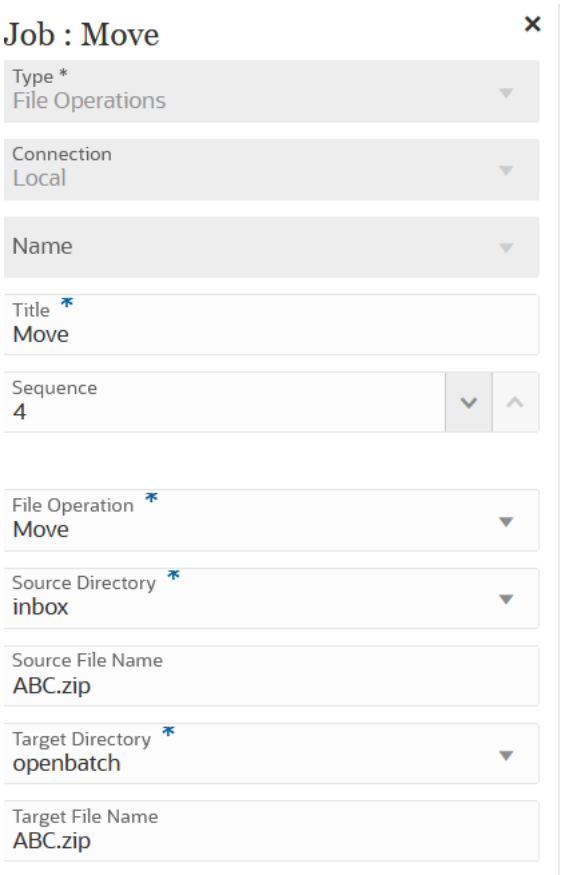
Here are sample File Operations job type parameters used to copy an **OBT/MultDMMDExp.zip** file from the `inbox` source directory to the `openbatch` target directory.

Copyright © 2017, 2026, Oracle and/or its affiliates.

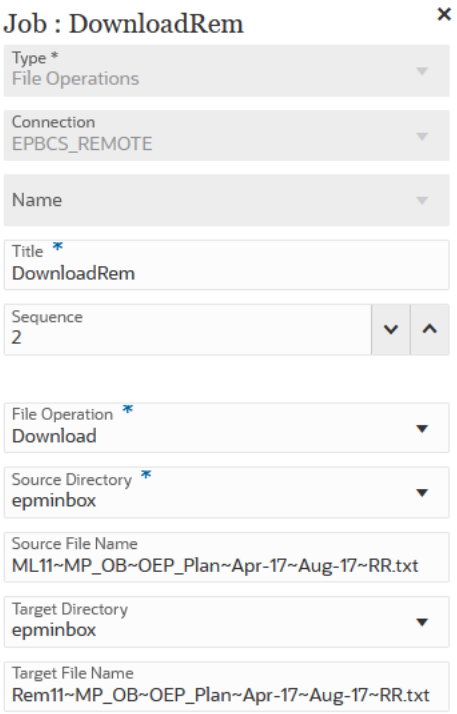


Here are sample File Operations job type parameters used to move the **ABC.zip** file from the `inbox` source directory to the `openbatch` target directory:

Copyright © 2017, 2026, Oracle and/or its affiliates.

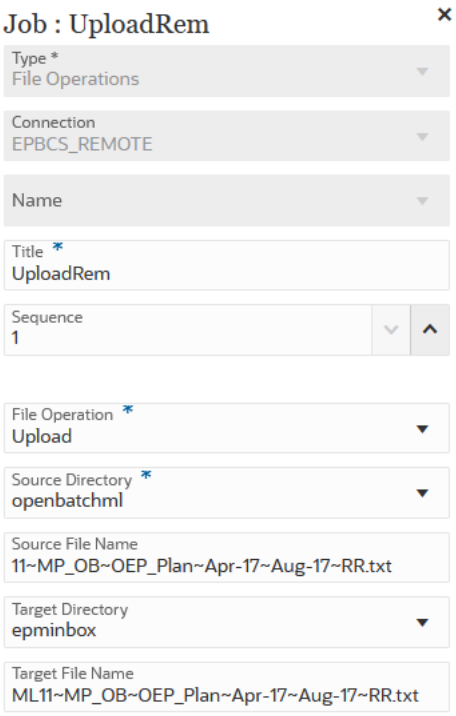


Here are sample File Operations job type parameters used to download the **ML11~MP_OB~OEP_Plan~Apr-17~Aug-17~RR.txt** file from the `epminbox` source directory to the `epminbox` target directory:



Copyright © 2017, 2026, Oracle and/or its affiliates.

Here are sample File Operations job type parameters used to upload the **ML11~MP_OB~OEP_Plan~Apr-17~Aug-17~RR.txt** file from the `epminbox` source directory to the `epminbox` target directory:



Using a Generate Report for Account Reconciliation Job Type

Generates either a single predefined Reconciliation Compliance report, predefined Transaction Matching report or a custom report.

General Report for Account Reconciliation Job Type parameters include:

> **Note**
>
> All parameters must be specified for a report.

> **Note**
>
> For parameters that are common to all job types, see Pipeline Process Description.

Table 12-21 Generate Report for Account Reconciliation Job Type Parameters and Descriptions

Generate Report for Account Reconciliation Job Type Parameter	Description
Report Group	Select the name of the group the report is associated with. If the Report Group is not specified, an error occurs

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 12-21 (Cont.) Generate Report for Account Reconciliation Job Type Parameters and Descriptions

Generate Report for Account Reconciliation Job Type Parameter	Description
Report Name	Specify the name of the report to be generated. If the Report Name is not specified, an error occurs.
Name	Specify a unique file name for the report that will be generated. If this parameter is not provided, then the report is generated with the data for `Report Name` parameter value. Include a file extension to the Name field so the report is emailed in selected format, for example, Report1.pdf.
Format	Select the format of the report: • PDF • HTML (not supported for graphs and charts) • XLSX (not supported for graphs) • CSV • CSV2 The `CSV` format does not permit the formatting of data based on a template while `CSV2` does. Generating a `CSV2` formatted report takes more time compared to `CSV` output. The default format is **PDF**.
Module	Select the module within Account Reconciliation to which the report applies. Available options: • Reconciliation Compliance • Transaction Matching The default module is **Reconciliation Compliance**.
E-Mails	Specify a comma separated list of email addresses that will receive the report once it's generated.
Run Asynchronously	Select to run the report asynchronously by selecting **true**) or synchronously **false**. Oracle recommends setting this value to **true** (asynchronously) for larger reports.
Label/Value	(Optional): Specify any required parameters for generating the report. For example, the `Balance By Account Type` report takes two parameters `Period` with the value `July 2017` and `Currency Bucket` with the value `Entered`. You should specify these parameters as Label/Value = `Period=July 2017` and Label/Value = `Currency Bucket=Entered` .

Using an Import Attribute Values Job Type

For Reconciliation Compliance, use the Imports Attribute Values job type to import attribute values into an existing list attribute or group attribute. For Transaction Matching, you can only import group attributes using this job type.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Import Attribute Values job type parameters include:

> **Note**
>
> For parameters that are common to all job types, see Pipeline Process Description.

Table 12-22 Import Attribute Values Job Type Parameters and Descriptions

Import Attribute Values Job Type Parameters	Description
Attribute	Specify the name of the list attribute or group attribute into which values must be imported.
Import Type	Select the option that indicates the action to be taken when there are existing values for one or more rates. Available: • Replace—All rate values from the import file will be added to or will replace the existing rate values. Existing rate values that are not in the import file will not be changed. Use this import type when you are only moving the latest changes from a source system. For example, adding new store data from an acquisition. • Replace All—All values from the import file will completely replace the existing rate values. Existing rate values that are not in the import file will be deleted. Use this import type when you are mirroring values from a source system with a full update. For example, weekly updates to synchronize with store data from your ERP system. • Update—Compares using the key attribute and updates member attribute values with those in the file that you are importing. There is no effect on values for key attributes that are not specified in the import file. All values from the import file are added to or will replace the existing attribute values. Existing attribute values that are not in the import file will not be changed. Only attribute data for a particular key value will be replaced with the contents from the file. Attribute data for attributes that are not in the file are not changed. Any key values in the import file that are not in the attribute will cause an error. Use this import type when you want to update a few attributes across all attribute values. For example, updating the store managers after a reorganization, without affecting the rest of the store data.
File Location	Select the `.csv` file that contains the member values. If the import file contains an invalid value (that is, a value that is not present in the group attribute), that attribute is left blank and it can be updated subsequently.
Date Format	Specify the valid date formats (for example, `DD/MM/YYYY`, `DD-MMM-YYYY` (default), `MMM d, yyyy`, and `All`) to parse. You may specify multiple date format values separated using a semicolon.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 12-22 (Cont.) Import Attribute Values Job Type Parameters and Descriptions

Import Attribute Values Job Type Parameters	Description
Module	Select the name of the module for which the import group attributes process applies. Available options: • Reconciliation Compliance • Transaction Matching

Using an Import Balances Job Type

Use the Import Balances job type to import balance data using Data Integration from a previously created integration definition and return a success or failure status

Refer to the Data Integration log file to understand the reason for the job failure. If multiple data load rules fail, a separate log is created for each data load rule.

Import Balances job type parameters include:

> **Note**
>
> For parameters that are common to all job types, see Pipeline Process Description.

Table 12-23 Import Balances Job Type Parameters and Descriptions

Import Balances Job Type Parameters	Description
Period	Select the name of the period from which to import balances such as `July2016`.
Data Load Definition	Select the name of a previously saved integration definition.

Using an Import Mapping Job Type

The Import Mapping job type enables you to import member mappings from a .CSV or .TXT file format.

The Import Mapping job type supports Merge or Replace modes, along with validate or no validate options for target members.

Import Mapping job type parameters include:

Table 12-24 Import Mapping Job Type parameters

Import Mapping Job Type Parameter	Description
Name	Select the name of the Data Integration location where the member mappings are to be loaded. Member mappings are specific to the location in Data Integration and are based on the select connection ("local" or "remote.")

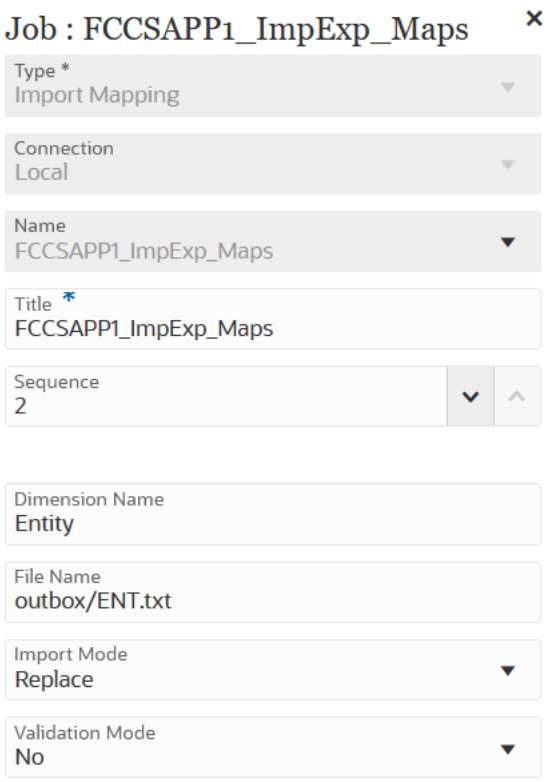
Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 12-24 (Cont.) Import Mapping Job Type parameters

Import Mapping Job Type Parameter	Description
Dimension Name	Specify the dimension name for a specific dimension to import, such as `ACCOUNT`, or `ALL` to import all dimensions.
File Name	The file and path from which to import mappings. The file format can be .CSV, .TXT, .XLS, or .XLSX. The file must be uploaded prior to importing, either to the inbox or to a sub-directory of the inbox. Include the inbox in the file path, for example, `inbox/BESSAPPJan-06.csv`.
Import Mode	Specify the import mode: `MERGE` to add new rules or replace existing rules, or `REPLACE` to clear prior mapping rules before importing. Available options: • `MERGE`—Add new rules or replace existing rules. • `REPLACE`—Clear prior mapping rules before import.
Validation Mode	Specify whether either to use validation mode. Specify **Yes** to validate the target members against the target application. Specify **No** to load the mapping file without any validations. The validation process is resource intensive and takes longer when the validation mode is **Yes**.

Here are sample Import Mapping job type parameters where the "Entity" dimension at the FCCSAPPI_ImpExp_Maps location will be loaded as a text file to the `inbox`.

Copyright © 2017, 2026, Oracle and/or its affiliates.



Using an Import Metadata Job Type

Use the Import Metadata job type to import metadata from a flat file in a comma-delimited, tab-delimited, or ZIP format.

> **Note**
>
> For information on uploading files to a server or downloading them to your computer, see Uploading and Downloading Files Using the Inbox/Outbox Explorer in *Administering Planning*.

Table 12-25 Import Metadata Job Type Parameters and Descriptions

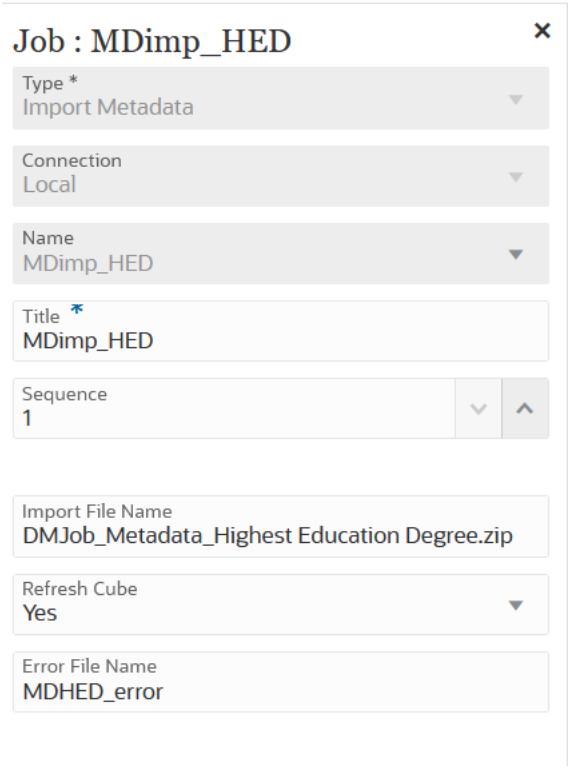
Import Metadata Job Type Parameters	Parameters
Import File Name	Specify the name of the file in comma-delimited (CSV) tab-delimited (TXT), or ZIP format from which metadata is to be imported. If specified, the contents of a ZIP file take precedence over the file names defined in the job. The ZIP file may contain one or more .csv files. The file names containing metadata for dimensions should match the import file names defined in the job or end with: `_DIMENSIONNAME.csv`; for example, `metadata_Entity.csv`. `metadata_HSP_Smart Lists.csv`, and `metadata_Exchange Rates.csv`.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 12-25 (Cont.) Import Metadata Job Type Parameters and Descriptions

Import Metadata Job Type Parameters	Parameters
Refresh Cube	Select **Yes** to refresh the application cube. Typically, you refresh the cube after importing metadata into an application. The time required to complete a cube refresh operation depends on the changes that you made to the application structure and the impact it has on the cube. For example, a refresh after updating a sparse block storage cube member may not take much time while a cube refresh after updating a dense block storage cube member or an aggregate storage cube member could take a considerable amount of time. You must ensure that the cube refresh operation is complete before the application is backed up during the next maintenance window.
Error File Name	Specify the name of the ZIP file in which rejected records, if any, during the import operations will be recorded. An identically named ZIP file in the outbox, if any, is overwritten.

Here are sample job parameters for an Import Metadata job type:



Using an Import Pre-Mapped Balances Job Type

Use the Import Pre-Mapped Balances job type to import pre-mapped balances from a CSV file in the Account Reconciliation repository and return the success or failure status.

Import Pre-Mapped Balances job type parameters include:

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> For parameters that are common to all job types, see Pipeline Process Description.

Table 12-26 Import Pre-Mapped Balances Job Type Parameters and Descriptions

Import Pre-Mapped Balances Job Type Parameters	Description
Period	Select the name of the period from which to import pre-mapped balances such as `July2016`.
Balance Type	Select the balance type of the pre-mapped balances. Available balance types: • source system • sub-system
Currency Bucket	Select the currency bucket type: Available options: • Entered • Functional • Reporting
File	Select the name of the name of the CSV file containing the data to be imported, such as `balances.csv`.

Using an Import Pre-Mapped Transactions (TM) Job Type

Use the Import Pre-Mapped Transactions (TM) job type to import pre-mapped transactions for a particular period from a CSV file in the Account Reconciliation repository and return the success or failure status.

Import Pre-Mapped Transactions (TM) job type parameters:

> **Note**
>
> For parameters that are common to all job types, see Pipeline Process Description.

Table 12-27 Import Pre-Mapped Transactions (TM) Job Type Parameters and Descriptions

Import Pre-Mapped Transactions (TM) Job Type Parameters	Description
Match Type	Select a Match Type from the list where the transaction file will be imported to. This list will contain Match Types with the status: Approved. If the Match Type ID is changed in Account Reconciliation after Account Reconciliation pipelines have been created in Data Integration, you must re-select the Match Type LOV option in the pipelines and save the changes. This step is necessary to ensure the pipelines run successfully.
Data Source	Specify the text ID of the data source where the transaction will be imported to.
File	Select the name of the CSV file from which data is to be imported.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 12-27 (Cont.) Import Pre-Mapped Transactions (TM) Job Type Parameters and Descriptions

Import Pre-Mapped Transactions (TM) Job Type Parameters	Description
Date Format	Select the data format. The date format is a parameter that includes the format of the date fields in the transactions import file. The default is `dd-MMM-yy`. Other supported date formats are: • `MM/dd/yyyy` • `dd/MM/yyyy` • `MM-dd-yyyy` • `d-M-yyyy` • `MMM d, yyyy`

Using Import Rates Job Type

Use the Import Rates Job Type to import currency rates based on a particular period and rate type from a CSV file in the Account Reconciliation repository and returns the success or failure status.

Import Rates job type parameters include:

> **Note**
>
> For parameters that are common to all job types, see Pipeline Process Description.

Table 12-28 Import Rates Job Type Parameters and Descriptions

Import Rates Job Type Parameters	Description
Period	Select the name of the period from which to import currency rates, such as `July2016`.
Rate Type	Select the rate type, such as `Accounting` when you require translation of transactions entered into the reconciliation. Configure only rate types used for period end balance translation in the source systems being reconciled.
File	Select the name of the file from which to import rates, such as `rates.csv`.
Import Type	Select the option that indicates the action to be taken when there are existing values for one or more rates. Available: • Replace—All rate values from the import file will be added to or will replace the existing rate values. Existing rate values that are not in the import file will not be changed. • Replace All—All values from the import file will completely replace the existing rate values. Existing rate values that are not in the import file will be deleted.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Using an Integration Job Type

Use an Integration job type to add and execute or run integrations from the Pipeline. You can add any integration created in Data Integration that extracts metadata and data from an Enterprise Resource Planning (ERP) source system (a direct integration) or data from a file, and then push it to an Oracle Fusion Cloud Enterprise Performance Management target application.

Integration job type parameters include:

Table 12-29 Integration Job Type Parameters

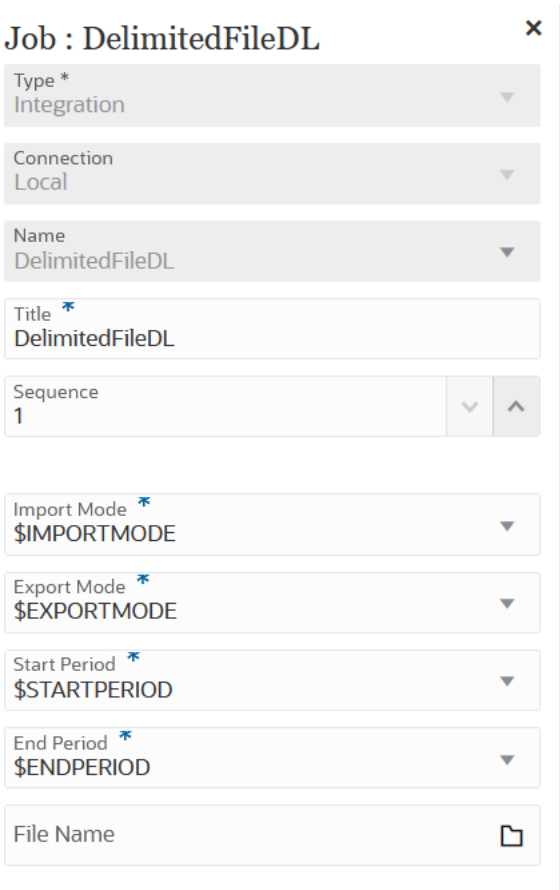
Integration Job Type Parameter	Description
Import Mode	**Optional**: From the **Import Mode** drop-down, select the import mode for the integration job. By default, the $IMPORTMODE parameter uses the value of the variable parameter defined in global variables for the import mode (see Editing Runtime Variables). However, you can select different import modes for different jobs. For example, to load metadata in one integration and just data in another, specify different import modes between the two jobs.
Export Mode	**Optional**: From the **Export Mode** drop-down, select the export mode for the integration job. By default, the $EXPORTMODE parameter uses the value of the variable parameter defined in global variables for the export mode (see Editing Runtime Variables). However, you can select different export modes for different jobs. For example, to load metadata in one integration and just data in another, specify different export modes between the two jobs.
Start Period	**Optional** From the **Start Period** drop-down, select the Start Period for the integration job. By default, the $STARTPERIOD parameter uses the value of the variable parameters defined in global variables for the start period (see Editing Runtime Variables). However, you can select different start periods for different jobs. For example, if you load metadata in a Pipeline, you can set the Start Period to be BegBalance in the job.
End Period	**Optional** From the **End Period** drop-down, select the End Period for the integration job. By default, the $ENDPERIOD parameter uses the value of the variable parameters defined in global variables for the start period (see Editing Runtime Variables). However, you can select different end periods for different jobs. For example, if you load metadata in a Pipeline, you can set the End Period to be EndBalance in the job.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 12-29 (Cont.) Integration Job Type Parameters

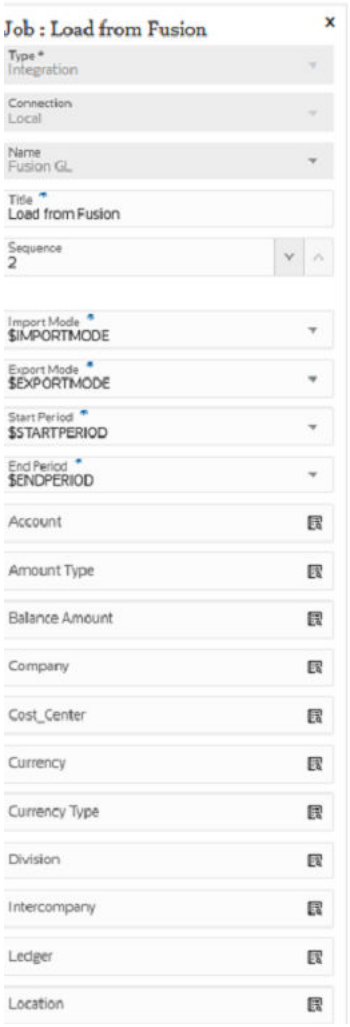
Integration Job Type Parameter	Description
File Name	In **File Name**, if no file profile has been selected for the integration, specify the file name to use at run time. Click to launch the File Browser and navigate to the target file. The directory specified here is appended to the directory specified in the integration definition options section. Provide only the directory relative to the directory in the integration definition. For example, if the `inbox` is specified in integration definition, then include only the location/filename here. If no directory is specified, then include: `inbox/location/filename`. If a Pipeline job specifies an integration on a remote server, you can specify a file in Pipeline job parameters on the local server, and this file is copied to the remote server and used as the input file for the integration on the remote server. For more information about integration options, see Defining File-Based Options.

Here are sample job parameters for a file-based Integrations job type:



Here are the sample job parameters for a Load from Fusion integration job type:

Copyright © 2017, 2026, Oracle and/or its affiliates.



Using an Integration with Smart Split Job Type

When importing a large volume of data that exceeds Essbase governor limits, use the **Integration with Smart Split** job type to split a data load into multiple smaller loads. This job type enables you to define smaller data slices without having to create multiple integrations.

> **Note**
>
> Smart Split is optimally used for extracting calculated data, which is determined by your data extract option. There are multiple Essbase extract options based on the type of your data extract option you select in .Data Integration. If you select "Stored Data", then this is applicable to BSO and uses the BSO data export method. All data uses "MDX" export method. Level 0 uses "MAXL" (Smart Split is not supported when the Level 0 Data Extract Option is enabled. Level 0 extracts all level 0 data and filters extracted data.) For more information about data extract options, see Defining Direct Integration Options.

To perform a Smart Split:

1. Create the base integration definition between the source application and the target Oracle Fusion Cloud Enterprise Performance Management application.

Copyright © 2017, 2026, Oracle and/or its affiliates.

a. For information about creating an integration, see: Creating Direct Integrations or Creating File-Based Integrations.

b. Map the dimensions between the Cloud EPM application and the dimensions in the target application.

   For more information, see Mapping Dimensions.

c. Map members to translate source values to valid members in each target dimension.

   For more information, see Mapping Members.

d. On the **Options** page, select any filters and application options.

   You can apply a filter for a Split Dimension as well as filters for other dimensions in the base integration. When specifying filters for the Split dimension in the base integration, you can simply provide the parent member name for the hierarchy branch to process or provide the member function ILvl0Descendants.

   The filter for the Split dimension supports only ILvl0Descendant member functions. If no filters are specified, then the system processes all level 0 members for the Split dimension.

   For more information on apply filters and application options, see Defining Filters.

2. Create a new Pipeline by completing steps 1-12 in the Pipeline Process Description.

3. On the **Pipeline** page, click to create a new stage card.

   In the **Stage Editor**, specify the stage definition:

   - **Stage Name**—Specify the name of the stage.
   - **Title**—Specify the name of the stage to appear on the stage card.
   - **Sequence**—Specify a number to define the chronological order in which a stage is executed.
   - **Parallel**—Toggle **Parallel** on to run jobs simultaneously.
   - **On Success**—Select how to process a stage when steps in the Pipeline definition are executed successfully.
   - **On Failure**—Specify how to process a stage when a step within a stage fails.

4. On the stage card, click **>** to expand the stage.

5. On the stage card, click (Create Job icon) to add a new job card.

6. In the **Job Editor**, then from **Type** drop-down, select **Integration with Smart Split**.

   > **Note**
   >
   > Add only one **Integration with Smart Split** job type per stage when running the Smart Split in parallel mode.

7. Complete the following

   a. From **Name**, select the name of the job.

   b. In **Title**, specify the title of job name to appear on the job card.

   c. In **Sequence**, select **1** for the run job order in the stage.

8. From **Split Dimension**, select the dimension from the list of dimensions in the source application on which to split the data loads.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Select only **one** dimension as the Split Dimension.

For example, select **Ledger** to run a data load per Ledger.

Split Dimension on Period and Year Considerations:

"Period" and "Year" are special dimensions and require special processing when used as a Split dimension

a. When "Period " is used as the Split Dimension, only Single and Group are supported split methods. If other methods are used, an error occurs.

b. When "Year" is used as the Split Dimension, only Single is the supported split method. Running multiple years in a single integration run is not supported. If other methods are used, an error occurs.

c. For Standard Mode, the **Start Period** and **End Period** are used to determine the list of periods to split. Based on the "Period" range, the system determines the source periods and applies the source period filter.

d. For Quick Mode, source filters for the "Period" and "Year" are used to determine the period or year for each child integration and applies the source filter.

9. From **Split Method**, select how members in the dimension are split (separated). The system creates a separate child integration (data rule) for split criteria in the dimension based on the split method.

   Each child integration is a copy of the base integration and is available on the Data Integration home page. The child integration is identical to the base integration except for the source filter used for the split dimension. Child integrations are recreated every time the Pipeline is executed, so do not modify the child integrations.

   The child integrations can be deleted using the **Delete Integration** system maintenance task. For more information, see Deleting an Integration.

   Available options:

   - **Single**—System runs a separate process for each member in the dimension. For example, you can run a load per "Ledger" or "Entity."
   - **Group**—System splits the list of members in dimensions into multiple groups. The size of the group is based on the **Number of Groups** parameter.
   - **Custom** (Pattern)—You specify the values of for the split dimension member. A Custom Pattern split method can be specified as a list, range, wild-card in the **Label/Value** parameter below. The split is processed in the order of filter name.

     For example, you might specify the following custom pattern using these expressions:

     `Like US*`: All members starting with US

     `Between 100-199`: All members in the 100 range

     `List 100,200,300`: List of individual members

     If a member is picked up by earlier filter, then it is not included the later filters.
   - **Custom-Functions**—You specify the values of for the split dimension member using member functions. This option enables you to limit members to a specific subset of members.

     Custom-Functions can be specified as member functions in the **Label/Value** parameter below.

     If you are using a custom function, then do not define filters for the Split Dimension in the base integration. Only the filter of the custom dimension is applied.

Copyright © 2017, 2026, Oracle and/or its affiliates.

10. From **Processing Mode**, select the mode for processing the jobs in the Pipeline.

    Available options:

    Parallel—When jobs are run in parallel mode, at runtime, the system runs jobs together in parallel (not sequentially).
    Serial—When jobs are run in serial mode, at runtime, the system runs the jobs one after another in a specific sequence.

11. From **Import Mode**, select the import mode for the integration job.

    By default, the $IMPORTMODE parameter uses the value of the variable parameter defined in global variables for the import mode (see Editing Runtime Variables). However, you can select different import modes for different jobs. For example, to load metadata in one integration and just data in another, specify different import modes between the two jobs.

    > **Note**
    >
    > When you run the integration in Replace mode, the system runs the Clear data process by "Entity" and other POV dimensions. If the Split dimension is by "Entity", then there are no issues with default Replace mode behavior.
    > If the Split Dimension is by Entity and a governor limit exception occurs, and you select a different dimension as the Split Dimension, then there are additional considerations.
    >
    > For Planning and FreeForm applications, define the Clear Region instead of the default Clear region. In the Clear Region, select the Split Dimension as one of the dimensions using " Derive from Data" as the option. If the Split Dimension used for the source filter does not exist in the target application (for example, Ledger), then define a "Clear Cube" job in the platform and call the Clear Cube job prior to running the integration job. In this case, the Integration job must be run in Merge mode.

12. From **Export Mode**, select the export mode for the integration job.

    By default, the $EXPORTMODE parameter uses the value of the variable parameter defined in global variables for the export mode (see Editing Runtime Variables). However, you can select different export modes for different jobs. For example, to load metadata in one integration and just data in another, specify different export modes between the two jobs.

13. From **Start Period** drop-down, select the Start Period for the integration job.

    By default, the $STARTPERIOD parameter uses the value of the variable parameters defined in global variables for the start period (see Editing Runtime Variables). However, you can select different start periods for different jobs. For example, if you load metadata in a Pipeline, you can set the Start Period to be BegBalance in the job.

14. From **End Period** drop-down, select the End Period for the integration job.

    By default, the $ENDPERIOD parameter uses the value of the variable parameters defined in global variables for the start period (see Editing Runtime Variables). However, you can select different end periods for different jobs. For example, if you load metadata in a Pipeline, you can set the End Period to be EndBalance in the job.

15. In **Number of Groups**, specify the maximum number of groups for the Group Split method (system splits the list of members in dimensions into multiple groups).

16. In **Use Fully Qualified Name**, select **Yes** to include the member name and the names of its ancestors to the level that uniquely defines the member.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Select **Yes** to show the fully qualified name. If the dimension includes Shared Members, then set it to **Yes** in the integration, then this parameter must be set to **Yes** since shared members must have unique parent members.

Select **No** to show the member name only. If you don't have shared members, then the fully qualified name is not required.

17. In **Label/Value**, specify the values for the Custom (Pattern) or Custom-Function split methods.

    Job type parameters for the Custom (Pattern) or Custom-Function split methods are added as Label and Value pairs (key value pairs) where **Label** is the name of an attribute, and **Value** is an assigned value for this attribute.

    To add a new Label/Value pair, click ⊕

    To delete a Label/Value pair, click ⊖

18. Click **Save**.

19. Run the integration with Smart Split job.

    For more information, see Running the Pipeline.

Integration with Smart Split job type parameters include:

Table 12-30 Integration with Smart Split job type parameters and descriptions

Integration with Smart Split job type parameters	Description
Name	Enter the name of the base integration to which to apply the split dimension.
Split Dimension	Specify the dimension in the source application on which to split the data loads. **Split Dimension on Period and Year Considerations:** 1. "Period" and "Year" are special dimensions and require special processing when used as a Split dimension. 2. When "Period " is used as the Split Dimension, only Single and Group are supported split methods. If other methods are used, an error occurs. 3. When "Year" is used as the Split Dimension, only Single is the supported split method. Running multiple years in a single integration run is not supported. If other methods are used, an error occurs. 4. For Standard Mode, the **Start Period** and **End Period** are used to determine the list of periods to split. Based on the "Period" range, the system determines the source periods and applies the source period filter. 5. For Quick Mode, source filters for the "Period" and "Year" are used to determine the period or year for each child integration and applies the source filter.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 12-30 (Cont.) Integration with Smart Split job type parameters and descriptions

Integration with Smart Split job type parameters	Description
Split Method	Select how members in the dimension are split (separated). The system creates a separate child integration (data rule) for split criteria in the dimension based on the split method. The child integration is identical to the base integration except for the source filter for split dimension. Child integrations are recreated every time the Pipeline is executed so do not modify the child integrations. The child integrations can be deleted using the Delete Integration system maintenance task. Available options: • **Single**—System runs a separate process for each member in the dimension. For example, you run a load per "Ledger" or "Entity." • **Group**—System splits the list of members in dimensions into multiple groups. The size of the group is based on the **Number of Groups** parameter. • **Custom** (Pattern)—You specify the values for the split dimension member. A custom pattern can be specified as list, range, wild-card in the **Label/Value** parameter. For example, you might specify a following custom pattern using these expressions: `Like - US*`: All members starting with US `Between 100-199`: All members in the 100 range `List 100,200,300`: List of individual members If a member is picked up by earlier filter, then it is not included the later filters. • **Custom-Functions**—You specify the values for split dimension member using Member Functions. Custom-Functions can be specified as member functions in the **Label/Value** parameter below.
Import Mode	Select the import mode for the integration job. By default, the $IMPORTMODE parameter uses the value of the variable parameter defined in global variables for the import mode (see Editing Runtime Variables). However, you can select different import modes for different jobs. For example, to load metadata in one integration and just data in another, specify different import modes between the two jobs.
Export Mode	Select the export mode for the integration job. By default, the $EXPORTMODE parameter uses the value of the variable parameter defined in global variables for the export mode (see Editing Runtime Variables). However, you can select different export modes for different jobs. For example, to load metadata in one integration and just data in another, specify different export modes between the two jobs.

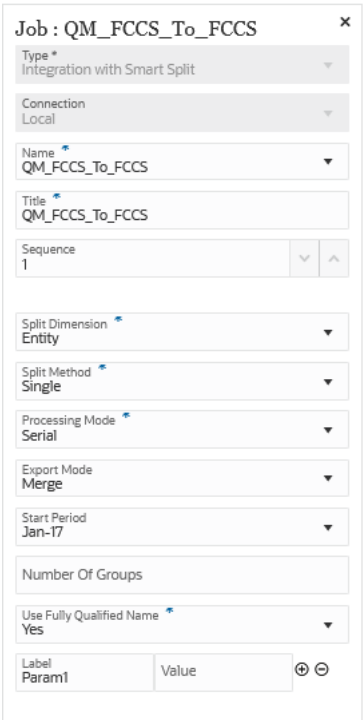
Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 12-30 (Cont.) Integration with Smart Split job type parameters and descriptions

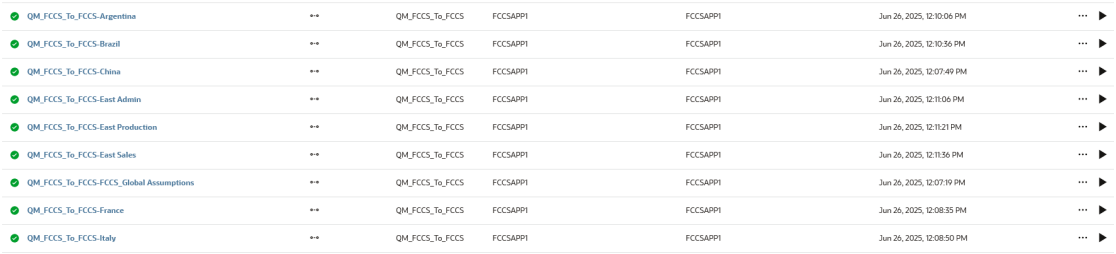
Integration with Smart Split job type parameters	Description
Start Period	**Optional** From the **Start Period** drop-down, select the Start Period for the integration job. By default, the $STARTPERIOD parameter uses the value of the variable parameters defined in global variables for the start period (see Editing Runtime Variables). However, you can select different start periods for different jobs. For example, if you load metadata in a Pipeline, you can set the Start Period to be BegBalance in the job.
End Period	**Optional** From the **End Period** drop-down, select the End Period for the integration job. By default, the $ENDPERIOD parameter uses the value of the variable parameters defined in global variables for the start period (see Editing Runtime Variables). However, you can select different end periods for different jobs. For example, if you load metadata in a Pipeline, you can set the End Period to be EndBalance in the job.
Number of Groups	Specify the maximum number of groups for the Group Split method.
Use Fully Qualified Name	Select **Yes** to include the member name and the names of its ancestors to the level that uniquely defines the member. If the dimension includes Shared Members, then set it to **Yes** in the integration, then this parameter must be set to **Yes** since shared members must have unique parent members. Select **No** to show the member name only.
Label/Value	Job type parameters are added as Label and Value pairs (key value pairs) where **Label** is the name of an attribute, and **Value** is an assigned value for this attribute. To add a new Label/Value pair, click ⊕ To delete a Label/Value pair, click ⊖

Here are sample parameters for an Integration with Smart Split job type where the Split Dimension is "Entity" and the Split Method is "Single":

Copyright © 2017, 2026, Oracle and/or its affiliates.

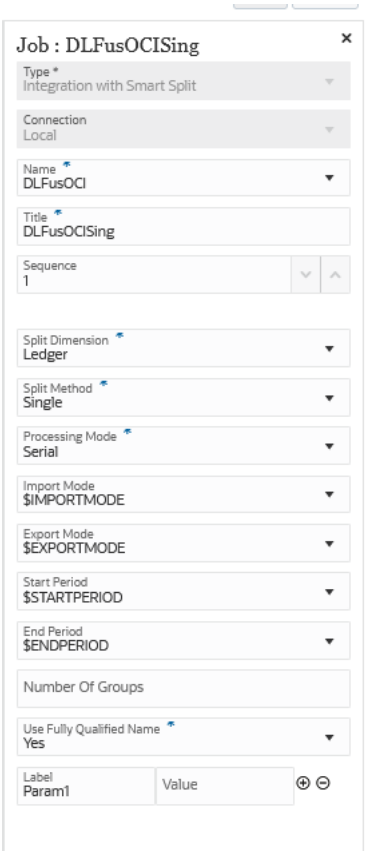


Here are the integrations created with Split Dimension on "Entity" and a Split Method of "Single":



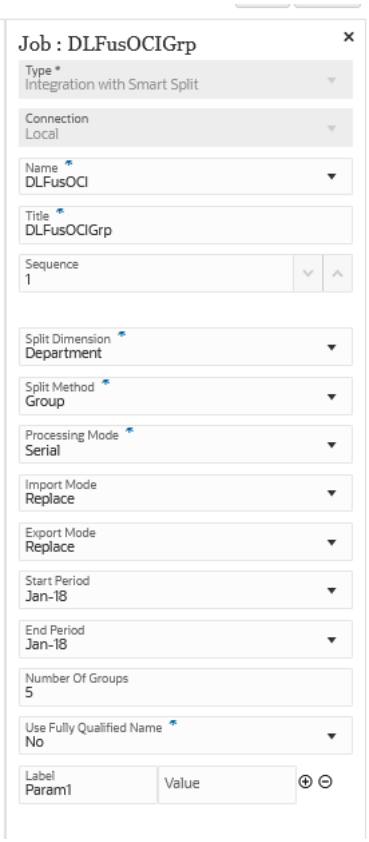
Here are sample parameters for an Integration with Smart Split job type where the Split Dimension is "Ledger" and the Split Method is "Single":

Copyright © 2017, 2026, Oracle and/or its affiliates.

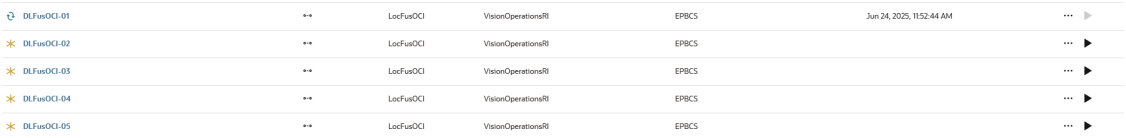


Here are sample parameters for an Integration with Smart Split job type where the Split Dimension is "Department" and the Split Method is "Group" with a maximum number of 5 groups:

Copyright © 2017, 2026, Oracle and/or its affiliates.

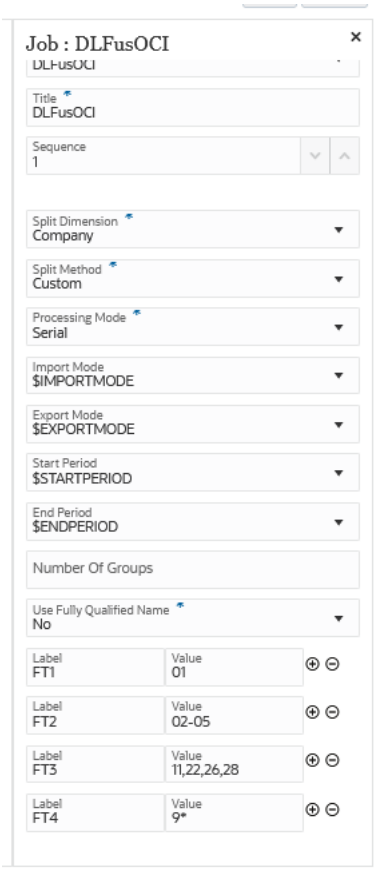


Here are the integrations created by the Integration with Smart Split job type above:

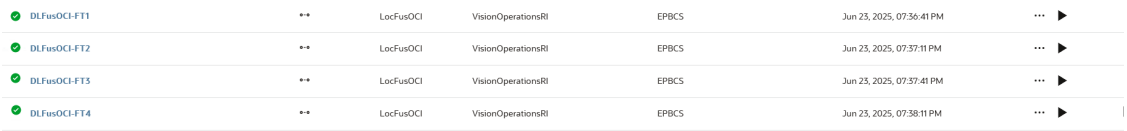


Here are sample parameters for an Integration with Smart Split job type where the Split Dimension is "Company" and the Split Method is "Custom":

Copyright © 2017, 2026, Oracle and/or its affiliates.

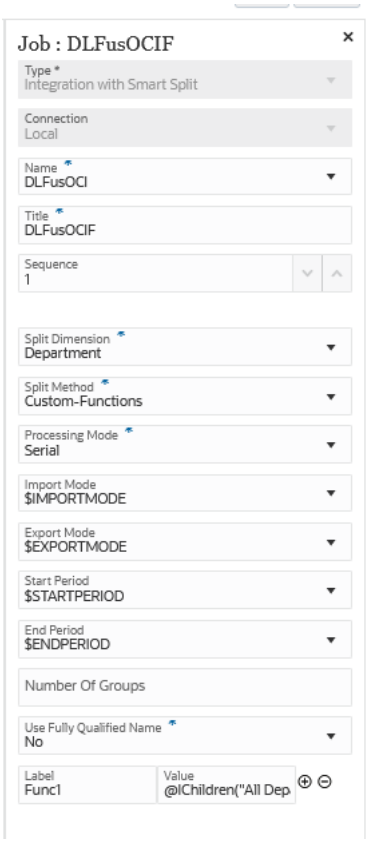


The DLFusOCI-FT1 - DLFusOCI-FT4 integrations below are the results of the Integration Smart Split job type using a Split Method of "Custom":



Here are sample parameters for an Integration with Smart Split job type where the Split Dimension is "Department" and the Split Method is "Custom-Functions":

Copyright © 2017, 2026, Oracle and/or its affiliates.



The DLFusOCI-Func1 integration below is the result of the Integration with Smart Split job type using a Split Method of "Custom-Function":

Using Open Batch Job Types

You can use Open Batch job types to add the following types of open batches to the Pipeline:

- Using an Open Batch - File Job Type
- Using an Open Batch - Location Job Type
- Using an Open Batch - Name Job Type

Open batches enable you to:

- Combine one or more load integrations in a batch and execute it at one time.
- Run jobs in a batch in serial or parallel mode.
- Define the parameters of the batch.

> **Note**
>
> It is recommended when processing batches in the Pipeline, review, and if necessary, reset the maximum time a batch can run before timing out. If a batch job exceeds the maximum time a batch can run, the job in the Pipeline moves on to the next job and the Pipeline email shows the Pipeline has failed even though no jobs in the job console show up as failed. You can reset the maximum time in the **Batch Timeout in Minutes** option in System Settings. For more information, see System Settings.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> You can migrate any batches in Data Management to the Pipeline, using the "Migrate Batches to Pipeline" script from **System Maintenance Tasks** in Data Management.

Using an Open Batch - File Job Type

Use the Open Batch - File job type to load and run multiple files to the Oracle Fusion Cloud Enterprise Performance Management in a batch. The file name format specifies the POV to load the data.

The Open Batch - File job type parameters include:

Table 12-31 Open Batch - File Job Parameters

Open Batch - File Job Type Parameters	Description
Open Batch Directory	Specify the folder under `inbox/batches/openbatch` subdirectory where the files to be imported are copied. If this field is blank or null, all files under `inbox/batches/openbatch` are processed.
File Name Separator	Select the character to use when separating the five segments of a batch file name. Options include: • ~ • @ • _ • ;
Execution Mode	In **Execution Mode**, select the • Serial—The serial execution mode processes files sequentially, requiring that one file complete its process before the next file starts its process. • Parallel—The parallel execution mode processes files simultaneously.
Number of Parallel Jobs	Specify the maximum number of Open Batch - File jobs that can be processed in parallel execution mode.
Auto Create Integration	To create the integration automatically for file-based integration, select **Yes**. When Data Integration assigns the integration name, it checks whether an integration with the name "Location_Category" exists. If this name does not exist, Data Integration creates the integration using the following file naming conventions: • Integration Name—Location_Category • Description—"Auto created data rule" • Category—Category • File Name—Null • Mode—Replace
Plan Type	Select the plan type of the target system. You can also register a custom cube as a separate application with no limit to the number of custom cube applications that can be registered.

Copyright © 2017, 2026, Oracle and/or its affiliates.

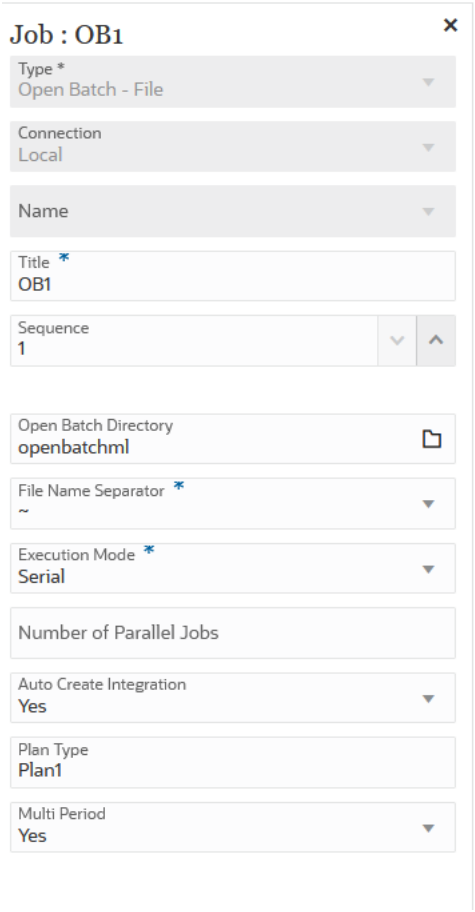
Table 12-31 (Cont.) Open Batch - File Job Parameters

Open Batch - File Job Type Parameters	Description

Multi Period	Select **Yes** to read file-based data sources with multiple periods and import them into the appropriate POV based on the file name.
	The names of multiple period batch files consist of the following segments in the following order:
	• File ID—A free-form field used to control load order. Batch files load in alphabetic order by file name.
	• Location
	• Category
	• Start Period
	• End Period
	• Load Method—A two-character item (Character 1 = append or replace, and Character 2 = target append or replace). Valid values are **A** and **R**.
	Examples of open batch for a multiple period file name:
	`a_Texas_Actual_ Jan-2023_ Jun-2023_RR.txt (Loc, Cat, Start Period, End Period)`
	and
	`b_TexasDR1_ Jan-2023_ Jun-2023_RR.txt (Data Rule, Start Period, End Period)`
	Select **No** to not include file-based data source with multiple periods and import them into the appropriate POV based on the file name.

Here are sample job parameters for an Open Batch - File job type:

Copyright © 2017, 2026, Oracle and/or its affiliates.



Using an Open Batch - Location Job Type

Use the Open Batch - Location Job type to add and run all integration jobs assigned to a specific location.

> **Note**
>
> When Data Integration assigns the integration name, it checks whether an integration with the name "Location_Category" exists. If this name does not exist, Data Integration creates the integration using the following file naming conventions:

- Integration Name—Location_Category
- Description—"Auto created integration"
- Category—Category
- File Name—Null
- Mode—Replace

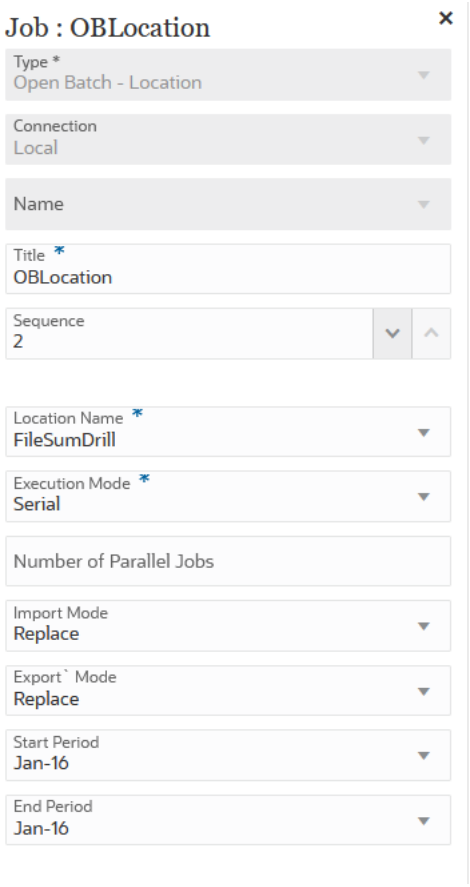
The Open Batch - Location job type parameters include:

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 12-32 Open Batch - Location Job Type Parameters

Open Batch - Location Job Type Parameters	Description
Location Name	In **Location Name** drop-down, select the location of the jobs to include in the Open Batch - Location.
Execution Mode	In **Execution Mode**, select the • Serial—The serial execution mode processes files sequentially, requiring that one file complete its process before the next file starts its process. • Parallel—The parallel execution mode processes files simultaneously.
Number of Parallel Jobs	Specify the maximum number of Open Batch - Location jobs that can be processed in parallel execution mode.
Import Mode	From the **Import Mode** drop-down, select the import mode for the Open Batch - Location job.
Export Mode	From the **Export Mode** drop-down, select the export mode for the Open Batch - Location job.
Start Period	From the **Start Period** drop-down, select the Start Period for the Open Batch - Location job.
End Period	From the **End Period** drop-down, select the End Period for the Open Batch - Location job.

Here are sample job parameters for Open Batch - Location job type:



Copyright © 2017, 2026, Oracle and/or its affiliates.

Using an Open Batch - Name Job Type

Use the Open Batch - Name job type to add and run all integration jobs that match an integration name with masking (wildcards * and ?) characters.

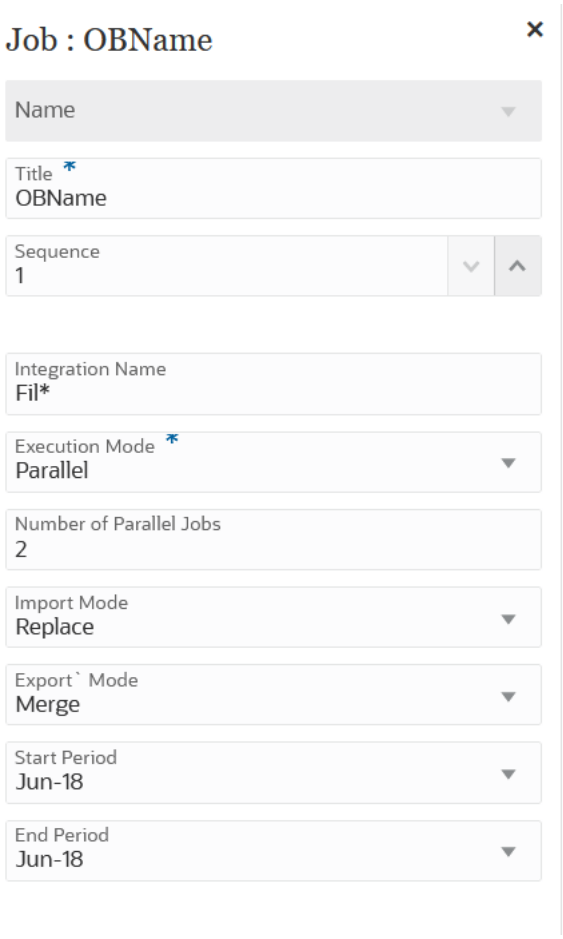
The Open Batch - Name job type parameters include:

Table 12-33 Open Batch - Name Job Type Parameters

Open Batch - Name Job Type Parameters	Description
Integration Name	Specify the integration name with masking characters that matches the integration jobs to run. The ? wildcard runs an integration name that matches any one character. The * character runs an integration name that matches any sequence or string of characters. For example, if you specify an integration name DL?ABC*, the system runs all integrations named DL_ABC123, DL-ABC456666, etc.
Execution Mode	In **Execution Mode**, select the • Serial—The serial execution mode processes files sequentially, requiring that one file complete its process before the next file starts its process. • Parallel—The parallel execution mode processes files simultaneously.
Number of Parallel Jobs	Specify the maximum number of Open Batch - Name jobs that can be processed in parallel execution mode.
Import Mode	From the **Import Mode** drop-down, select the import mode for the Open Batch - Name job.
Export Mode	From the **Export Mode** drop-down, select the export mode for the Open Batch - Name job.
Start Period	From the **Start Period** drop-down, select the Start Period for the Open Batch - Name job.
End Period	From the **End Period** drop-down, select the End Period for the Open Batch - Name job.

Here are sample job parameters for an Open Batch - Name job type:

Copyright © 2017, 2026, Oracle and/or its affiliates.



Using a Plan Type Map Job Type

Use a Plan Type Map job type to copy data from a block/aggregate storage cube to an aggregate storage cube or from a block/aggregate storage cube to another block/aggregate storage cube based on the settings specified in a Planning job of the type plan type map.

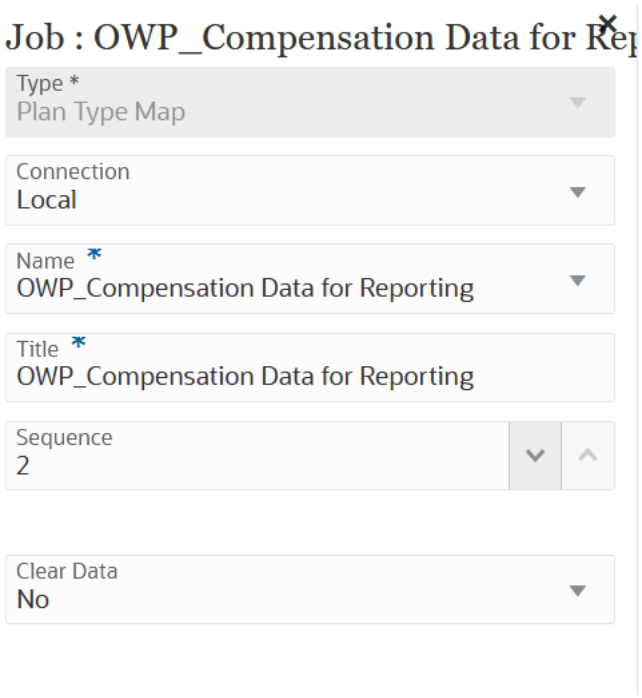
Plan Type Map job type parameters include:

Table 12-34 Plan Type Map Job Type Parameters and Descriptions

Plan Type Map Job Type Parameters	Description
Name	Select the job name of the type: `plan type map` defined in the application.
Clear Data	Specify whether the data in the target database should be removed before copying data. Select **Yes** to remove the data before copying the data. Select **No** to keep the data before copying the data

Here are sample job parameters for a Plan Type Map job type:

Copyright © 2017, 2026, Oracle and/or its affiliates.



Using a Run Auto Alert Job Type

Use the Run Auto Alert job type to run the auto alert process for a specified match type in Transaction Matching. This job type processes the alert rules defined for the match type and then automatically creates alerts for unmatched transactions. You can monitor the status of an Auto Alert job through the Job History tab in Account Reconciliation.

Run Auto Alert job type parameters and descriptions:

> **Note**
>
> For parameters that are common to all job types, see Pipeline Process Description.

Table 12-35 Run Auto Alert Job Type Parameters and Descriptions

Run Auto Alert Job Type Parameters	Description
Match Type	Specify the text ID of the match type (previously known as Reconciliation type) for which automated alerts must be generated. If the Match Type ID is changed in Account Reconciliation after Account Reconciliation pipelines have been created in Data Integration, you must re-select the Match Type LOV option in the pipelines and save the changes. This step is necessary to ensure the pipelines run successfully.

Using a Run Auto Match Job Type

Use the Run Auto Match job type to run the auto match process, which matches transactions using the rules defined by a Service Administrator in Transaction Matching.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> Run this job type after you import transactions data into Transaction Matching.

Run Auto Match Job Type parameters and descriptions:

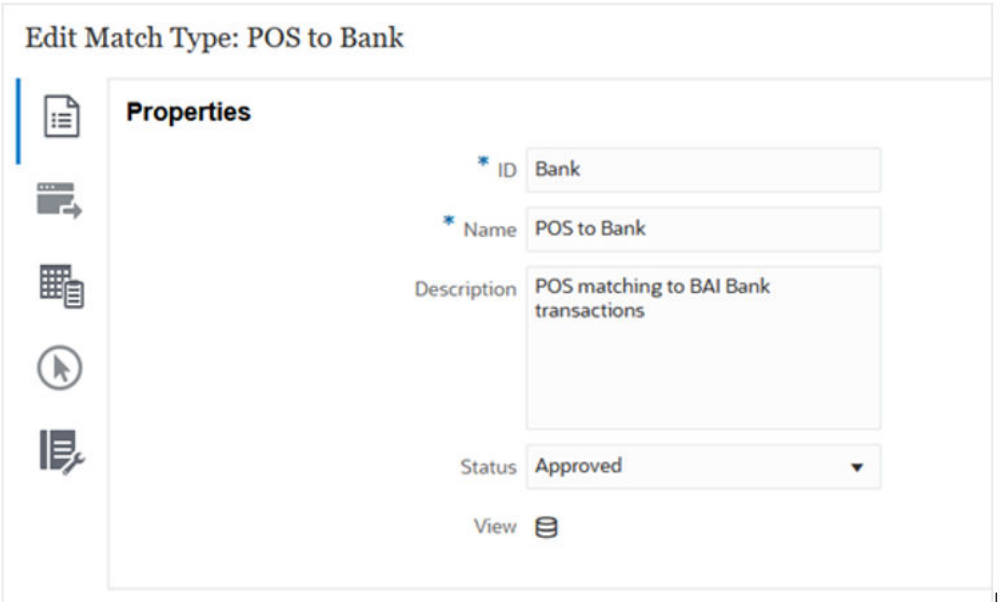
> **Note**
>
> For parameters that are common to all job types, see Pipeline Process Description.

Table 12-36 Run Auto Match Job Type Parameters and Descriptions

Run Auto Match Job Type Parameters	Description
Match Type	Select the ID of the Match Type for which automated alerts must be generated. The Match Type may contain up to 25 alphanumeric characters and cannot include any special characters except a period (.), dash (-), or underscore (_). If the Match Type ID is changed in Account Reconciliation after Account Reconciliation pipelines have been created in Data Integration, you must re-select the Match Type LOV option in the pipelines and save the changes. This step is necessary to ensure the pipelines run successfully.

Customers can view the Match ID for "Bank" on the Edit Match Type page

Figure 12-1 Image shows the Edit Match Type page.



Copyright © 2017, 2026, Oracle and/or its affiliates.

Using a Set Period Status Job Type

Sets the status for a period (open, closed, or locked) and returns the success or failure status.

When a period's status is changed to **Open**, the returned job corresponds to the opening of reconciliations for the specified period. The job's success or failure does not impact the period's status because this change is made immediately. Even if there are failures while reopening reconciliations, the period status still remains **Open**.

If a period's status is set to Closed or Locked, no job is returned.

Set Period Status job type parameters and description:

> **Note**
>
> For parameters that are common to all job types, see Pipeline Process Description.

Table 12-37 Set Period Status Job Type Parameters

Set Period Status Job Type Parameters	Descriptions
Period	Select the name of the period to which to set the period status, such as July**2026**.
Status	Select the status to be changed. Available options: • open • closed • locked

Using a Set Substitution Variable Job Type

Substitution variables act as global placeholders for information that changes regularly. For example, you can set the current month member to the substitution variable `CurMnth` so that when the month changes, you need not update the month value manually in the job. You create and assign values to substitution variables within the application. These substitution variables are then available in the application when you select members for a form.

For more information, see About Substitution Variables in *Administering Planning*.

Set Substitution Variable job type parameters include:

Table 12-38 Set Substitution Variable Job Type Parameters

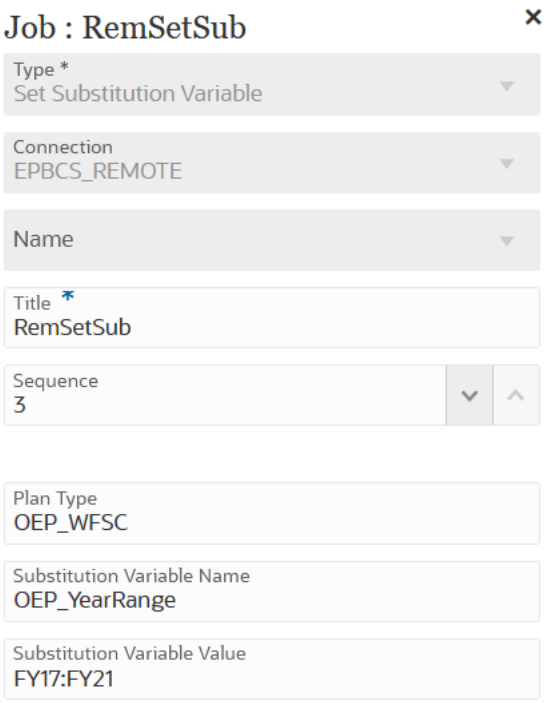
Set Substitution Variable Job Type Parameters	Description
Plan Type	Enter the plan type (cube) of the target system. If you are using "All Cubes" as the plan type, leave the Plan Type (cube name) blank.
Substitution Variable Name	Enter the name of the substitution variable. You can enter up to 80 characters for the name.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 12-38 (Cont.) Set Substitution Variable Job Type Parameters

Set Substitution Variable Job Type Parameters	Description
Substitution Variable Value	Enter a value for the substitution variable (up to 255 characters). You can specify a range of values when defining substitution variables; for example, `FY16:FY18`. You can define a substitution variable range for base members and parent members. For base members, you can use a single colon (`:`) or a double colon (`::`) as delimiters; for example, `SunEnT 110::111`. However, for parent members, you can only use a single colon (`:`). If the substitution variable is to be used in a data export job definitions for block storage cubes and the member name is numeric only, the member name must be placed in double quotation marks in the **Value** field; for example, `"1100"` or `"000"`. The double quotation marks aren't necessary for alphanumeric member names; for example, `a1110`. However, if this variable is to be selected in a data export job definitions for aggregate storage cubes, you can't place the member name in double quotation marks in the **Value** field. To use the same member name in substitution variables for data export jobs defined for both types of cubes, define two different substitution variables; one with quotes and one without. A substitution variable used as a parameter value in Substitution Variable Value must be defined for all cubes.

Here are sample job parameters for a Set Substitution Variables job type:



Copyright © 2017, 2026, Oracle and/or its affiliates.

Using a Wait Job Type

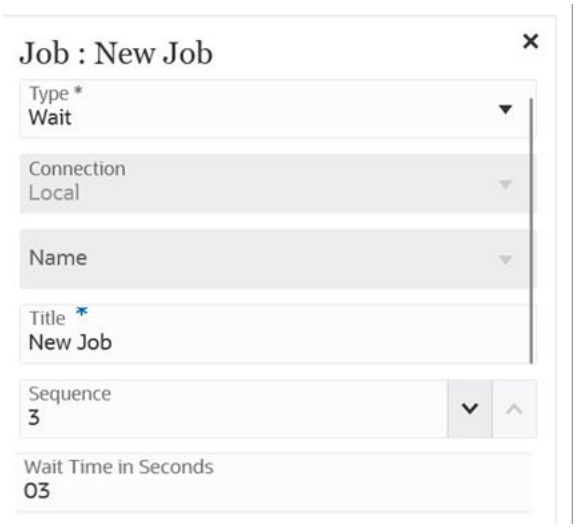
Use a controlled delay within a pipeline job. When multiple bursts are executed simultaneously, they can create a backlog during post-processing. To mitigate this, you can add a timed wait step (up to 300 seconds) within the job that pauses execution for a specified duration before resuming, thereby smoothing workload distribution and preventing processing bottlenecks.

> **Note**
>
> For parameters that are common to all job types, see Pipeline Process Description

Table 12-39 Table Wait Job Type Parameters

Wait Job Type Parameters	Description
Title	Provide a title for the job.
Sequence	Specify a number to define the chronological order in which a stage is executed.
Wait Time in seconds	Specify the wait duration (in seconds). Maximum wait is 300 seconds.

Here's a sample job parameter for Wait job type:



Editing Runtime Variables

Shows how to select a set of variables that can be used as runtime parameter for Pipeline jobs.

On the Variables page, you can select a set of variables that can be used as runtime parameter for Pipeline jobs. When you execute a Pipeline, you are provided an input value for these variables - a set of out-of-box or runtime values, from which parameter values can be used for individual jobs. Variables can be pre-defined types, such as: "Period," "Import Mode," or "Export Mode."

You can also add custom variable parameter as a job parameter. For example, if you want to pass static values using a substitution variable for the Start Period, change the Variable type to

Copyright © 2017, 2026, Oracle and/or its affiliates.

"Text" and then specify a substitution variable as a parameter (accepted by the integration RESI API), such as `&ThisMonth#&ThisYear`. (A substitution variable is a user variable name preceded by one or two ampersands (&)).

The default runtime variables include:

- Start Period
- End Period
- Import Mode
- Export Mode
- Send Mail
- Send To
- Attach Log
- Stage Execution

To modify Pipeline runtime variables:

1. From the Data Integration home page, click ••• to the right of the Pipeline, and then select **Pipeline Details**.
2. From the **Pipeline** page, click .
3. Click the **Variables** tab.
4. Select the variable to modify.
5. In **Variable Name**, to add or modify a predefined variable name, specify the name.

   The variable name can be referenced as a job parameter. The name can be no longer than 20 characters and can contain no spaces.
6. In **Display Name**, specify the display prompt name for the variable.
7. In **Display Order**, specify the display order of the variable.
8. Check **Required** if a value for variable is required to execute the Pipeline.
9. From **Validation Type** drop-down, select the type of value used to validate the variable.

   Validation types include:

   - List—Use a LOV query and bind values are required.
   - Text—Free form text field
   - File—File name. A file browser window is available to select a file.
   - Custom List—User defined list of values

     To use this validation type, provide a list of values in the Validation Parameter window. The values must be specified in a `Value,Label` format and every line must end with a new line character, for example,

     ```
     M,Month
     Q,Quarter
     Y,Year
     ```

   - Lookup—Use of Lookup values available.
10. From **Validation Parameters** drop-down, select the predefined parameters for the variable based on the validation type.

Copyright © 2017, 2026, Oracle and/or its affiliates.

For example, you could select a Start Period variable parameter based on a list validation type such as:

- Pipeline Status
- Pipeline Export Mode LOV Query
- Global Period Name LOV Query
- Location No Security LOV Query
- Trial Balance Converted by Target Entity/Account Query
- Process Monitor Query

11. From **Default Value**, select the default value for the variable.

    The default value can be a static value or a Planning substitution variable.

    The default runtime variables and their descriptions are provided in the following table.

Table 12-40 Default Runtime Variables and Descriptions

Variable Name	Available Values
Start Period	Specify the first time period included in the pipeline's import/export processing scope. It sets the beginning of the period range used by the Data Exchange step (often together with **End Period**) to determine which periods' data to process.
End Period	Specify the last time period included in the pipeline's import/export processing scope. It sets the end of the period range (used with **Start Period**) to define which periods' data the Data Exchange step will process.
Import Mode	Specifies how the Data Exchange pipeline should handle loading the incoming data during an import step, that is, whether the import replaces existing data or adds to/updates the existing data in the target. • **Replace**: (Default value) Replaces existing target data for the import scope, then loads the new data. • **Append**: Retains existing target data and adds new records and/or updates matching records based on the import rules. • **No Import**: Doesn't load data. • **Map and Validate**: Runs the import using the defined mappings and validation rules to transform and validate the source data, but doesn't write the data to the target. It's used to confirm the file/format, mapping coverage, and rule validations and to review any errors or warnings before performing a load/commit mode.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 12-40 (Cont.) Default Runtime Variables and Descriptions

Variable Name	Available Values
Export Mode	Specifies how the Data Exchange pipeline should export the data: • **Merge**: (Default value) Exports only the data in the current run scope and combines it with any existing export output for the target file/location. Existing records may be updated where keys match; new records are added. • **Accumulate**: Adds the current run's export output to the existing output as an additional set of records (append-only), preserving prior exported content. • **Replace**: Generates a new export output for the current run scope and overwrites the existing export file/content in the target location. • **Dry Run**: Executes the export process (scope selection, formatting, validations) without producing or writing the final export output, allowing you to verify results and detect issues before generating files. • **No Export**: Doesn't export the data.
Send Mail	Controls whether the pipeline sends an email notification for the pipeline run: • **Always**: Sends an email notification for every pipeline run, regardless of outcome (success, warning, or failure). • **No**: (Default value) Doesn't send any email notification for the pipeline run. • **On Failure**: Sends an email notification only when the pipeline run fails (or ends in an error status). • **On Success**: Sends an email notification only when the pipeline run completes successfully.
Send To	Specify the email recipient list for pipeline run notifications when Send Mail is enabled.
Attach Logs	Controls whether the pipeline run's execution logs are included as file attachments with the pipeline results/notifications (for example, in the job output or any configured email/notification payload). • **No**: (Default value) Doesn't attach logs • **Yes**: Attaches logs
Stage Execution	• **Start from beginning**: (Default value) When Pipeline fails, restart the process from the beginning. • **Start from failure**: When Pipeline fails, instead of restarting the process, it starts the job from the last failed stage. When a pipeline is restarted from a failed stage, the status of the jobs from the previous stage are carried over to the new Pipeline execution. **Note** If you select **Start from failure**, but the pipeline was successful, the pipeline won't run.

12. Click **Save**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Adding Custom Variable Parameters

When working with global variables for a Pipeline, you can add a custom variable to use as an input for jobs.

1. From the Data Integration home page, click ••• to the right of the Pipeline, and then select **Pipeline Details**.
2. From the **Pipeline** page, click .
3. Click the **Variables** tab.
4. Select the variable to which to add as a custom list as parameter.
5. From **Validation Type**, select the type of validation, such as file or custom list.
6. From **Validation Parameters**, specify the parameters to use for the job custom variable.

   For example, say you want to pass a Scenario name to a business rule and you want to accept the Scenario Name as a variable. To do this, from **Validation Type**, select **Custom List**. Then in **Validation Parameters**, click , specify the parameters in the **Edit Custom List**, and then click **OK**.

   ```
   Actual,Actual
   Plan,Plan
   Forecast,Forecast
   ```

Adding a Substitution Variable as an Input Parameter

You can use a substitution variable as an input value in Pipeline Variables. You can use existing substitution variables or create new substitution variables. Substitution variables are variables that are defined at the Oracle Fusion Cloud Enterprise Performance Management or Oracle Essbase data source that serves as placeholder for specific members that change regularly.

Some considerations when working with substitution variables:

- The substitution variable must be defined for all cubes to be used as an input value.
- For an existing substitution variable, change the **Validation Type** to **Text**. For a new substitution variable, assign the **Validation Type** as **Text**.
- Use of a substitution variable in the `&Mon#&Year` format is supported for integration jobs only when the target is a Cloud EPM application (for example, Planning, Financial Consolidation and Close, and FreeForm).

  Using a `&Mon#&Year` format is not supported in an integration job with a Data Export to File target application. Instead of using the `&Mon#&Year` format, define a variable, for example `&StPeriod` for the start period and set the value of the period name in Data Integration, such as "Jan-24." In this case, the Pipeline resolves the substitution variable and passes it as input to the integration job.

To add a new Pipeline variable and set the input value for the pipeline variable as a substitution variable:

1. From the Data Integration home page, click to the right of the Pipeline, and then select **Pipeline Details**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

2. From the **Pipeline** page, click .
3. Click the **Variables** tab.
4. To add a new substitution variable as an input variable, click

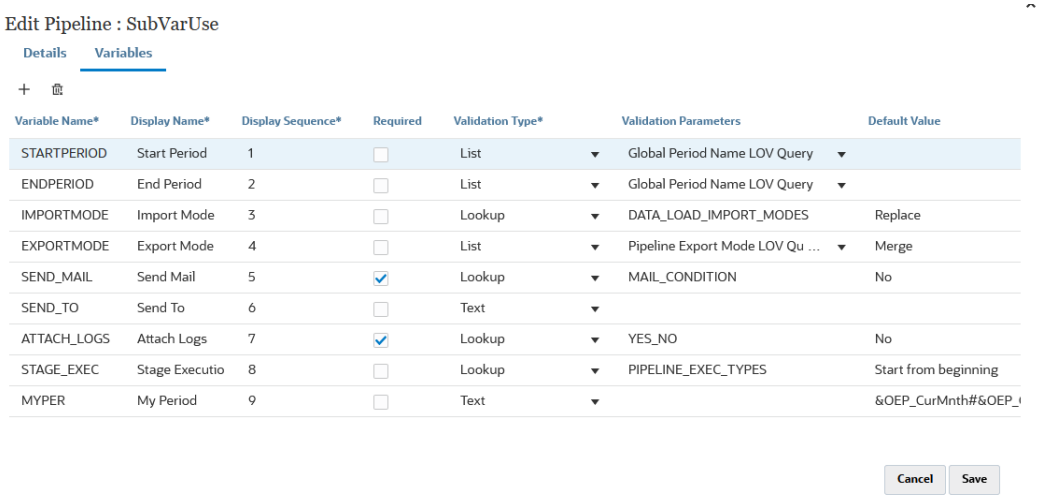
   A blank line of variable entry fields is displayed.
5. Scroll to the bottom of the variable list and add the following values for the substitution variable:
   - In **Variable Name**, specify the substitution variable name.

     The variable name can be referenced as a job parameter. The name can be no longer than 20 characters and can contain no spaces.
   - In **Display Name**, specify the display prompt name for the Pipeline variable.
   - In **Display Order**, specify the display order of the substitution variable.
   - Check **Required** if a value for the substitution variable is required to execute the Pipeline.
   - From **Validation Type** drop-down, select **Text**.
6. From **Default Value**, enter: ***&<Substitution Variable Name>*** as the default value (up to 255 characters).

   You can use the substitution variable *CurrMonth* to represent the current month (Jan), and you can insert it as a member selection in a report. Create variables at the data source by creating the variable and the string value. You can then change the value of the variable at the data source any time.

   Substitution Variables are prefixed with an "&" (for example, *&CurrMonth*).

   In the following example, the substitution variable "MYPER" shows *&CurMonth* and *&CurYr* values.



7. Click **Save**.

Running the Pipeline

Shows how to execute the series of jobs in the stages of the Pipeline, perform integration processes, send emails, and attach logs.

Copyright © 2017, 2026, Oracle and/or its affiliates.

You run the Pipeline to execute the series of jobs in the stages of the Pipeline, perform integration processes, send emails, and attach logs. During the run process, stages are executed serially and jobs assigned to each stage are run in the order in which the job has been sequenced either serially or in parallel.

When the Pipeline is running, the system shows the status on each stage card. You can also see the status of the Pipeline running in Process Details. Each job included in the Pipeline is submitted separately and creates a separate job log in Process Details (for more information, see Viewing Process Details).

To run a Pipeline:

1. From the **Data Integration** home page, select a Pipeline, and then click ▶.

   Each Pipeline is identified with a on the **Data Integration** home page. You can run a Pipeline by selecting it and then clicking ▶.

   You can also click ▶ on the Pipeline page.
2. On the **Run Pipeline** page, select any variables and then click **Run**.

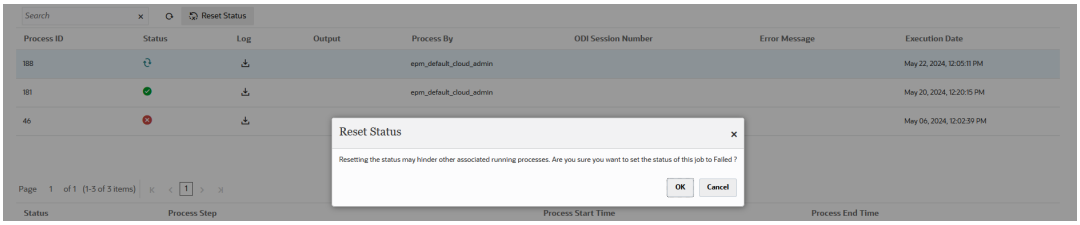
   Pipeline job status include:

Table 12-41 Pipeline Job Statuses

Job Status	Description
✓	When a job has been successfully executed, a ✓ a check mark appears on the job card. To download the log file for a job, select the job from the stage card and click ✓. You can check the status of the Pipeline from the Pipeline or on the Pipeline page. In the Pipeline, the status is displayed on each job card. This provides a overall snapshot view of the Pipeline execution status. You can click on the status icon to download the log file for each job. No logs are available for a stage.
✗	When a job has failed, a ✗ is displayed. In cases where a Pipeline has failed at a particular stage, you can restart the process at the step that failed instead of rerunning the entire Pipeline. This is helpful when a Pipeline includes many steps that took a long time to process, and you do not want to rerun the entire pipeline from the start, but restart from the error.
⚠	When a job has processed with warnings, a ⚠ shows.

If you need to stop an incorrect execution of a Pipeline job without restarting the instance, you can reset the status of a Pipeline to a "Failed" status. To do this, go to **Process Details** for the job and click **Reset Status**. When prompted, click **OK** on the **Reset Status** page.

Copyright © 2017, 2026, Oracle and/or its affiliates.



For more information about Process Details, see Viewing Process Details.

A separate job log in Process Details is generated each time a Pipeline is run as shown below.

```
2023-05-18 19:35:39,456 INFO  [AIF]: -------------------- Start Stage : Load Dimension Members --------------------
2023-05-18 19:35:39,456 INFO  [AIF]: Executing Stage : Stage 1, Mode : Serial
2023-05-18 19:35:39,457 INFO  [AIF]: Executing pipeline job:FusionEntity, Type:integration
2023-05-18 19:35:39,457 INFO  [AIF]: Job Parameters :
2023-05-18 19:35:39,459 INFO  [AIF]:  [jobType = integration, jobName = FusionEntity, jobID = 10886, jobSeq = 1, jobObject =FusionEntity, jobConnection =
LOCAL, parameters =[{paramLevel=SOURCE_FILTER, paramName=COA_APP_NAME, paramValue=null}, {paramLevel=SOURCE_FILTER, paramName=DIM_NAME, paramValue=null},
{paramLevel=SOURCE_FILTER, paramName=HIER_TOP_NODE, paramValue=null}, {paramLevel=SOURCE_FILTER, paramName=REFRESH_FROM_SOURCE, paramValue=null},
{paramLevel=JOB, paramName=endPeriod, paramValue=$ENDPERIOD}, {paramLevel=JOB, paramName=exportMode, paramValue=$EXPORTMODE}, {paramLevel=JOB,
paramName=importMode, paramValue=$IMPORTMODE}, {paramLevel=JOB, paramName=periodName, paramValue={$STARTPERIOD}{$ENDPERIOD}}, {paramLevel=JOB,
paramName=startPeriod, paramValue=$STARTPERIOD}]]
2023-05-18 19:35:39,461 INFO  [AIF]: Executing pipeline job:FusionAccount, Type:integration
2023-05-18 19:35:39,461 INFO  [AIF]: Job Parameters :
2023-05-18 19:35:39,462 INFO  [AIF]:  [jobType = integration, jobName = FusionAccount, jobID = 10887, jobSeq = 2, jobObject =FusionAccount, jobConnection =
LOCAL, parameters =[{paramLevel=SOURCE_FILTER, paramName=COA_APP_NAME, paramValue=null}, {paramLevel=SOURCE_FILTER, paramName=DIM_NAME, paramValue=null},
{paramLevel=SOURCE_FILTER, paramName=HIER_TOP_NODE, paramValue=null}, {paramLevel=SOURCE_FILTER, paramName=REFRESH_FROM_SOURCE, paramValue=null},
{paramLevel=JOB, paramName=endPeriod, paramValue=$ENDPERIOD}, {paramLevel=JOB, paramName=exportMode, paramValue=$EXPORTMODE}, {paramLevel=JOB,
paramName=importMode, paramValue=$IMPORTMODE}, {paramLevel=JOB, paramName=periodName, paramValue={$STARTPERIOD}{$ENDPERIOD}}, {paramLevel=JOB,
paramName=startPeriod, paramValue=$STARTPERIOD}]]
2023-05-18 19:35:39,466 INFO  [AIF]:
2023-05-18 19:35:40,105 INFO  [AIF]: Job Process ID  : 1708
2023-05-18 19:35:55,271 INFO  [AIF]:
2023-05-18 19:35:55,837 INFO  [AIF]: Job Process ID  : 1709
2023-05-18 19:36:10,995 INFO  [AIF]: -------------------- End Stage : Load Dimension Members --------------------
2023-05-18 19:36:10,998 INFO  [AIF]:
2023-05-18 19:36:10,998 INFO  [AIF]: -------------------- Start Stage : Copy Files --------------------
2023-05-18 19:36:10,998 INFO  [AIF]: Executing Stage : Stage 2, Mode : Serial
2023-05-18 19:36:10,998 INFO  [AIF]: Executing pipeline job:Copy File from Object Store, Type:copyFromObjectStore
2023-05-18 19:36:10,998 INFO  [AIF]: Job Parameters :
2023-05-18 19:36:11,000 INFO  [AIF]:  [jobType = copyFromObjectStore, jobName = Copy File from Object Store, jobID = 10880, jobSeq = 1, jobObject =null,
```

Sending Pipeline EMAIL Notifications

Shows how to send an email notification and attach a log based on your runtime variable definitions.

After running the Pipeline, you can send an email notification and attach the log based on your runtime variable definitions.

The **Send Mail** variable determines when an email is sent when a Pipeline is run. Options include:

- Always
- No—Default value
- On Failure
- On Success

The **Send To** variable determines the recipient email ID for the email notification. Email IDs are comma separated.

The **Attach Log** variable determines if Pipeline logs are zipped and included as an attachment in an email. A download link for the log is added automatically for each job.

Available options:

- Yes—Logs are zipped and included as an attachment to an email, which can then be download.
- No—Default value

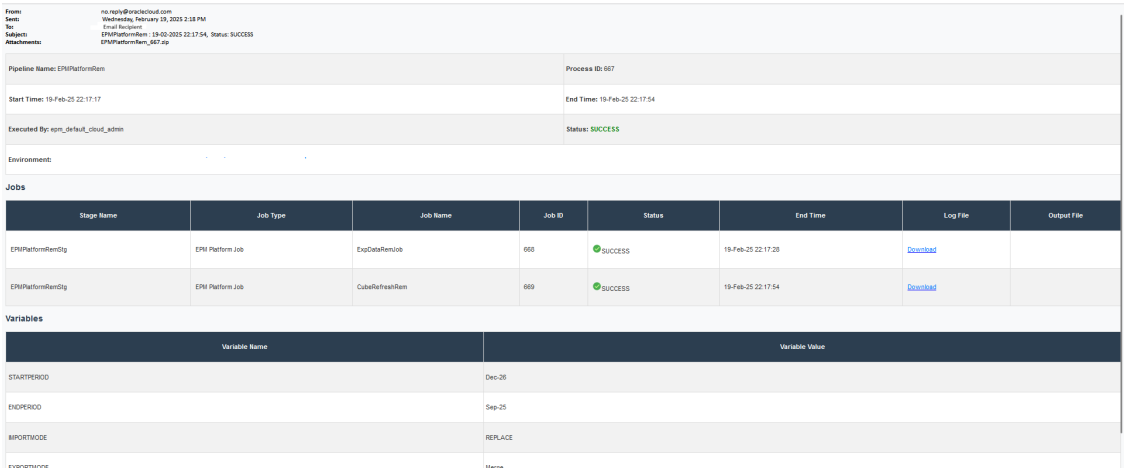
By default, an email includes a link to view the log file. When you click the link, you are prompted to login to the system and download the log file. If you are already logged in, the logs are downloaded automatically.

Copyright © 2017, 2026, Oracle and/or its affiliates.

For more information about the runtime variables, see Editing Runtime Variables.

The email sent to the recipient includes:

- Stage Name
- Job Type
- Job Name
- Job ID
- Status
- End Time
- Log File
- Output File
- Variable Name
- Variable Value



From: no.reply@oracleclouds.com
Sent: Wednesday, February 19, 2025 2:18 PM
To: Email Recipient
Subject: EPMPlatformRem : 19-02-2025 22:17:54, Status: SUCCESS
Attachments: EPMPlatformRem_667.zip

Pipeline Name: EPMPlatformRem | Process ID: 667
Start Time: 19-Feb-25 22:17:17 | End Time: 19-Feb-25 22:17:54
Executed By: epm_default_cloud_admin | Status: SUCCESS
Environment:

Jobs

Stage Name	Job Type	Job Name	Job ID	Status	End Time	Log File	Output File
EPMPlatformRemStg	EPM Platform Job	ExpDataRemJob	668	SUCCESS	19-Feb-25 22:17:28	Download	
EPMPlatformRemStg	EPM Platform Job	CubeRefreshRem	669	SUCCESS	19-Feb-25 22:17:54	Download	

Variables

Variable Name	Variable Value
STARTPERIOD	Dec-26
ENDPERIOD	Sep-25
IMPORTMODE	REPLACE

Allowing Non-Administrators to Execute Jobs in the Pipeline

Shows how to enable non-administrators to execute an individual Pipeline job type using proxy administrator user credentials.

As a service administration, you can allow non-administrators to execute an individual Pipeline job type using proxy administrator user credentials for which they have process ownership. Optionally, you can manage which Pipelines a non-administrator can see on the Data Integration home page by the location(s) assigned to them in Location Security.

If the user executing a job is a non-administrator and no proxy administrator user credentials are specified, the system executes all jobs as non-administrator users. In this case, only jobs that do not require a service administrator to execute them, are included in the jobs run in the Pipeline, for example, an integration job. If jobs require an service administrative to execute them and a non-administrator without a proxy administrator user credentials attempts to execute them, then the Pipeline process fails.

Location Security is an additional framework in which service administrators can secure access to the Pipeline. When Location Security is enabled, Data Integration evaluates the user groups by location to which a non-administrator has been assigned and determines the following view access:

Copyright © 2017, 2026, Oracle and/or its affiliates.

- When location security is enabled, non-administrators can view Pipeline jobs in ready-only mode (cannot edit or create a Pipeline) and run individual jobs in the Pipeline based only on the users groups by location to which they have been assigned in Location Security. (The service administrator can create, edit, and run any job type in the Pipeline.)

  For more information about location security, see Defining Location Security in *Administering Data Management for Oracle Enterprise Performance Management Cloud*.

- When location security is not enabled, non-administrators can view all Pipeline jobs in read-only mode (cannot edit or create a Pipeline) and run any individual job in the Pipeline. (The service administrator can create, edit, and run any job type in the Pipeline.)

To create a proxy server user credential:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Configure** actions drop-down, select **System Settings**.
2. From the **System Settings** page, then in the **Enable Pipeline Execution for Non-Admin**, select **Yes**.

   This setting enables non-administrator users to view Pipelines on the Data Integration home page.
3. Click **Save**.
4. From the Data Integration home page, click (Create), and then select **Pipeline**.
5. From the **Create Pipeline** page, complete the following:
   - **Pipeline Code**—specify the user-assigned code for the Pipeline.

     The code can contain up to 30 alphanumeric characters with a minimum of 3 characters and a maximum of 30 characters. This code cannot be updated after a Pipeline is created. Use the pipeline code to execute the Pipeline using a REST API.
   - **Maximum Parallel Jobs**—When jobs are run in parallel mode, specify the maximum number of jobs to run in parallel mode.

     When jobs are run in parallel mode, at runtime, the system runs jobs together in parallel (not sequentially). You can enter between 1 to 25 jobs to run in parallel mode.
6. In **Location**, specify a new location or select an location to associate with the Pipeline.

   > **Note**
   >
   > The name of the location selected for the Pipeline cannot be the same as a name of a location selected for an integration.

7. In **Proxy User**, enter the username with the service administrator role.

   If the username is different from the current username, then the system prompts the user to enter the password.

   For information on how roles are defined in the Oracle Fusion Cloud Enterprise Performance Management, see Access Control in *Administering Access Control* .

   If the **Proxy User** name is the same name as the service administrator, the system does not prompt for a password.
8. In **Proxy Password**, enter the password for the proxy user.
9. Click **Save and Continue**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

13

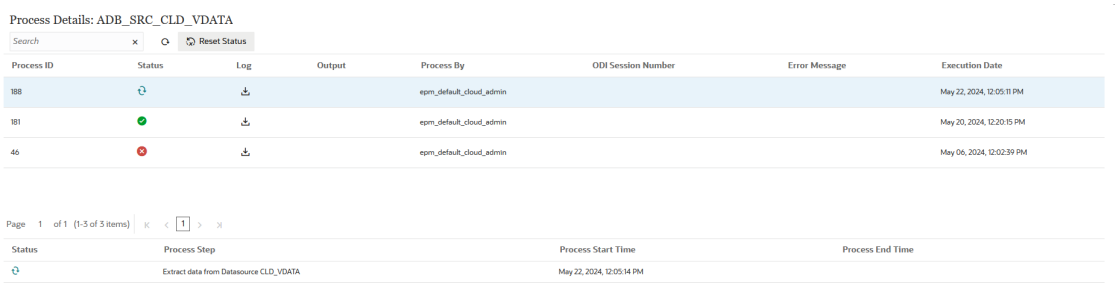
Reviewing Process Details and Viewing Results in the Workbench

You can review and validate your integrations at any time during the workflow process to help you evaluate the status and quality of the data to be loaded. Process Details and the Workbench provide a framework from which you can import, view, verify and export data from source systems.

Viewing Process Details

View the status of a submitted integration or Pipeline jobs and any associated logs.

Use Process Details to view the status and logs of a submitted integration or Pipeline. Additionally, you can reset the status of an executed job on this page.



To view process details:

1. From the **Data Integration** home page, then the **Actions** drop-down, select **Process Details**.

   To view process details for a specific integration, click *** to the right of the data integration or Pipeline, and then select **Process Details**.

2. Select the **Process ID** of the integration or Pipeline to view the status of the submitted job.

   The Process Details page shows the processes executed job. The following columns are displayed for each process:

   - **Process ID**—A system generated identification number for each submitted job
   - **(Status)**—Displays a visual indicator of the process status. You can rest the cursor over the icon to view a Screen Tip.

     Available statuses:

     – —Job is running.

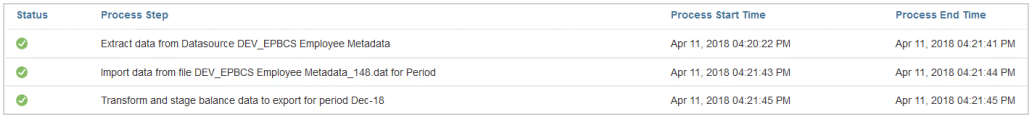
     —Job was processed successfully.

     – —Job did not complete successfully.

     – —Job was processed with warnings.

Copyright © 2017, 2026, Oracle and/or its affiliates.

- **Log**—Click to show the log for the submitted job. For each job, the log shows the status, process step, process start time, and process end time.



Status	Process Step	Process Start Time	Process End Time
✓	Extract data from Datasource DEV_EPBCS Employee Metadata	Apr 11, 2018 04:20:22 PM	Apr 11, 2018 04:21:41 PM
✓	Import data from file DEV_EPBCS Employee Metadata_148.dat for Period	Apr 11, 2018 04:21:43 PM	Apr 11, 2018 04:21:44 PM
✓	Transform and stage balance data to export for period Dec-18	Apr 11, 2018 04:21:45 PM	Apr 11, 2018 04:21:45 PM

- **Output File** —When an output file has been generated for the job, a is shown in this field.

  An output file is generated when the option to export the data file is enabled. The name of the data file is <target application name>_<process ID>.dat, and it is written to the <application root folder>/outbox directory. If you open the output file from the Process Details page, the output file is available in pdf format. You open or save the output file by clicking .

  > **Note**
  >
  > The system keeps Process log only for 7 days after which they are purged.
  >
  > If you want to download a Process log, click to download the log or you can use the downloadfile EPM Automate command to download log to a local folder.

  The output file may also contain a validation error report, which shows rejected data cells, intersections and the rejection reason when you load data. The validation error report option is enabled when the **Display Validation Failure Reasons** target option is enabled. See Defining Target Options.

  When a load fails during the run step, an error message output file is generated and made available in this field. For more information, see Viewing an Error Messages Output File.

- **Process By**—User ID of the person who submitted the integration.
- **ODI Session Number**—The session number in Oracle Data Integrator. You can use this to look up a session in Oracle Data Integrator.

  > **Note**
  >
  > The ODI Session number is present in Process Details only when the data is processed during an offline execution.

- Output
- **Error Message**—Shows the reason why a submitted integration failed.
- **Last Execution**—Shows the last date the submitted integration was run.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Viewing an Error Messages Output File

Shows how to capture any error message encountered during the load data steps in a separate output file, which contains only the error message.

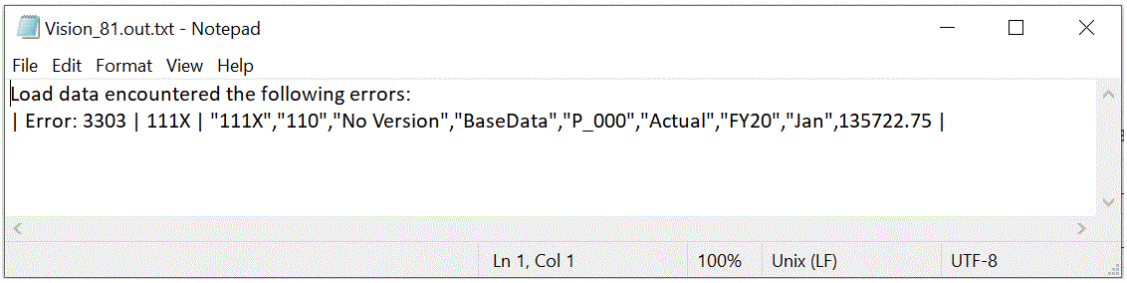
When exporting data to Planning, any error message encountered during the load data steps is captured in a separate output file, which contains only the error message. This information is a valuable debugging tool. You can easily identify any errors by simply reviewing the output files or download it and make it available to users by way of email.

> **Note**
>
> Error messages are also shown in the Process Log, but may not be available if the log level is set to "1." In addition, the Process log also shows statuses, process steps, and process times, which may make it harder to find and identify error messages.

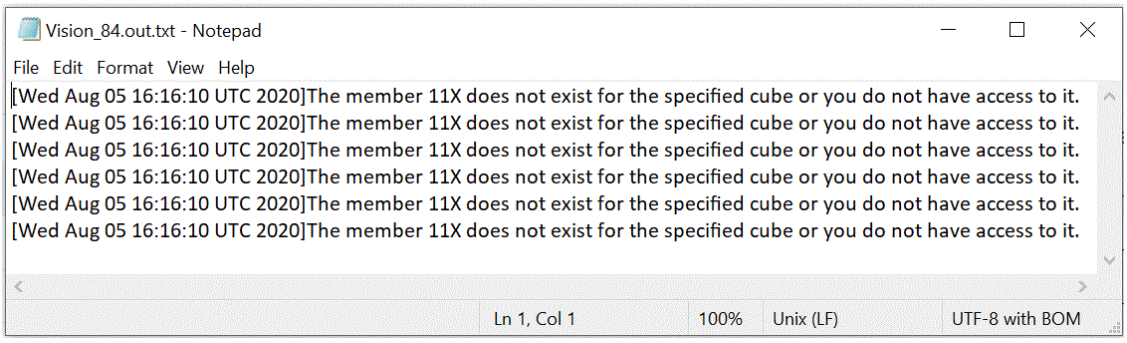
Error messages generated when loading data to Planning occur when one of three methods below is used.

- Quick Mode load to Essbase (used when Numeric Data Only is the load method)
- Outline Load Utility (used when All Data Type with Security - Admin Users is the load method)
- REST API used when All Data Type with Security (when Security is enabled for the administrator)

In the following example, the error message is generated when a Quick Mode load made to Oracle Essbase has failed:



In this example, the error message identifies an invalid member number that was incurred when the load was made from the Outline Load Utility.



To view the error messages output file for an integration:

Copyright © 2017, 2026, Oracle and/or its affiliates.

1. From the **Data Integration** home page, then the **Actions** drop-down, select **Process Details**.

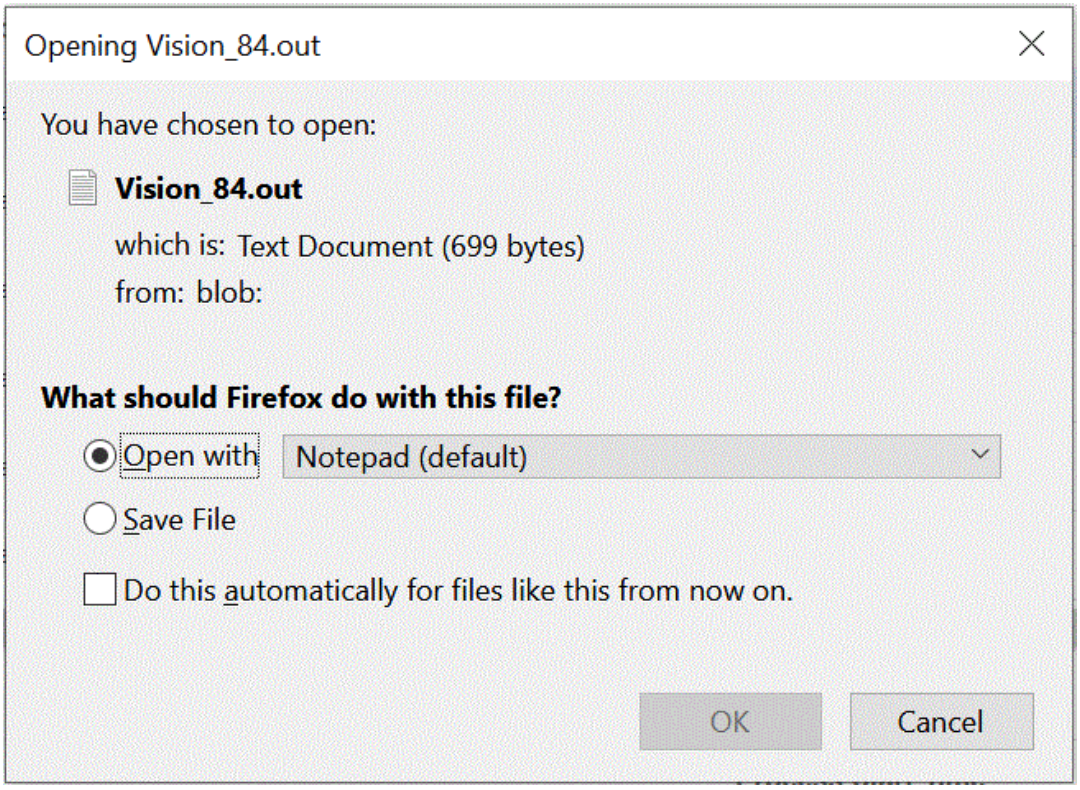
   To view process details for a specific integration, click ••• to the right of the data integration, and then select **Process Details**.

2. Select the **Process ID** of the integration that has generated an error message output file.

   When an integration execution fails, a displays in the **Status** column for the integration.

3. From **Output**, click .

   The Planning application_processid.out page displays.



4. Open or save the error message output file.

   The error message output file is opened in your default text editor.

Resetting an Integration or Pipeline Job Status

Shows how to stop an incorrect execution of a integration or Pipeline job without restarting the instance.

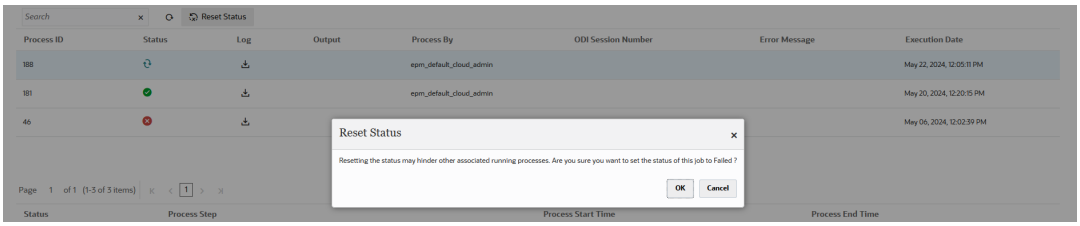
If you need stop an incorrect execution of a integration or Pipeline job without restarting the instance, you can reset the status of an executed job to a "Failed" status. You might do this when a job keeps polling and executing due to an incorrect name or parameter and there is no option to stop or terminate the job except to restart the environment.

To reset a job status:

1. Select the process details for the specific integration or Pipeline by clicking the job ••• to the right of the job, and then selecting **Process Details**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

2. Click **Reset Status**.
3. When prompted, click **OK** on the **Reset Status** page.



Using the Workbench

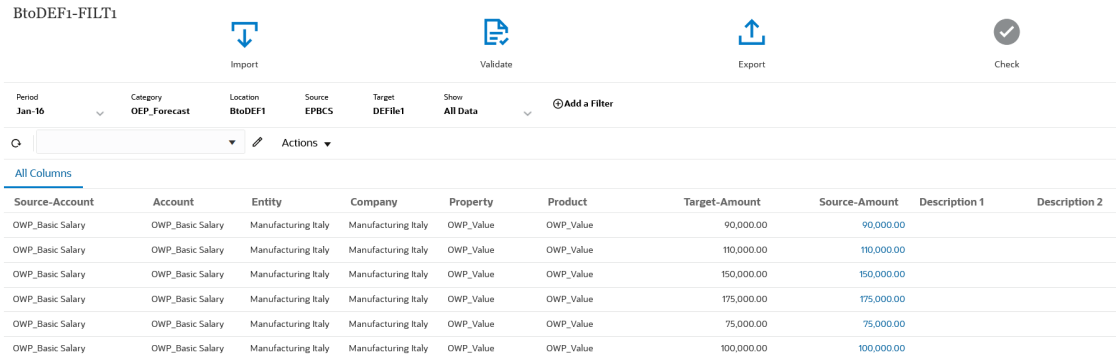
The Workbench provides a point of view (POV) framework to import, view and verify, and export data from your source system.

Key features of the Workbench include:

- an interactive load process with options to import ( ), validate ( ), export ( ), and check data ( ).
- provisions to view Source (All)/Source (Mapped)/Target/Source and Target values
- a PTD/YTD value display for ready reference

Additionally from the Actions menu in the Workbench also enable you to:

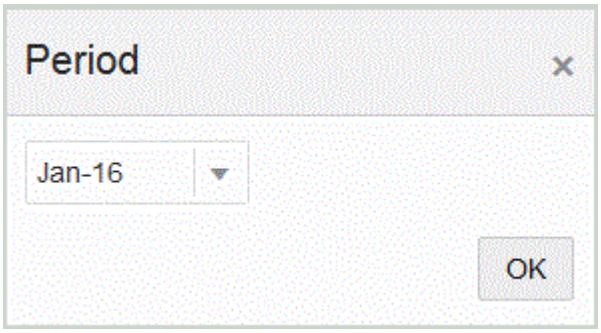
- Export Data—Export data to a .csv file,
- File Browser—Launch the File Browser
- Target View—Display only the target view of the Workbench data.
- Delete Imported Data—Delete all the associated data for a selected period from the staging table?



To display the Workbench:

1. From the **Data Integration** home page, click ••• to the right of the data integration, and then select **Workbench**.
2. Click **Period**, and then from **Period** drop-down, select the POV of the period and click **OK**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



3. **Optional**: To show only selected data in the Workbench, from the **Show** drop-down, select the data to show:

   Available options:

   - All Data
   - Valid Data
   - Invalid Data
   - Ignored Data

4. **Optional**: To filter the data in the Workbench, click **Add a Filter**, and select the value (dimension) to which to apply the filter.

   For more information, see Filtering Workbench Data.

Using the Workbench Workflow

The workflow bar in the Workbench consists of the following tasks:

When a task has been completed successfully, the icon is highlighted. If the task has not been completed, or has been run unsuccessfully, the icon is gray.

You can navigate between tasks on the workflow bar simply by clicking a task.

Task	Description	More Information
Import	Imports the data from the source system and performs the necessary transformations.	Importing Source Data
Validate	Confirms that all members are mapped to a valid target system account. If there are any unmapped dimension maps within the source file, a validation error occurs.	Validating Source Data
Export	Export data to the target application.	Exporting Data to Target Applications
Check	Execute a check to display a Check log for the current POV.	Checking the Data

Copyright © 2017, 2026, Oracle and/or its affiliates.

Importing Source Data

When directed to import a source file, **Data Integration** uses the current POV to determine location, category, and period, and then performs the following:

- if the Replace option is selected, deletes the existing file,
- loads the source file,
- maps source dimensions to target dimensions.

To import source data:

1. From the **Data Integration** home page, click ⋯ to the right of the integration, and then select **Workbench**.
2. **Optional**: Click **Period** to select another period from the current POV.
3. Click **Import** ( ).
4. From **Import Mode**, select the method for importing the data.

   Available import modes:

   - **Append**—Keep existing rows for the POV (not merge data within) but append new rows to the POV. For example, a first-time load has 100 rows and second load has 50 rows. In this case, 50 rows are appended. After this load, the row total for the POV is 150.
   - **Replace**—Clears all data for the POV in the target, and then loads from the source or file. For example, a first-time load has 100 rows, and a second load has 70 rows. In this case, 100 rows are removed, and 70 rows are loaded to TDATASSEG. After this load, the row total is 70.

     For a Planning application, Replace clears data for Year, Period, Scenario, Version, and Entity that you are loading, and then loads the data from source or file. Note that when you have a year of data in the Planning application, but are only loading a single month, this option clears the specific month before performing the load.
5. **For a file-based integration only:** In **File**, depending upon whether you want to upload the data file from the application Inbox or from another location, click **File Browser** to navigate to the file on the **File Browser** and click **OK**.
6. Click **Run**.

Validating Source Data

Validating the source data confirms that all members are mapped to a valid target system account. It also compares the dimension mapping to the source file and identifies unmapped dimensions. The workflow cannot continue until all dimensions are properly mapped, and any validation error are corrected.

To validate source data:

1. From the **Data Integration** home page, click ⋯ to the right of the integration, and then select **Workbench**.
2. **Optional**: Click **Period**, to select another period from the current POV.
3. Click **Validate** ( ).

Copyright © 2017, 2026, Oracle and/or its affiliates.

The validation is submitted for execution. When the process has been executed and the validation fails, the message: "Status Message: Failed" is displayed and the Validate icon is shown in grey. If the validation is successful, the Validate icon is shown in blue.

> **Note**
>
> If you rerun validations in the Workbench, the system won't pick up expressions, just mapping rules.

4. **Optional**: To run the validation offline, click **Continue Offline**.
5. **Optional**: Click to download the log, and then open it.

Exporting Data to Target Applications

After source data passes the validation process, a load file is created. Select the Export option after you have reviewed the data in the data grid and you are sure you want to export it to the target application.

To export data to the target application:

1. From the **Data Integration** home page, click ••• to the right of the integration, and then select **Workbench**.
2. **Optional**: Click **Period**, to select another period from the current POV.
3. Click **Export** ( ).
4. From **Export Mode**, select the method for exporting the data.

   Available export modes:

   - **Merge**—Overwrite existing data with the new data from the load file. (By default, all data load is processed in the Merge mode.) If data does not exist, create new data.
   - **Replace**—Clears all data for the POV in the target, and then loads from the source or file. For example, a first-time load has 100 rows, and a second load has 70 rows. In this case, 100 rows are removed, and 70 rows are loaded to TDATASSEG. After this load, the row total is 70.

     For a Planning application, Replace clears data for Year, Period, Scenario, Version, and Entity that you are loading, and then loads the data from source or file. Note that when you have a year of data in the Planning application, but are only loading a single month, this option clears the specific month only before performing the load.
   - **Accumulate**—Accumulate the data in the application with the data in the load file. For each unique point of view in the data file, the value from the load file is added to the value in the application.
   - **Subtract**—Subtract the value in the source or file from the value in the target application. For example, when you have 300 in the target, and 100 in the source, then the result is 200.
5. Click **Run**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Checking the Data

After exporting data to the target system, execute the Check step to display the Check log for the current POV. If the check log data does not exist for the current POV, an empty log file is created.

To check the data:

1. From the **Data Integration** home page, click ••• to the right of the integration, and then select **Workbench**.
2. **Optional**: Click **Period** to select another period from the current POV.
3. Click **Check** ( ).

   The Check log is submitted for execution.
4. **Optional**: To run the Check log offline, click **Continue Offline**.
5. **Optional**: Click to download the log, and then open it.

Showing Data in the Workbench

You can select the type of data to display in the data grid of the Workbench.

To select the type of data, from the **Show** drop-down, select one of the following options.

Available options:

- Valid Data—Data that was mapped properly and is exported to the target application.
- Invalid Data—One or more dimensions that was not mapped correctly and as a result, the data is not exported to target.
- Ignored Data—User defined explicit map to ignore a source value when exporting to target. This type of map is defined in the member mapping by assigning a special target member with the value of ignore.
- All Data—Show all valid, invalid, and ignored data.

Exporting Data in the Workbench

You can export all the data for a POV to a CSV file. The export process can be run in online or offline mode and any filters can be applied to the export. Data is exported to a CSV or comma separated values text file where the comma character (,) typically separates each field of text. The exported data is named *WorkbenchData_<POV number>.csv* and it is downloaded to your browser's Download folder.

> **Note**
>
> Depending on your Windows setting for opening CSV files, you can open the exported data file automatically in Excel.

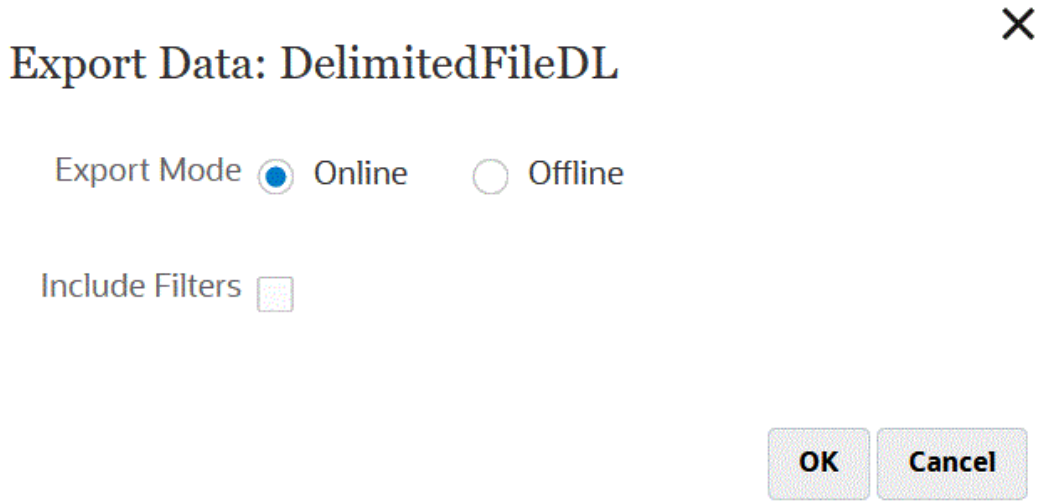
To export data:

Copyright © 2017, 2026, Oracle and/or its affiliates.

1. From the **Data Integration** home page, click ••• to the right of the integration, and then select **Workbench**.
2. From the **Actions** drop-down, select **Export Data**.
3. From the **Export Data** page, then **Export Mode**, select the mode for exporting data.

   Available options:

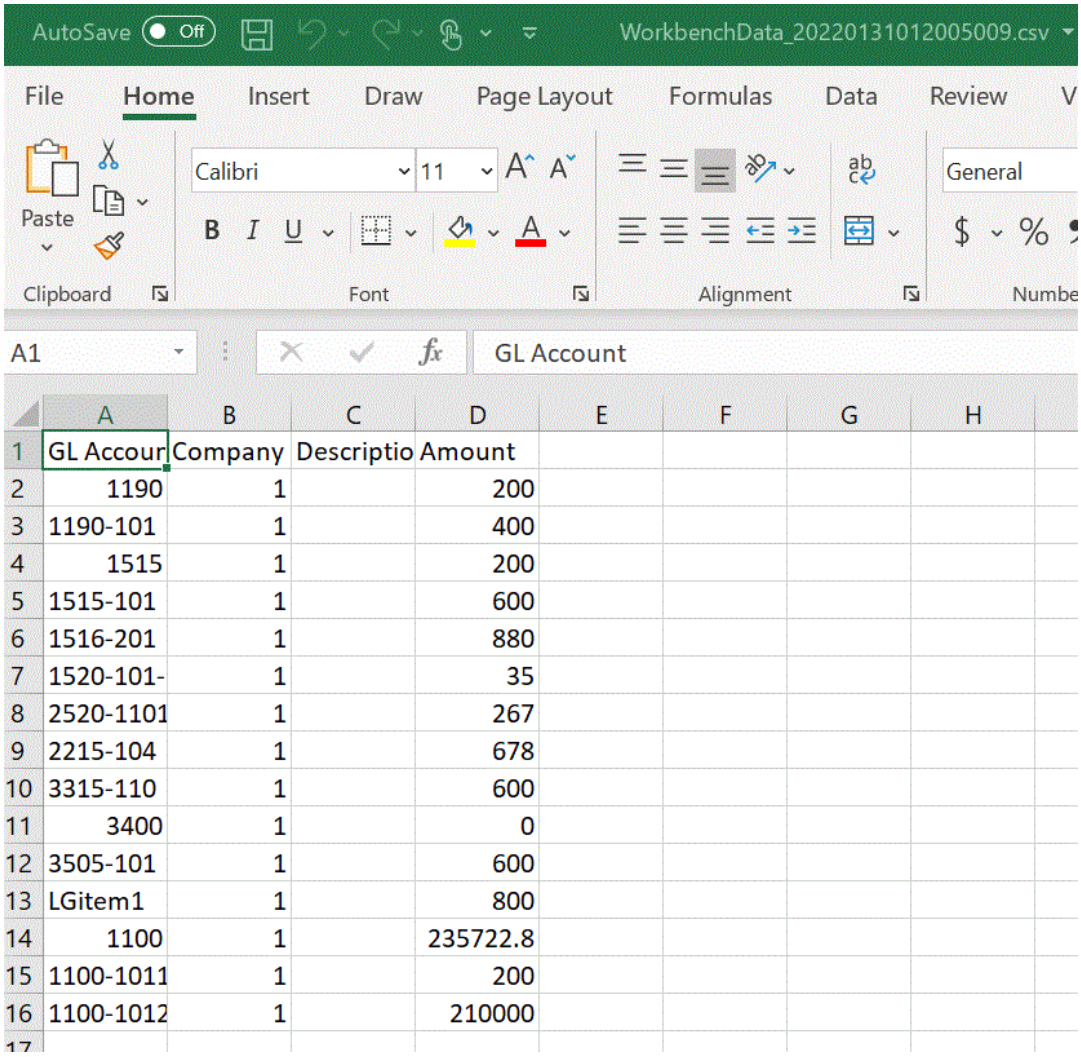
   - online—Processes the export immediately.
   - offline—Runs the export in the background. (recommended if there are a lot of rows).



4. From **Include Filters**, check to apply any filters defined in the integration.
5. Click **OK**.

   Here is an example of an exported data file in Excel format:

Copyright © 2017, 2026, Oracle and/or its affiliates.



	A	B	C	D
1	GL Accour	Company	Descriptio	Amount
2	1190	1		200
3	1190-101	1		400
4	1515	1		200
5	1515-101	1		600
6	1516-201	1		880
7	1520-101-	1		35
8	2520-1101	1		267
9	2215-104	1		678
10	3315-110	1		600
11	3400	1		0
12	3505-101	1		600
13	LGitem1	1		800
14	1100	1		235722.8
15	1100-1011	1		200
16	1100-1012	1		210000

Viewing Target Data

You can view specific details associated with target data used in the integration such as the dimensions to which data is loaded. In this view, you can also select a target row and view the source data associated with the item.

> **Note**
>
> If you have a large volume of data to be viewed, make sure that there are no more than 5,000 rows in the target view. To ensure this, filter the data on the Workbench page.

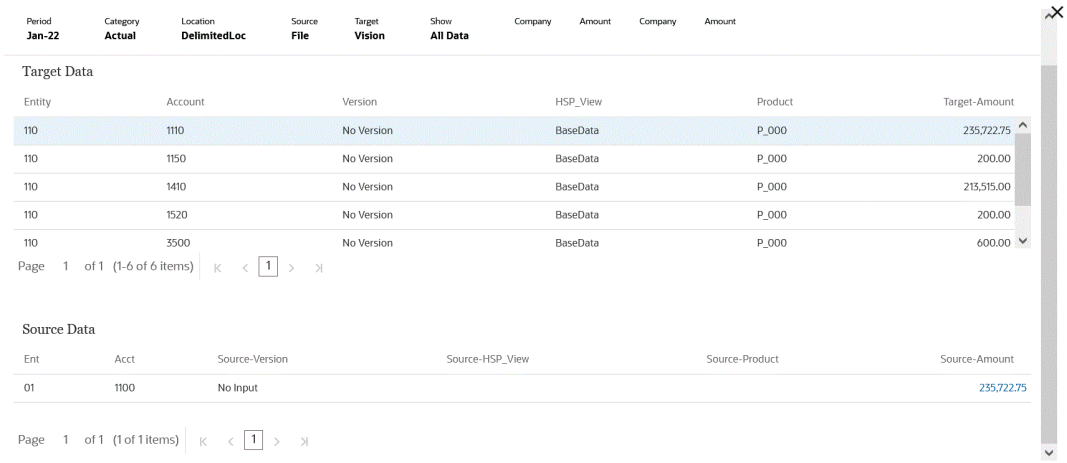
Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> For a Data Export to File application, set the workflow mode to **Full** when you want to see the data in Workbench for testing purpose (by default the workflow mode is Simple for a Data Export to File application. See Selecting the Workflow Mode in *Administering Data Management for Oracle Enterprise Performance Management Cloud*.

To display data in a target view:

1. From the **Data Integration** home page, click ••• to the right of the integration, and then select **Workbench**.
2. From the **Actions** drop-down, select **Target View**.
3. **Optional**: To display the source data associated with a target row, click the target row.

   The source data information is displayed below the Target Data rows.



Deleting Imported Data in the Workbench

Use the Delete All Imported Data option to clear all imported data from your data tables for a POV period. This option enables you to clear data from the system so that you can start over when loading data.

When executed, the Delete All Imported Data option deletes the following:

- mapping audit used for loading
- audit information about the data load
- process load status
- process parameters

> **Note**
>
> There is **no backup** to recover any deleted imported data. It is recommended that you use extreme caution before executing this process.

Copyright © 2017, 2026, Oracle and/or its affiliates.

After the deletion, note the following:

- You cannot view any data in the Workbench for the period.
- You cannot drill down from the Oracle Fusion Cloud Enterprise Performance Management to Data Integration.

> **Note**
>
> Drill regions are not deleted as part of the process. If you need to delete a drill region, then delete it manually.

- You cannot restore maps for the given POV period.

> **Note**
>
> All setup data for example application registration, import format, and mappings are retained and not impacted by the purge process.

To delete all imported data:

1. From the **Data Integration** home page, click ••• to the right of the integration, and then select **Workbench**.
2. From the **Actions** drop-down, select **Delete Import Data**.

   The Delete Confirmation page is displayed with the following message: "Are you sure you want to delete all the associated data for period Jan-22 from staging table?"
3. Click **OK**.

Viewing Mappings in the Workbench

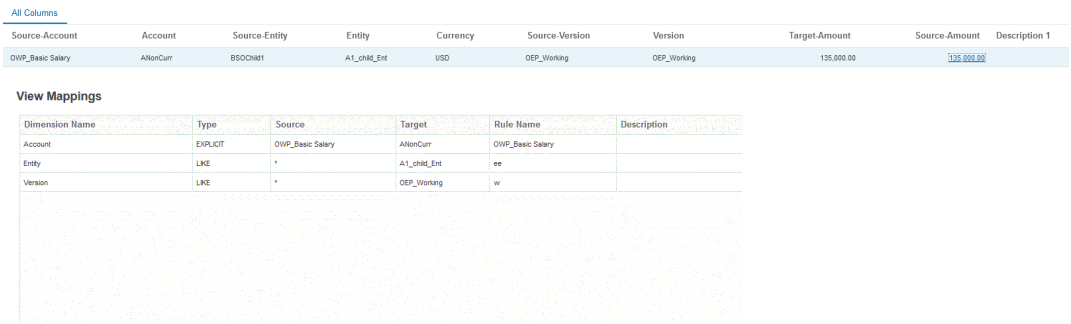
Use View Mappings to view how source dimensionality translates to target dimensionality based on source values. Member mappings are referenced during the data load, enabling Data Integration to determine how data is dimensionalized when loaded to the target application.

To view mappings:

1. From the **Data Integration** home page, click ••• to the right of the integration, and then select **Workbench**.
2. Click the source amount for which you want to view mapping and then select **View Mapping**.

   In the following example the View Mapping page has been launched for the 135,000.00 source amount.

Copyright © 2017, 2026, Oracle and/or its affiliates.

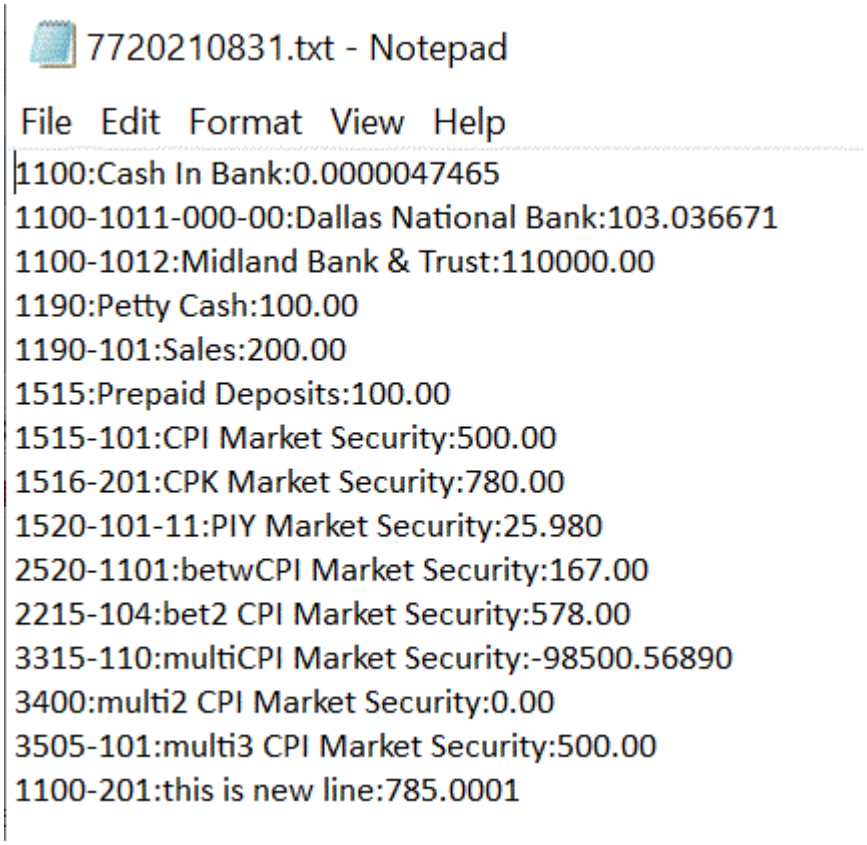


Viewing the Source Document in the Workbench

You can view the source file for an integration in the Workbench.

To view the source document for a file:

1. From the **Data Integration** home page, click ⋯ to the right of the file-based integration, and then select **Workbench**.
2. Click the source amount for which you want to view mapping and then select **View Source Document**.
3. When prompted open or save the source document in your text editor.



Filtering Workbench Data

You can apply filters to view specific values in the Workbench. You can create filter criteria based on one or dimensions using a "Contains" or an "Equal" operand and a specific value.

Copyright © 2017, 2026, Oracle and/or its affiliates.

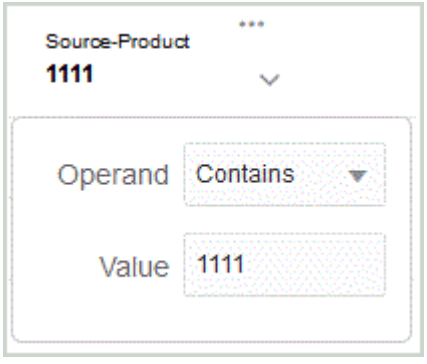
To filter Workbench data:

1. From the **Data Integration** home page, click ••• to the right of the integration, and then select **Workbench**.
2. **Optional**: Click **Period**, to select another period from the current POV.
3. Click **Add Filter**.
4. Select the value (dimension) to which to apply the filter.

5. From the **Operand** drop down, select the operand and value for the filter.

   Available operands are:

   - Contains—Return values that contain the string characters in the Value field.
   - Equals—Return values that are equal to the string characters in the Value field.



Copyright © 2017, 2026, Oracle and/or its affiliates.

6. In **Value**, specify the specific value to use for the filter.

Defining a Custom View in the Workbench

When working in the Workbench, you can create a view definition of the columns in the display. This feature enables you to apply a custom view of the data set that is active in the Workbench. Views provide you with different ways to look at the columns in the Workbench by allowing you to rearrange the representation of data. For example, you can create a view that enables you to focus on specific Account columns or hide Entity columns based on your requirement. Columns can be renamed, resorted, included, or excluded in a custom view definition.

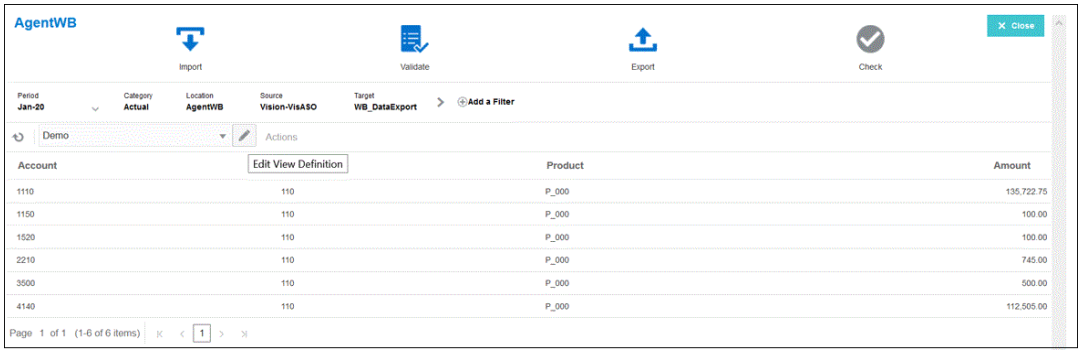
Administrators can also associate a view definition as a public view, which makes it available to everyone or assign it as private, which makes is available only to the administrator.

> **Note**
>
> When drilling into Oracle Smart View for Office, Data Integration uses the last used view on the Drill landing page. If no last used view is found, Data Integration uses the default view selection in this setting

To create a view definition:

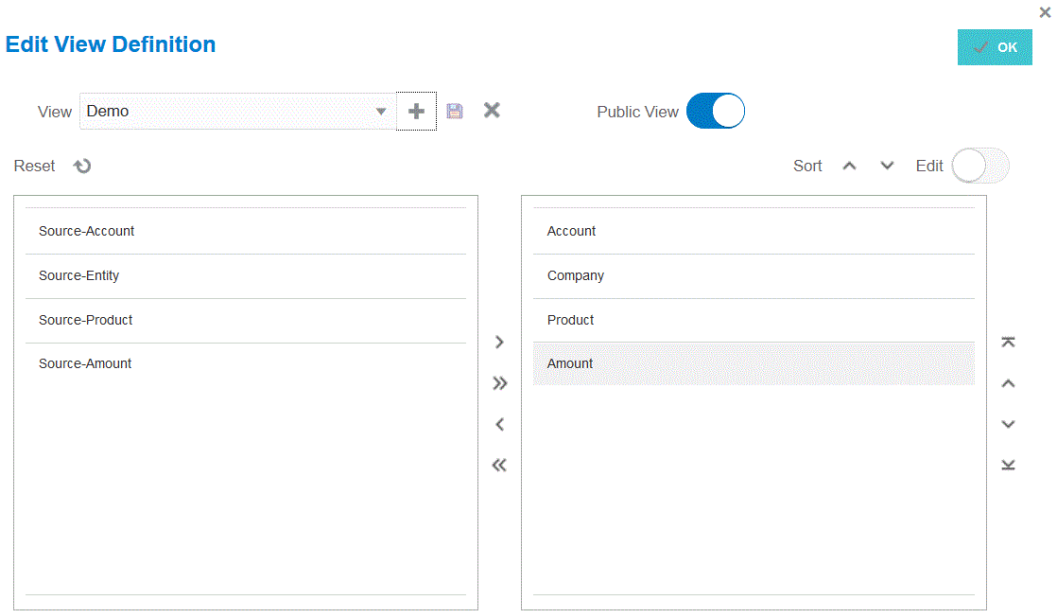
1. From the **Data Integration** home page, click ••• to the right of the data integration, and then select **Workbench**.
2. Click **Period**, and then from **Period** drop-down, select the POV of the period and click **OK**.
3. From the **View** drop-down, select the default view from which to build your custom view definition and click .

> **Note**
>
> Workbench views displayed on any POV are only filtered by the target application and not by the target plan type selected for the integration. This may show views in the drop-down that may not work for the POV.



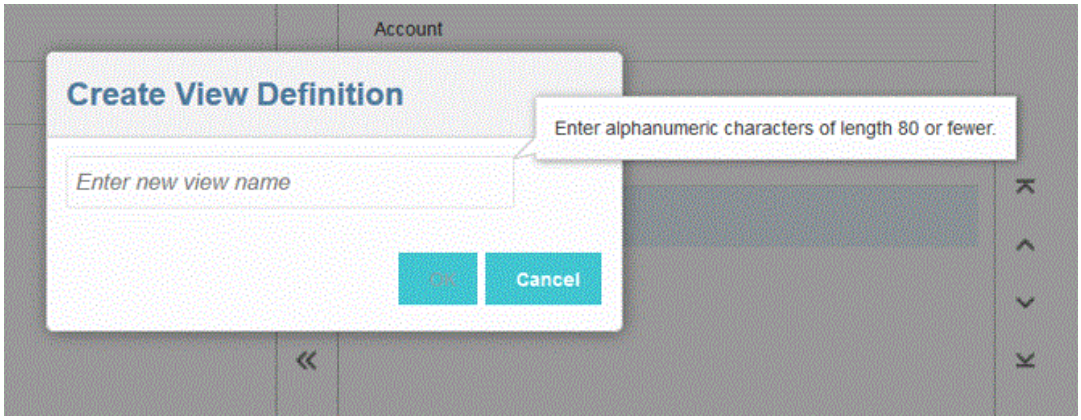
4. From the **Edit View Definition** page, click .

Copyright © 2017, 2026, Oracle and/or its affiliates.



5. In the **Create View Definition** windows, enter the name of the custom view definition.

   The view definition name can contain alphanumeric characters. The view definition name can have up to eighty characters or less.



6. From the Available Columns left pane, select the column(s) to add to the Show Columns pane and click ›.

   You can add all available columns to the Show Columns pane by holding down Shift and clicking ».

   Additional move options include:

   - ‹—Move a selected column from the **Show Columns** pane to the **Available Columns** pane.
   - «—Move all selected columns from the **Show Columns** pane to the **Available Columns** pane.

   To clear the populated values in the view, click ↻ (Reset icon).

7. **Optional**: To rename a column in the **Show Columns** pane, select the column, slide **Edit** on, and then type the new column name.

Copyright © 2017, 2026, Oracle and/or its affiliates.

8. **Optional**: To associate a view definition as public, slide **Public** on.

   A view definition associated with a public view is available to all users.

   To associate a view definition as private, slide **Public** off.

   A view definition associated with a private view is available to the administrator only.

9. **Optional**: To rearrange a column in the **Show Columns** pane, select the column and then select one of the following:

   - —Move column to the top of the display.
   - —Move column up one position of the display.
   - —Move column down one position of the display.
   - —Move column down to the bottom the display.

10. **Optional**: Click **Sort** to sort columns in the **Shown Columns** pane in or order.

11. Click **OK** or to save and update the view definition.

12. **Optional**: To delete a view, click .

Validating Source Data

Shows how to validate the source data confirms so that all members are mapped to a valid target system account. If there are any unmapped dimension maps within the source file, a validation error occurs. Validation compares the dimension mapping to the source file and identifies unmapped dimensions. The process flow cannot continue until all dimensions are properly mapped. As part of the validation process, a Fix Mapping feature is available. Users can view and fix mapping errors right from the Validate page.

When you use the Workbench to import, view, verify, and export data from source systems, Data Integration validates the data to be integrated automatically.

Validation of the source data confirms that all members are mapped to a valid target system account. If there are any unmapped dimension maps within the source file, a validation error occurs. Validation compares the dimension mapping to the source file and identifies unmapped dimensions. The process flow cannot continue until all dimensions are properly mapped. As part of the validation process, a Fix Mapping feature is available. Users can view and fix mapping errors right from the Validate page.

> **Note**
>
> If you rerun validations in the Workbench, the system won't pick up expressions, just mapping rules.

Validation with No Mapping Errors

When validation is successful, the Validate icon turns blue.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Validation Mapping Errors

Shows how to display validate source data and then display mapping errors (if any).

Because newly added dimension members may be unmapped, the addition of dimension members to source systems can produce validation errors. If a dimension member is unmapped and you validate the load, a Validation Error page is launched automatically in the Workbench, which shows the number of dimension members that are not mapped (consequently, undefined). You must correct any unmapped dimension members before running the validation again.

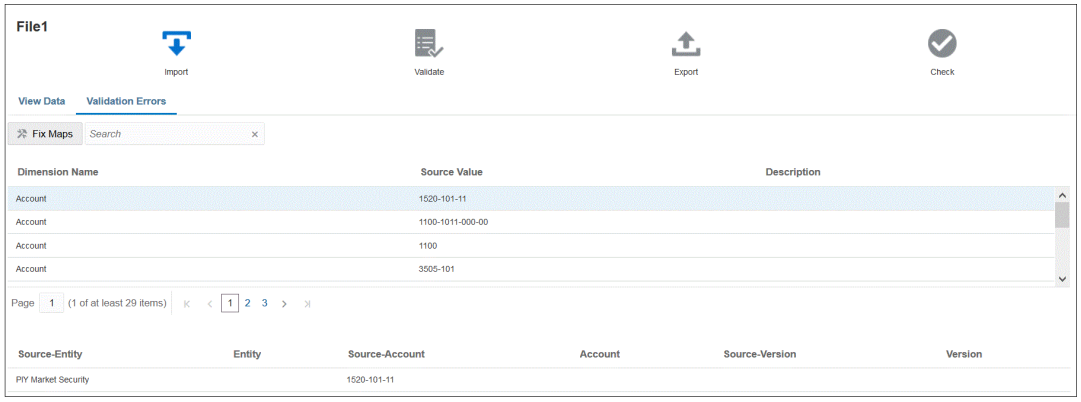
To validate source data:

1. From the **Data Integration** home page, click ••• to the right of the integration, and then select **Workbench**.
2. Click **Validate** ( ).

   To run the validation offline, click **Continue Offline**.

   Click to download the log, and then open it.

   The validation is submitted for execution. When the process has been executed and a validation fails, a Validation Errors page is displayed.



   When the validation is successful, the Validate icon turns blue **Validate** ( ) and no Validation Errors page is launched.

Fixing Mapping Errors

Shows how to view mapping errors and fix them when an integration is executed.

In the Workbench, you can view mapping errors and fix them instantly when mapping errors have occurred.

To validate source data:

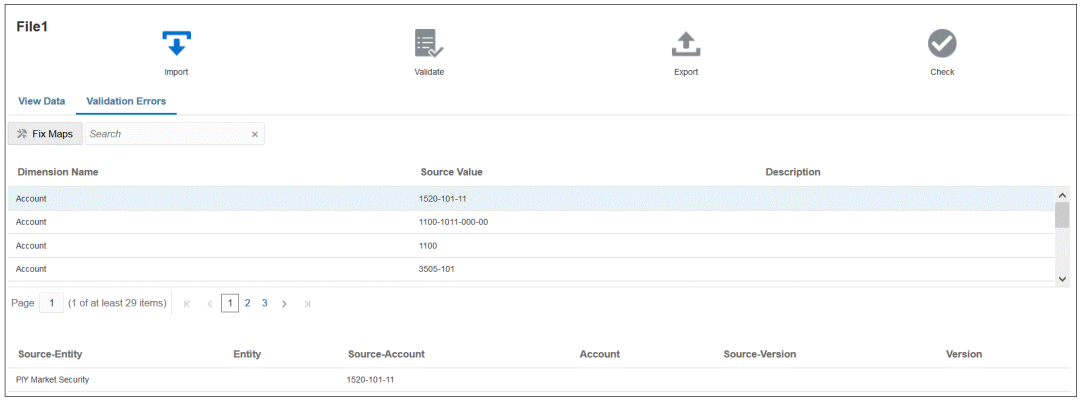
1. From the **Data Integration** home page, click ••• to the right of the integration, and then select **Workbench**.
2. Click **Validate** ( ).

   To run the validation offline, click **Continue Offline**.

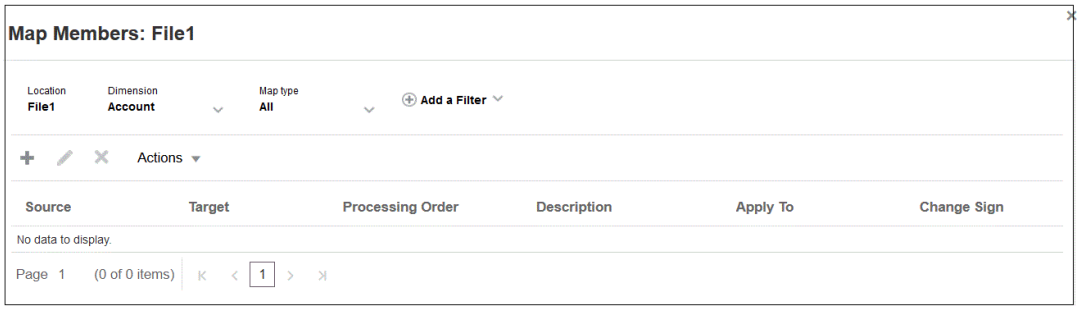
Copyright © 2017, 2026, Oracle and/or its affiliates.

Click to download the log, and then open it.

The validation is submitted for execution. When the process has been executed and a validation fails, a Validation Errors page is displayed.



3. From the **Validation Errors** page, then under **Dimension Name**, select the dimension name or member that needs to mapped or fixed, and then click **Fix Maps**.



4. From the Map Members page, complete one of the following actions.

(Add)—Add a member mapping on the Add Member Mapping page. For more information, see Mapping Members.

(Edit)—Edit a member mapping.

(Delete)—Delete a member mapping.

Copyright © 2017, 2026, Oracle and/or its affiliates.

14

Exporting Data

Various tools, modes, and options for exporting data.

Data Integration provides the tools for exporting data from an Oracle Fusion Cloud Enterprise Performance Management business process to a file, on-premises database, or Oracle Autonomous Database.

To export data, you define an integration with the source as a Cloud EPM business process and the target as one of the data export applications.

When exporting data, use one of the following modes:

- Standard Workflow Mode—Extracts data from Cloud EPM, performs necessary transformations using member mapping and exports the data to the appropriate target application. For more information, see Exporting Data.
- Quick Mode—Extracts data from the Cloud EPM, performs simple transformations using Import Expressions and exports the data to the appropriate target application. In this mode data is not imported to the staging tables and data is directly processed so the performance is significantly better. This mode is also ideal for exporting very large data volumes. For more information, see Quick Mode to Export Data.

For both Standard and Quick Mode methods, you can extract data from a source based on the Essbase storage option using one of the following methods:

1. All Data – ASO or Hybrid enabled BSO
2. Stored Data - BSO
3. Stored Data with Dynamic Calc - BSO
4. Level 0 Data – ASO and BSO (only available for Quick Mode)

Using the Standard Workflow

Extract data using the Standard Workflow mode, which enables you to transform member mappings and then export the data to the target application.

Use the Standard Mode to extract data from the Oracle Fusion Cloud Enterprise Performance Management, perform necessary transformations using member mapping to dimensionalize data, and export the data to the appropriate target application.

- Defining a Data Integration
- Running an Integration

Creating a Data Export File Integration

Export data from an Oracle Fusion Cloud Enterprise Performance Management application to a data file.

Data Export to File enables you to export data from an Cloud EPM application to a data file. You can use the data file to load data into an Oracle ERP Cloud application or an external system.

Copyright © 2017, 2026, Oracle and/or its affiliates.

When creating a data export to file application, note the following:

- When dimensions are created, the order of the column in the data file is used to assign the column order. The first column in file is assigned ACCOUNT data column.
- It is recommended that you export only a year at a time. If date ranges cross a year boundary, data may be duplicated.
- The POV category is not validated.
- Check Rules can be created from a data export file. Check rules cannot be based on target values.
- Write-back is not applicable from data export files.
- Only numeric data types are supported. If you need to export text, SmartList, and dates, use an EPM Data File application with the type: "EPM Data File," and the data extract option "Level 0 Data." Use this application as the target when creating your integration.

  You can also use the groovy public class DataExporter. For more information, see Class DataExporter.
- The default Workflow Mode setting is **Simple** for Data Export to File. When you are exporting data from the Cloud EPM, there is no drill support so the default workflow mode is set to Simple.

  If you want to see the data in Workbench for testing purpose set the Workflow Mode to **Full**. For more information, see Defining Application Detail Options.
- To add a timestamp, create a new export column in the target application, then create a SQL Like Mapping for the next column and in the script just type `CURRENT_TIMESTAMP`.

  Note this method uses a full time stamp, such as `21-JAN-22 04.14.33.677306 PM +00:00`, which is based on the server time and not necessarily the local time.

  To format the timestamp, use the TO_CHAR syntax. For more information, see TO_CHAR (datetime).
- Target values are not validated for data export to file applications.
- When the integration is executed, Data Integration exports the data. Data Integration creates an output data file. The name of the data file is `<Target App Name>_<Process ID>.dat`, and it is written to the `<APPL ROOT FOLDER>/outbox` directory. You can access the data file from the Process Details page from the `OUTPUT file` column.
- Using the **Download File Name** field in Data Export Target Application options, you can also specify your own file name when generating a data export output file as part of a data export to file integration definition, for example, **PlanExport.csv**.

  The system creates the output file in the platform `inbox/outbox` folder. The `inbox/outbox` folder is accessible by navigating to the Application card from the **Home** page, then **Overview**, and then from the **Actions** drop-down, selecting `Inbox/Outbox Explorer`.

  You can also create the file under Data Integration folders by prefixing the folder to the file name, for example **inbox/PlanExport.csv** or **outbox/PlanExport.csv**.

> **Note**
>
> Sub-folders under the `inbox` are supported, for example, inbox/userx/PlanExport.csv. You cannot create sub-folders under the `outbox`.

Copyright © 2017, 2026, Oracle and/or its affiliates.

If the Download File Name field is left blank, the output file is created in the outbox folder with the name **<Application Name>_<Process ID>.dat**.

- If you have a data export job scheduled on a daily basis and need to have the output file distributed to an individual automatically, use the Download Application Snapshot in *REST APIs* to download a particular file, provide the path to that file as the value of `applicationSnapshotName`. For example, to download a Data Integration file called `s112.csv` in the inbox, refer to the file as `"inbox/s112.csv"` as the path parameter.

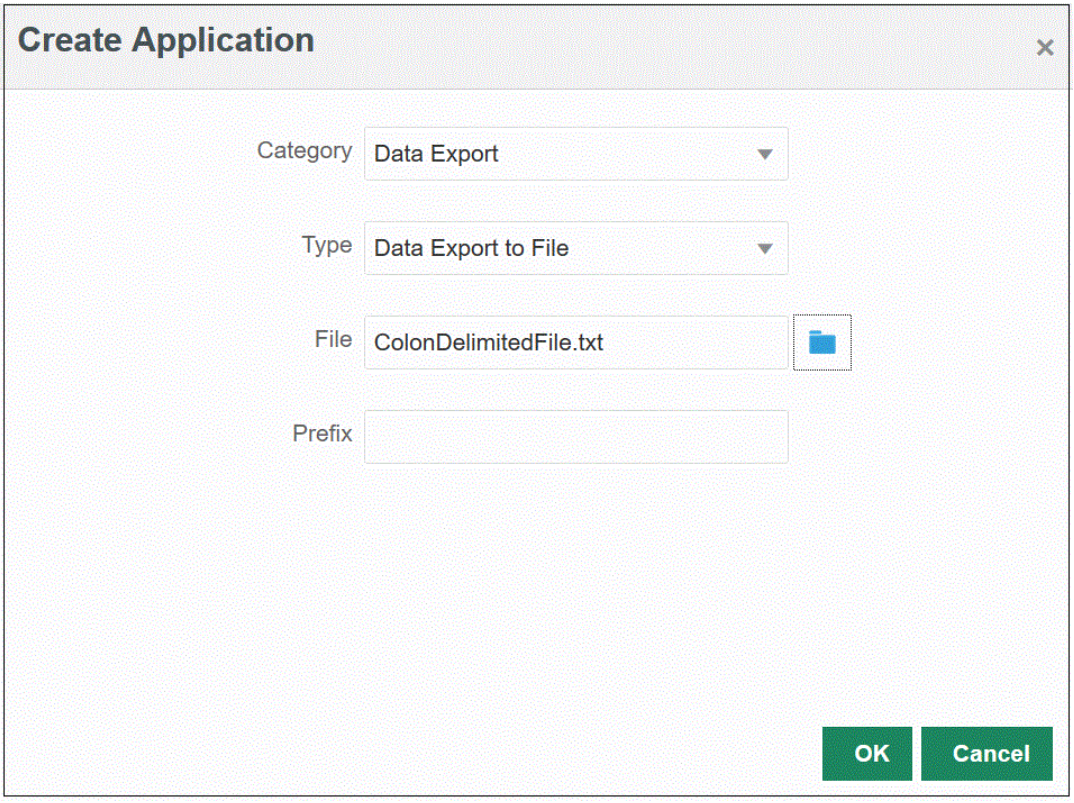
To create a Data Export File integration definition:

1. Create a .CSV file with the list of columns in the order that you want exported.

> **Note**
>
> Do not include the Amount column in the data file. If it's included, you can delete it after the application is created.

The name of the file is the name of the application so name the file appropriately.

2. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications** .
3. On the **Applications** page, click ⋯ before the **Actions** menu, and click **Create Application**.
4. On the **Create Application** page, and then **Category**, select **Data Export**.



5. From **Type**, select **Data Export to File**.
6. In **File**, click **File Browser** and specify the name of the source file from which to create the application.

Copyright © 2017, 2026, Oracle and/or its affiliates.

7. In **Prefix**, specify a prefix to make the application name unique.

   The prefix is concatenated with the file name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.

8. Click **OK**.

   The system registers the application.

9. On the **Application** page, click ••• to the right of the data target application, and then select **Application Details**.

10. Click the **Dimensions** tab.

11. Edit the **Dimension Name** and **Data Table Column Name** as needed.

12. In **Sequence**, specify the order in which the maps are processed.

    For example, when Account is set to **1**, Product is set to **2**, and Entity is set to **3**, then Data Integration first processes the mapping for Account dimension, followed by Product, and then by Entity.

13. In **Column Order**, specify the order of each column in the data export file.

    By default, Data Integration assigns the "Account" dimension as the first column in the order.

14. Click **Save**.

15. In **Application Details**, select the **Options** tab.

16. Select any applicable properties and values for the data export file.

    For more information, see Data Export Target Application Options.

17. Click **Save**.

18. On the Data Integration home page, click ••• before the **Actions** menu, and then select **Create Integration**.

    The General page is displayed in Create Integration view.

19. In **Name** and **Description**, enter a name and description for the new direct integration.

20. In **Location**, enter a new location name, or pick an existing location to specify where to load data.

    A location is used to link a Source and Target along with the associated member mappings. You can define multiple integrations within a same location with different set of Options and Filter criteria. The location is primarily used to control access to an integration. You can use Location Security to control access to end users.

    If you enter a new location, you must provide the Source and Target. When you save the integration, Data Integration creates the location automatically.

    If you select an existing location, Data Integration populates the Source and Target values automatically and you cannot change them.

21. From **Source**, select the direct integration source system.

22. From **Target**, select the data export file target application.

23. Click **Save**.

24. Select **Map Dimensions** and map source dimension members to their corresponding target application dimension members.

    For more information, see Mapping Dimensions.

Copyright © 2017, 2026, Oracle and/or its affiliates.

25. Run the integration to extract the data from the source system and load it to the data export file target application.

    For more information, see Running an Integration.

Quick Mode to Export Data

Extract data using the Quick mode, which supports simple transformations using import expressions but does not support member mappings.

The Quick Mode method provides significant performance improvement over the Standard workflow method. It supports simple transformations using import expressions and does not support member mappings. Data is not loaded to a staging table so you cannot view the data in the Workbench. For example, the Level 0 extract method allows you to apply filters and select columns to include in the output file. This method is ideal for extracting large data slices from the system without reaching query processing limits.

> **Note**
>
> For information on a Quick Mode load using the EPM Integration Agent, see Performing a Quick Mode Load using the EPM Integration Agent.

Considerations:

Consider the following points associated with a Quick Model method:

1. When you create and save a Quick Mode load integration job, you cannot later change it to a standard integration job. However, you can delete the data load integration job.
2. When mapping dimensions, target expression types *are* supported. Target expressions enable you to transform the source value read from the source to target dimension values to be loaded to the target application.

   Target expressions that can be used for the Period dimension include: substring(), split(), map() toPeriod() and toYear().

   With the exception of the SQL target expression type, all remaining target expressions are supported.
3. When mapping dimensions, source expression types *are not* supported.
4. Mapping members is not supported.
5. If you select the Level 0 data extract method, the system automatically creates the business rule "DM BR Data Export" to perform the data extract from the Oracle Fusion Cloud Enterprise Performance Management.
6. When you use the Level 0 data extract method, you cannot use a shared member in the source filter.
7. When running multiple column loads using Quick Mode, two header rows are not supported, that is, the column format `2,1` is not supported.
8. Period processing in Quick Mode does not use periods defined in the Period Mapping option. Instead periods are handled like other dimensions and can be filtered in the Source Filters option. You can also specify a single period during execution in which case all the data is loaded to the single period. The other option is to derive the period dimension based on the source system period name using `toPeriod` and `toYear` target expressions.

Copyright © 2017, 2026, Oracle and/or its affiliates.

9. Dynamic Time Series (DTS) members (for example, Y-T-D, H-T-D, Q-T-D, etc.,) that are surfaced for forms and reports are not supported for source filters when using Quick Mode, or anywhere that the member selector for the period dimension is supported.
10. When running the Quick Mode load, valid export modes are:
    - For Planning—Replace, Merge, and Accumulate
    - For Financial Consolidation and Close—Replace, Merge, Accumulate

    Replace is the default mode.

    If using Quick Mode in Merge mode, and there are any errors during the load, the load fails and nothing is loaded. If using Replace mode, valid data is loaded, and invalid data is not loaded. Replace mode results in a partial load of valid data.

    No import modes are available.
11. Multi-year loads using the Replace export mode is not supported at this time.

> **Note**
>
> The Change Sign option is not supported for Quick Mode loads.

12. A direct drill though to source is required when data is loaded using the Quick Mode method. For more information, see Defining Direct Drill.
13. To export text, smart list and dates from your application using Quick Mode, you need to create a target application using the category: "Data Export", the type: "EPM Data File", and the data extract option "Level 0 Data." Use this application as the target when creating your integration.

Quick Mode to Extract Data Process Description

This section describes how to use the Quick Mode method to extract data from your data sources and then load the data directly to a data export file.

1. In **Applications**, create and register a **Data Export** target application with one of the following types of data export file applications to which to export the data:
    - Data Export to File
    - Data Export to On Premises Data
    - Data Export to Oracle Autonomous Database
    - EPM Data File

> **Note**
>
> In Quick Mode, you can export the data to file in the native format. To perform this kind of export, register an application EPM Data File and then define the integration (there are no map dimensions or map members - the system simply provides the file).

    For more information, see Registering a Data Export File Application.
2. Create the integration job between the source and target applications for the data export file:

Copyright © 2017, 2026, Oracle and/or its affiliates.

a. On the Data Integration home page, click ••• before the **Actions** menu, and then select **Create Integration**.

b. On the **Create Integration** page, and then in **Name** and **Description**, enter a name and description for the integration job.

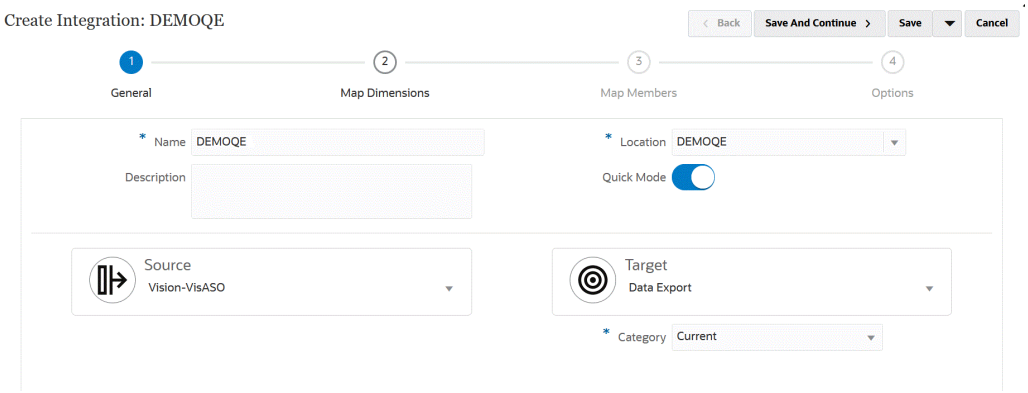
c. In **Location**, enter a new location name, or pick an existing location to specify where to load data.

d. Tap the **Quick Mode** slider on.

When you associate an integration job with the Quick Mode method and then save the job, you cannot reverse the Quick Mode association. However, the integration job can be deleted.

e. From the **Source** drop-down, select the source application.

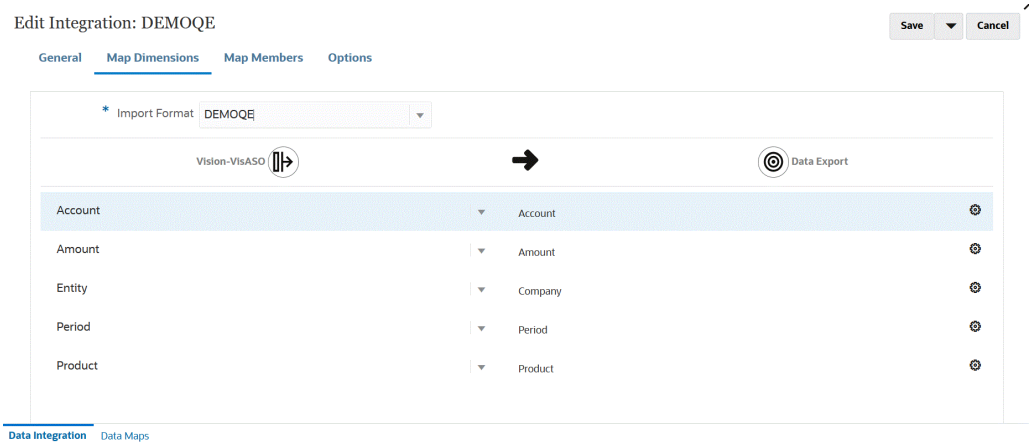
f. From the **Target** drop-down, select the target application for the data export file.



g. Click **Save and Continue**.

3. Map the dimensions between source and the target application:

a. On the **Mapping Dimensions** page, from **Type**, select the type of data load method.



b. In the mappings grid, map the source columns source to the dimensions in the target application for the data export file by completing the following:

i. In **Select Source Dimension**, select the name of the source dimension to map to the Oracle Fusion Cloud Enterprise Performance Management dimension.

ii. **Optional**: Add a target expression for each of the Cloud EPM dimensions.

Copyright © 2017, 2026, Oracle and/or its affiliates.

For information on using target expressions, see Using Target Expressions.

> **Note**
>
> You cannot use a SQL target expression types with the Quick Mode method.
> Source expression types are not available with the Quick Mode method.

c. Click **Save and Continue**.

4. The Map Members option is not available when using the Quick Mode method.
5. Click **Options**.

   You can also go to the Options page from the **Data Integration** home page by clicking ••• to the right of the integration job and then select **Options**.
6. To select a period for the Quick Mode period, click the **Filters** tab.

   Period processing using the Quick Mode method does not use periods defined in the Period Mapping option. Instead, periods are handled like other dimensions and can be filtered in the (Source) Filters option. If you do not filter for the Year and Period, specify a single period at the time of execution.

   > **Note**
   >
   > If you map a period dimension and specify a target expression for the period on the Mapping Dimensions page for the integration, then the **Period** drop-down is not available for selection because the period is derived from the mapping.
   > If you do not specify a filter for the Year and Period, specify a single period when running the integration.

7. Click ➕.
8. From the **Dimension Name** drop-down, select **Period** and then select the single period from the source file from which to load data from **Filter Condition**.
9. Click ➕.
10. **(Optional)**: From the **Dimension Name** drop-down, select **Year** and then select the year from the source file from which to load data from **Filter Condition**.

    For multi-period loads, you can specify filters for the Year and Period dimensions and extract multiple years only when you specify all periods in a year. Partial period extracts crossing multiple periods are not supported.
11. Click **Save**.
12. From the **Data Integration** home page, click ••• to the right of the direct integration, and then select **Options**.
13. Click the **Options** tab.
14. In **Category**, specify an explicit source filter for the **Scenario** dimension.

    Category is used to categorize and map source system data to a target Scenario dimension member in the integration.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Category is not used to determine the Scenario for Quick Mode processing.

15. In **Data Extract Option**, select the method for extracting data:

    Available options:

    - All Data—Extracts stored values and dynamically calculated values for both the Dense and Spare dimensions.

      The All Data data extract option uses the MDX Query Export method to extract data.

      Applies to ASO and BSO Hybrid storage options.

    - Stored and Dynamic Calculated Data—Extracts for stored members and dynamic calculated members for the Dense dimension only and not Spare dimensions. The Stored and Dynamic Calculated Data data option uses the DATAEXPORT command method to extract data.

      Applies to the BSO storage option.

    - Stored Data Only—Extracts stored data only. Dynamically calculated values are excluded in this type of extract. The Stored Data Only option uses the DATAEXPORT command method to extract data.

      Applies to the BSO storage option.

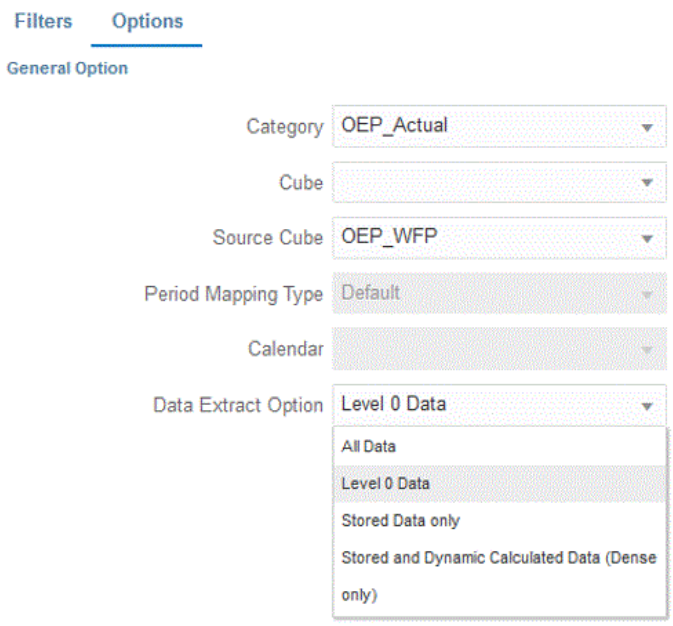
    - Level 0 Data—Extracts entire members at the bottom of dimension (raw data stored in a database) and enables you to apply filters and select columns to include in the output file. This extract option also enables you to export non-numeric data if the target application is an EPM data file. The Level 0 Data option uses the MAXL Export method to extract data. This method can be used only by the Service Administrator. The application is read only when the extract step is executing.

      Applies to ASO and BSO but only for quick mode exports.

      When you select the Quick Mode method, the following target options are not available:

      – Export Attribute Columns
      – Accumulate Data
      – Sort Data
      – Pivot Dimension

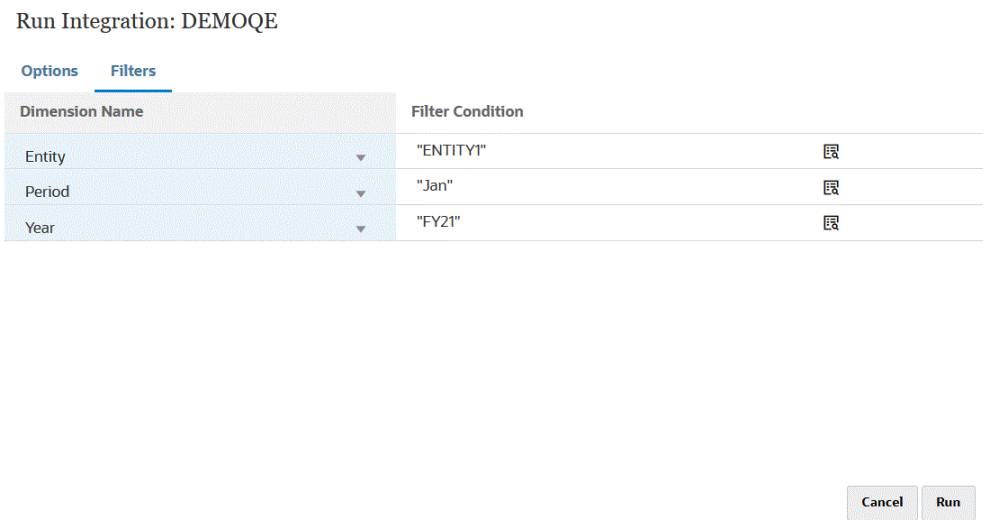
Copyright © 2017, 2026, Oracle and/or its affiliates.



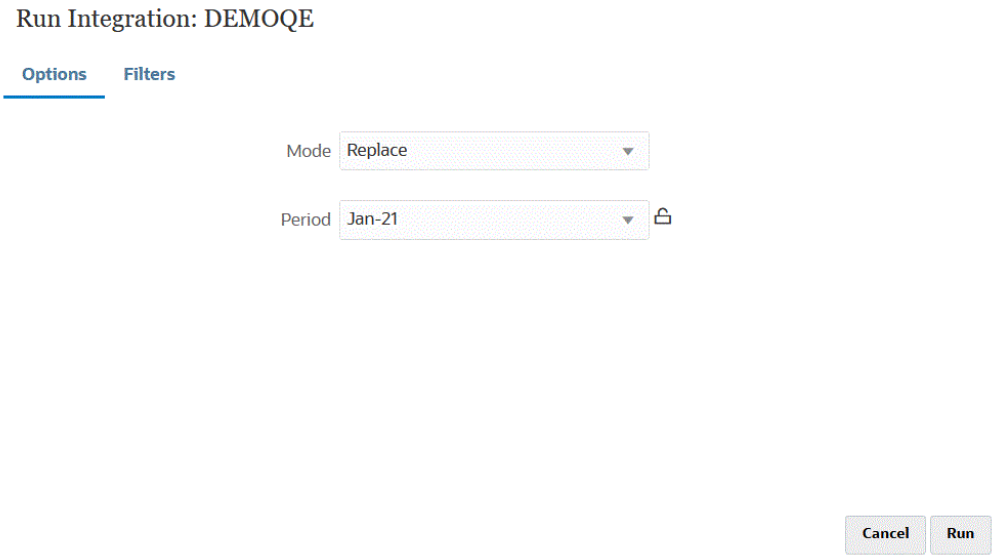
16. On the **Options** page, select any necessary target options needed and click **Save**.
17. Click **Save**.
18. **Run the integration**:
    a. From the **Data Integration** home page, select the integration job associated with the Quick Mode load and then click ▶.
    b. From the **Run Integration** page, the default value for the **Mode** is **Replace**.
    c. If no period was defined on the **Options** page, then from the **Period** drop-down, select the single period from the source file from which to load data.

       If you map a period dimension and specify a target expression for period on the Map Dimension page for the integration, then the **Period** drop-down is not available for selection because the period is derived from the mapping.
    d. If filters have been defined for the integration, click the **Filters** tab and make any changes as desired.

Copyright © 2017, 2026, Oracle and/or its affiliates.

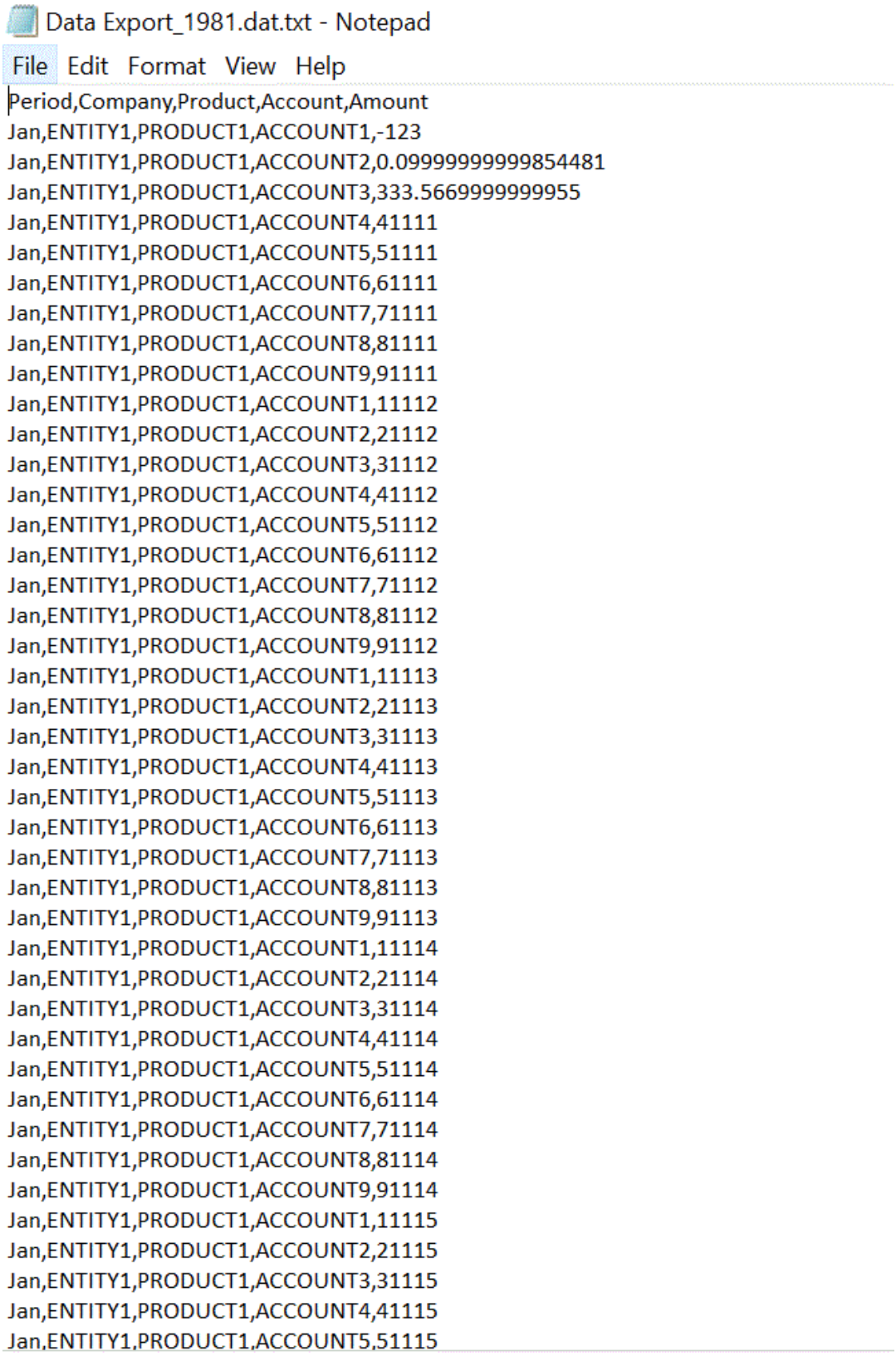


e. Click **Run**.



The following example shows the result of the data export filtered by entity and period.

Copyright © 2017, 2026, Oracle and/or its affiliates.



Data Export_1981.dat.txt - Notepad

File Edit Format View Help

```
Period,Company,Product,Account,Amount
Jan,ENTITY1,PRODUCT1,ACCOUNT1,-123
Jan,ENTITY1,PRODUCT1,ACCOUNT2,0.09999999999854481
Jan,ENTITY1,PRODUCT1,ACCOUNT3,333.5669999999955
Jan,ENTITY1,PRODUCT1,ACCOUNT4,41111
Jan,ENTITY1,PRODUCT1,ACCOUNT5,51111
Jan,ENTITY1,PRODUCT1,ACCOUNT6,61111
Jan,ENTITY1,PRODUCT1,ACCOUNT7,71111
Jan,ENTITY1,PRODUCT1,ACCOUNT8,81111
Jan,ENTITY1,PRODUCT1,ACCOUNT9,91111
Jan,ENTITY1,PRODUCT1,ACCOUNT1,11112
Jan,ENTITY1,PRODUCT1,ACCOUNT2,21112
Jan,ENTITY1,PRODUCT1,ACCOUNT3,31112
Jan,ENTITY1,PRODUCT1,ACCOUNT4,41112
Jan,ENTITY1,PRODUCT1,ACCOUNT5,51112
Jan,ENTITY1,PRODUCT1,ACCOUNT6,61112
Jan,ENTITY1,PRODUCT1,ACCOUNT7,71112
Jan,ENTITY1,PRODUCT1,ACCOUNT8,81112
Jan,ENTITY1,PRODUCT1,ACCOUNT9,91112
Jan,ENTITY1,PRODUCT1,ACCOUNT1,11113
Jan,ENTITY1,PRODUCT1,ACCOUNT2,21113
Jan,ENTITY1,PRODUCT1,ACCOUNT3,31113
Jan,ENTITY1,PRODUCT1,ACCOUNT4,41113
Jan,ENTITY1,PRODUCT1,ACCOUNT5,51113
Jan,ENTITY1,PRODUCT1,ACCOUNT6,61113
Jan,ENTITY1,PRODUCT1,ACCOUNT7,71113
Jan,ENTITY1,PRODUCT1,ACCOUNT8,81113
Jan,ENTITY1,PRODUCT1,ACCOUNT9,91113
Jan,ENTITY1,PRODUCT1,ACCOUNT1,11114
Jan,ENTITY1,PRODUCT1,ACCOUNT2,21114
Jan,ENTITY1,PRODUCT1,ACCOUNT3,31114
Jan,ENTITY1,PRODUCT1,ACCOUNT4,41114
Jan,ENTITY1,PRODUCT1,ACCOUNT5,51114
Jan,ENTITY1,PRODUCT1,ACCOUNT6,61114
Jan,ENTITY1,PRODUCT1,ACCOUNT7,71114
Jan,ENTITY1,PRODUCT1,ACCOUNT8,81114
Jan,ENTITY1,PRODUCT1,ACCOUNT9,91114
Jan,ENTITY1,PRODUCT1,ACCOUNT1,11115
Jan,ENTITY1,PRODUCT1,ACCOUNT2,21115
Jan,ENTITY1,PRODUCT1,ACCOUNT3,31115
Jan,ENTITY1,PRODUCT1,ACCOUNT4,41115
Jan,ENTITY1,PRODUCT1,ACCOUNT5,51115
```


Copyright © 2017, 2026, Oracle and/or its affiliates.

Creating an EPM Data File

Shows how to export text, smart list and dates from your Oracle Fusion Cloud Enterprise Performance Management application and push it to an EPM data file.

You can export text, smart list and dates from your Cloud EPM application and push it to an EPM data file. This process requires that you use the Quick Mode method to extract data from your data sources and then load the data directly to a data export file.

When the data is pushed, Data Integration creates an output data file. The name of the data file is `<Target App Name>_<Process ID>.dat`, and it is written to the `<APPL ROOT FOLDER>/outbox` directory. You can access the data file from the Process Details page from the `OUTPUT file` column.

Using the **Download File Name** field in Data Export Target Application options, you can also specify your own file name when generating a data export output file as part of a data export to file integration definition, for example, **PlanExport.csv**.

The system creates the output file in the platform `inbox/outbox` folder. The `inbox/outbox` folder is accessible by navigating to the Application card from the **Home** page, then **Overview**, and then from the **Actions** drop-down, selecting `Inbox/Outbox Explorer`.

You can also create the file under Data Integration folders by prefixing the folder to the file name, for example **inbox/PlanExport.csv** or **outbox/PlanExport.csv**.

> **Note**
>
> Sub-folders under the `inbox` are supported, for example, inbox/userx/PlanExport.csv. You cannot create sub-folders under the `outbox`.

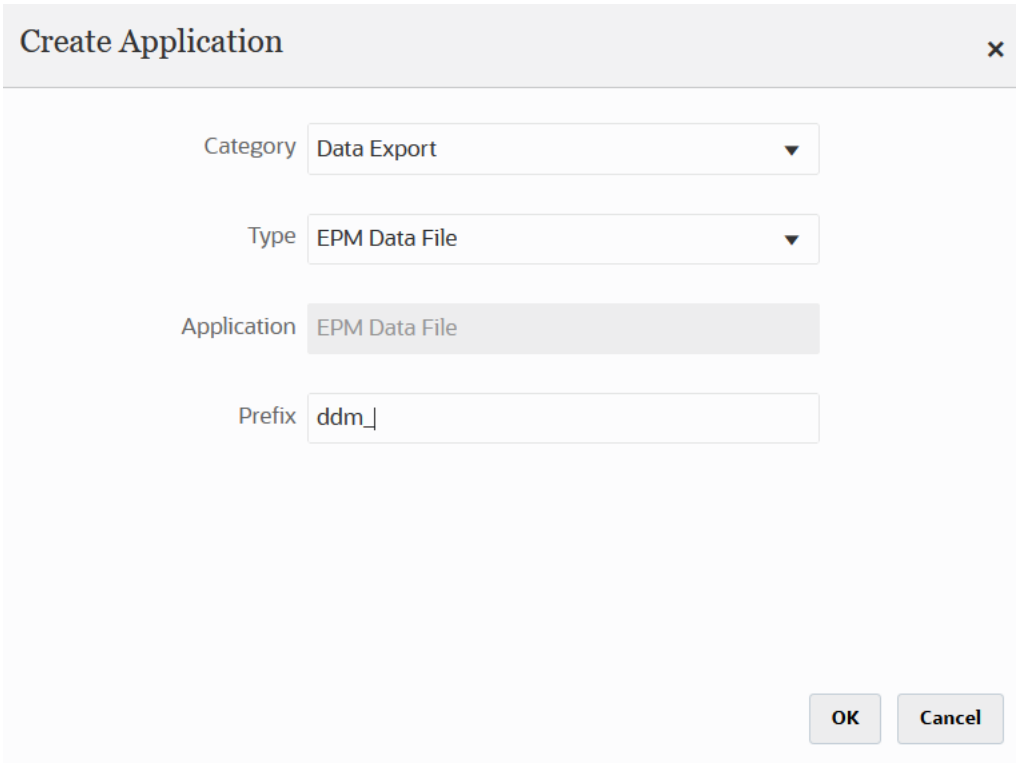
If the Download File Name field is left blank, the output file is created in the outbox folder with the name **<Application Name>_<Process ID>.dat**.

To register a Data Export File with an EPM Data File type application:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, click ••• before the **Actions** menu and then click **Create Application**.
3. On the **Create Application** page, and then **Category**, select **Data Export**.
4. From **Type**, select **EPM Data File**.

   The **Application** field shows the name **EPM Data File**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



5. **Optional**: In **Prefix**, specify a prefix to make the application name unique.

   The prefix is concatenated with the file name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.

6. Create the integration job between the source and target applications for the data export file:

   a. From the **Data Integration** home page, click ••• before the **Actions** menu and click **Create Integration**.

   b. On the **Create Integration** page, and then in **Name** and **Description**, enter a name and description for the integration job.

   c. In **Location**, enter a new location name, or pick an existing location to specify where to load data.

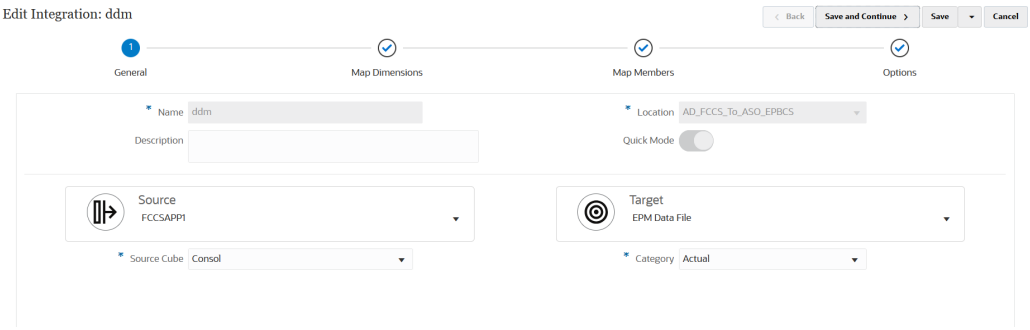
   d. Tap the **Quick Mode** slider on.

      When you associate an integration job with the Quick Mode method and then save the job, you cannot reverse the Quick Mode association. However, the integration job can be deleted.

   e. From the Source drop-down, select the source application.

   f. From the Target drop-down, select **EPM Data File**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



g. Click **Save and Continue**.

7. (**Optional**:) Map any dimensions between source and the target application:

   a. On the **Mapping Dimensions** page, , select the **Import Format**

   b. In the mappings grid, map the source columns source to the dimensions in the target application for the data export file by completing the following:

      i. In **Select Source Dimension**, select the name of the source dimension to map to the EPM data file dimension.

      ii. **Optional**: Add a target expression for each of the Cloud EPM dimensions.

      For information on using target expressions, see Using Target Expressions.

> **Note**
>
> You cannot use a SQL target expression types with the Quick Mode method.
> Source expression types are not available with the Quick Mode method.

   c. Click **Save and Continue**.

   d. The Map Members option is not available when using the Quick Mode method.

   e. Click **Options**.

   You can also go to the Options page from the **Data Integration** home page by clicking ⋮ to the right of the integration job and then select **Options**.

   f. To select a period for the Quick Mode period, select the **Filters** tab.

   Period processing using the Quick Mode method does not use periods defined in the Period Mapping option. Instead, periods are handled like other dimensions and can be filtered in the (Source) Filters option. If you do not filter for the Year and Period, specify a single period at the time of execution.

> **Note**
>
> If you map a period dimension and specify a target expression for the period on the Mapping Dimensions page for the integration, then the **Period** drop-down is not available for selection because the period is derived from the mapping.
> If you do not specify a filter for the Year and Period, specify a single period when running the integration.

Copyright © 2017, 2026, Oracle and/or its affiliates.

g. Click .

h. **(Optional)**: From the **Dimension Name** drop-down, select **Year** and then select the year from the source file from which to load data from **Filter Condition**.

For multi-period loads, you can specify filters for the Year and Period dimensions and extract multiple years only when you specify all periods in a year. Partial period extracts crossing multiple periods are not supported.

i. Click **Save**.

j. Select the **Options** tab.

k. In **Category**, specify an explicit source filter for the **Scenario** dimension.

Category is used to categorize and map source system data to a target Scenario dimension member in the integration.

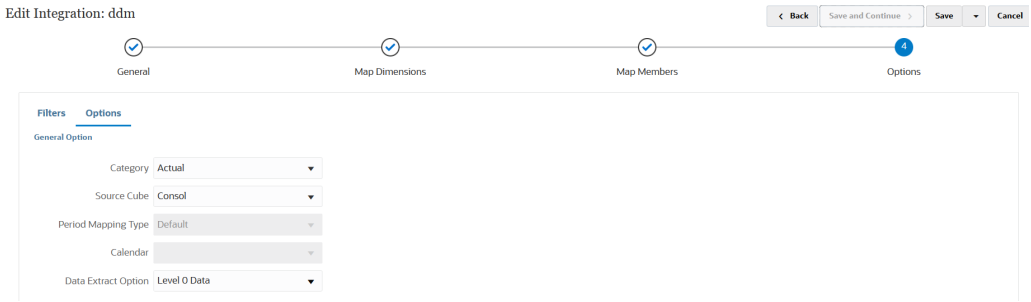
Category is not used to determine the Scenario for Quick Mode processing.

l. From **Data Extract Option**, select **Level 0 Data**.

The Level 0 Data data extract option enables you to extract entire members at the bottom of dimension (raw data stored in a database) and enables you to apply filters and select columns to include in the output file. This extract option also enables you to export non-numeric data if the target application is an EPM data file. The Level 0 Data option uses the MAXL Export method to extract data. This method can be used only by the Service Administrator. The application is read only when the extract step is executing.

If you select the Level 0 data extract method, the system automatically creates the business rule "DM BR Data Export" to perform the data extract from the Oracle Enterprise Performance Management Cloud.

> **Note**
>
> When you use the Level 0 data extract method, you cannot use a shared member in the source filter.



m. Click **Save**.

8. **Run the integration**:

a. From the **Data Integration** home page, select the integration job associated with the EPM Data file integration and then click .

b. From the **Run Integration** page, the default value for the **Mode** is **Replace**.

c. If no period was defined on the **Options** page, then from the **Period** drop-down, select the single period from the source file from which to load data.

Copyright © 2017, 2026, Oracle and/or its affiliates.

If you map a period dimension and specify a target expression for period on the Map Dimension page for the integration, then the **Period** drop-down is not available for selection because the period is derived from the mapping.

d. If filters have been defined for the integration, click the **Filters** tab and make any changes as desired.

e. Click **Run**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

15

Drill Through

Drill through from an Oracle Fusion Cloud Enterprise Performance Management application back to source system transactions.

Data Integration provides the framework to drill-through from an Cloud EPM application back to source system transactions. (Drill through is not supported for human resource data.) Users can drill-through to detail in the source system through Data Integration from the following products:

- Planning Modules
- Planning
- FreeForm
- Financial Consolidation and Close
- Tax Reporting
- Strategic Workforce Planning
- Profitability and Cost Management
- Enterprise Profitability and Cost Management
- Account Reconciliation

Using drill through in Data Integration, you can drill into your data enabling you to answer questions such as what values make up a value Drill through enables you to navigate from source balance within Data Integration back to the source system from which the data was extracted. This allows you to review the detailed (transactional) data that constitutes the source value.

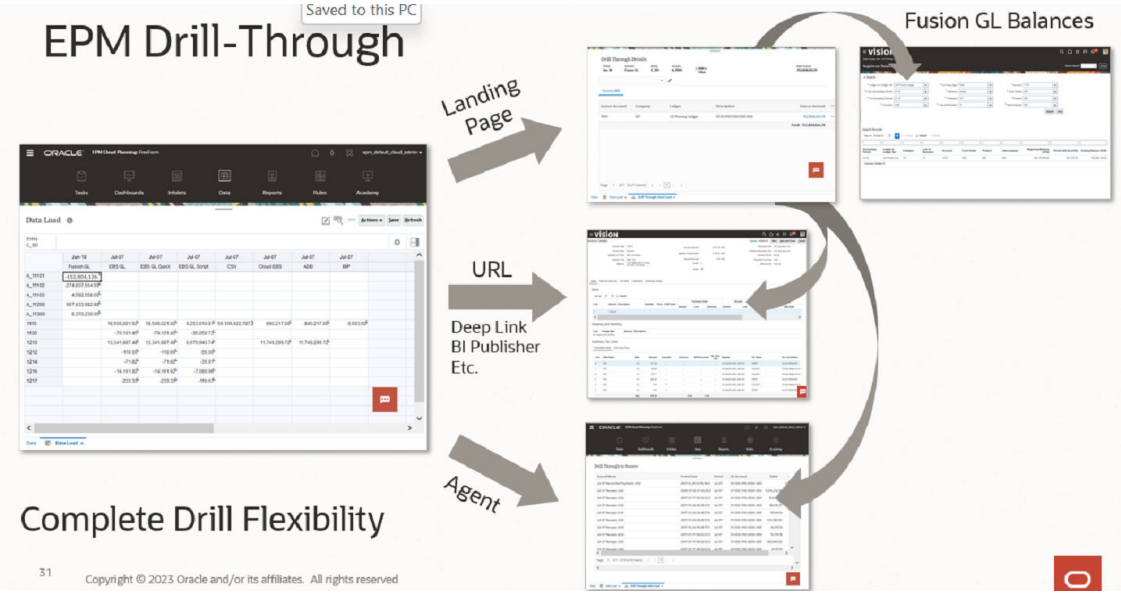
Drill Region definition in Cloud EPM application is used to indicate the drillable cells when you view data in Data Form, Report or Smart View. The drill region can be optionally auto-created by the system when you load data by setting the Create Drill Region option to **Yes**. Alternatively a custom drill region can be defined to support various drill to source options.

Types of Drill Through

For information on the types of drill through, see:

- Using Drill Through to Source
- Drill through to the Oracle ERP Cloud
- Drill through to a BI Publisher Report in the Oracle ERP Cloud
- Drill through to Source using the EPM Integration Agent

Copyright © 2017, 2026, Oracle and/or its affiliates.



Using Drill Through to Source

Explains the Standard and Direct Drill modes and the three types of URLs are supported for drill through.

Data Integration provides a framework that enables you to drill through from your target application to the source application. Use drill through when you want to understand the source of a data value, or you need a granular level of detail for a value.

These are the flavors of drill through supported in Data Integration:

1. **Standard Mode**—In this mode you drill from Oracle Fusion Cloud Enterprise Performance Management data form to a landing page and then drill to the source system data. Using this method, you must load data using Data Integration. You must use the Full Workflow mode, which is supported for all business service types. For this mode you can use the auto-created drill region or custom drill region.
2. **Direct Drill**—In this mode, you drill through directly from any platform-based form to source data, which can be queried and launched using an external URL or an EPM Integration Agent query.

   In this case, you can load the data using Full Workflow mode, Quick mode or even load directly to the Cloud EPM business process. This mode is not supported for Account Reconciliation and Profitability and Cost Management business processes. For this mode, you must define a custom drill region.

   For more information, see Defining Direct Drill.

Three types of URLs are supported in drill through:

1. **Landing Page**—The landing page URL Type enables you to drill from Cloud EPM data form to a landing page and then drill to the source system data. Using this method, you must load data using Data Integration. You must use the Full Workflow mode, which is supported for all business service types. For this mode, you can use the auto-created drill region or custom drill region.
2. **Custom URL or Deeplink**—Use a URL associated with a custom direct drill region definition and display the drill through results in a separate browser.

Copyright © 2017, 2026, Oracle and/or its affiliates.

3. **Agent**—When integrating to an on-premises data source, you can drill down to the source to view the details associated with account balance data loaded to the Cloud EPM.
4. **OADB**—When integrating to an Oracle Autonomous Database data source, you can drill down to the source to view the details associated with account balance data loaded to the Cloud EPM.

For more information about drill through, see Drilling Through to Source Systems in Cloud EPM.

Defining Drill Details

In Standard Drill mode you drill from Oracle Fusion Cloud Enterprise Performance Management data form to a landing page and then drill to the source system data. Using this method, you must load data using Data Integration. You must use the Full Workflow mode, which is supported for all business service types. For this mode you can use the auto-created drill region or custom drill region.

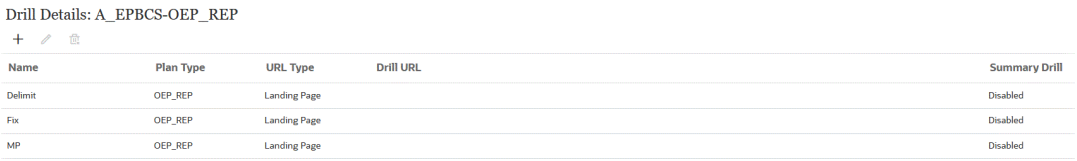
Using a Custom Drill Region, You can specify a drill region, which provides an Oracle Essbase drill slice defined using member names and member functions.

Using Direct Drill, you can drill to source data without staging the data in Data Integration. It also provides the flexibility to simplify the drill bypassing the landing page. Additionally, you can customize the scope of the drill region using Essbase member functions, which simplifies maintenance and improves performance.

With Direct Drill you can load the data using Full Workflow mode, Quick mode or even load directly to the Cloud EPM business process. This mode is not supported for Account Reconciliation and Profitability and Cost Management business processes. For this mode you must define a custom drill region.

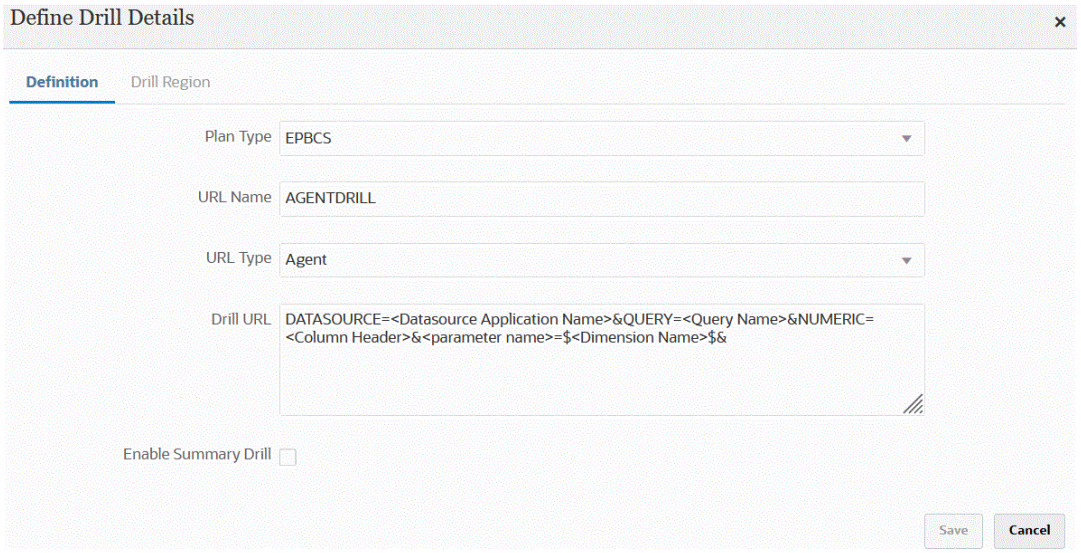
To define a drill definition:

1. From the **Data Integration** home page, then **Actions**, select **Applications**.
2. From the **Application** page, select an application by clicking ••• to the right of a local application, and then select **Drill Details**.



3. From the **Drill Details** page, click + .
4. Select the **Define Drill Detail** page, and then from **Plan Type**, select the source plan type to which you have an integration.

Copyright © 2017, 2026, Oracle and/or its affiliates.



5. In **URL Name**, specify the name of the drill.

   This is the Drill URL Name in Essbase when you open it in Calculation Manager.

   Essbase name restrictions apply to the name of the direct drill including:

   - Use no more than 8 characters when naming a direct drill for a non-Unicode-mode application.
   - Use no more than 30 characters when naming a direct drill Unicode-mode applications.
   - Do not use spaces in the name.
   - Do not use the following characters:

   Table 15-1 Essbase Naming Restrictions

Character	Description
*	asterisk
[]	brackets
:	colon
;	semicolon
,	comma
=	equal sign
>	greater-than sign
<	less-than sign
.	period
+	plus sign
?	question mark
"	double quotation mark
‘	single quotation mark
/	forward slash
\	backslash
|	vertical bars
	tabs

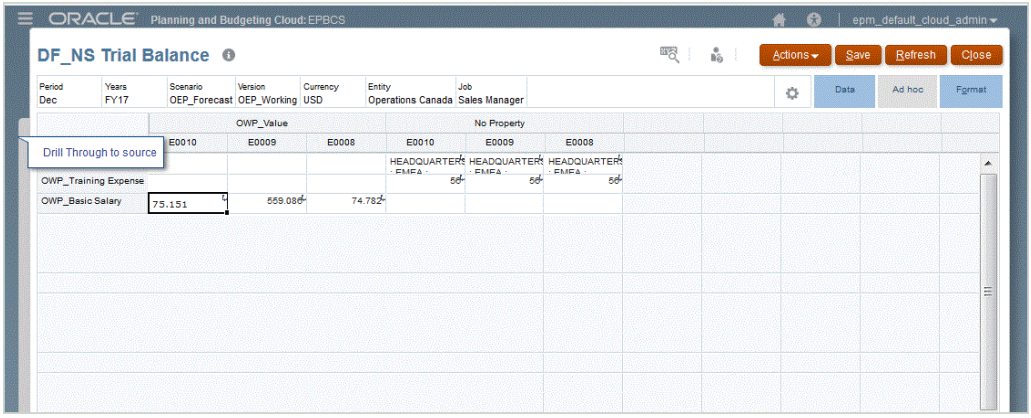
6. From **URL Type**, select the type of URL to use for the direct drill.

Copyright © 2017, 2026, Oracle and/or its affiliates.

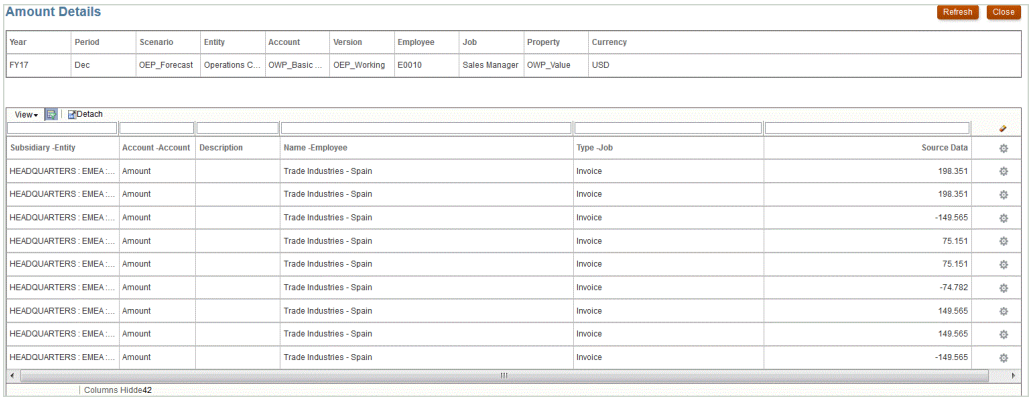
Available types:

- **Landing Page**—Have the system automatically launch the landing page. From this landing page, you can open the source document or continue to drill-through to the defined source system landing page as shown below.

  In the following example, drill through is available from a data form in Planning:



  From this landing page, you can open the source document or continue to drill through to the defined source system landing page.



> **Note**
>
> Drill through functionality is not supported for exchange rates data loaded to Planning applications.
>
> If you map to an alias when loading data to Essbase, then the drill-through might fail because the filter on the landing page uses the member from Essbase. If this case, use the member name instead of the alias.

- **Custom**—Use a URL associated with a custom direct drill region definition and display the drill through results either in Custom - Dynamic tab or Custom - Browser tab.

  In the **Custom - Dynamic** tab, results are shown in a new tab within the Oracle Fusion Cloud EPM.

  In the **Custom - Browser** tab, results are shown in a new browser tab.

  When you define a custom Drill using this setting for a Plan Type, the default drill creation does not create the drill region for the Plan Type. The create drill region flag is ignored if a custom drill is defined.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Optional: You can specify a drill region, which provides an Essbase drill slice defined using member names and member functions.

a. From the **Drill Details** page, select the **Drill Region** tab.

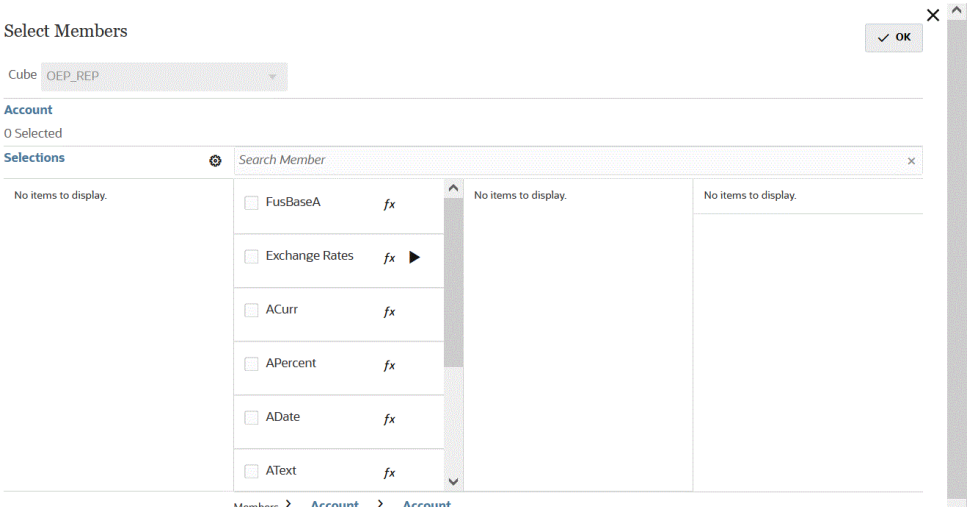
b. Click + .

c. From the **Dimension** drop-down, select the dimension(s) to add to the drill region.

d. From **Filter**, specify a member function for the selected dimension to limit results only to the specified members.

Specify the member between quotation ("") marks. For example, specify the member July as "Jul".

Optionally, you can click to display the Member Selector page and navigate to a selected member to drill on. For more information, see Selecting Members from the Member Selector in the *Smart View for Office User's Guide 22.100*.



- **Agent**—Use the parameters based on the SQL query and server information. In this case, the server information comes from the agent but the rest of the URL is derived from the parameters specified in the SQL query. To use this URL type, the drill URL must include the data source, query, and the parameters to pass.
- **OADB**—Use the parameters based on the SQL query and server information. To use this URL type, the drill URL must include the data source, query, and the parameters to pass.

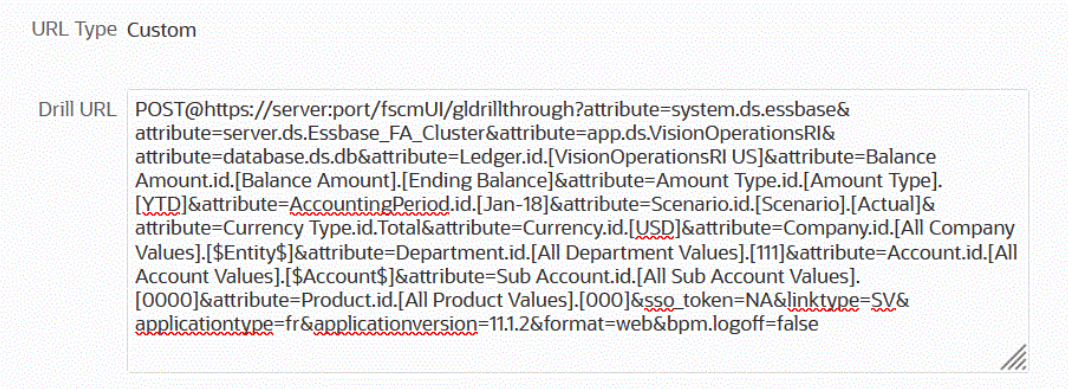
7. In **Drill URL**, specify the URL address for the custom, agent, and OADB direct drill through definitions.

For a custom URL Type, the drill URL must include the server, port, and URL parameters, for example, `https://server:port/<URL Parameters>`.

The following is a custom drill URL. This is a direct drill URL if you want to bypass the landing page:

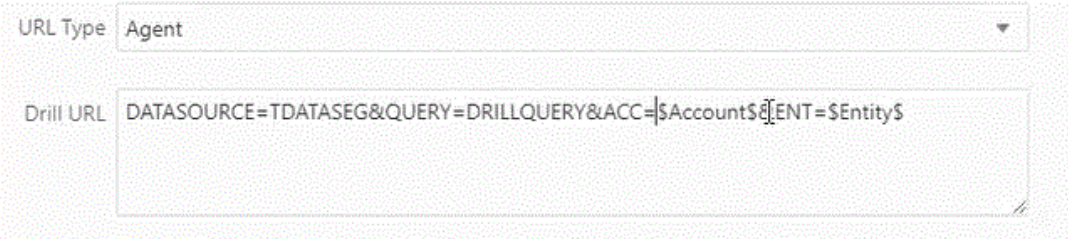
> **Note**
>
> If a static URL contains non-ascii characters, the non-ascii characters must be encoded in the URL definition.

Copyright © 2017, 2026, Oracle and/or its affiliates.



For agent and OADB URL Types, the drill URL must include the data source application name, SQL Query, and number column.

- DATASOURCE—Specify the name of the Data Source Application used to load the data. The source connection information is derived from this application.
- QUERY—Specify the SQL query to use for performing the drill.
- NUMERIC—Specify the number column so it is properly aligned.



For more information about drilling through using the EPM Integration Agent, see Drill through to Source using the EPM Integration Agent.

8. Select the **Enable Summary Drill** option to drill down from summary members in a data form or report and view the detail source data that makes up the number.

   After enabling this option and loading the data with the Drill Region option set to **Yes**, the Drill icon is enabled at the summary level. Drill is limited to 1000 descendant members for a dimension.

   For a summary drill the 1,000 descendants limit is for each dimension. When there are more than 1,000 descendants, the system behavior is controlled by the **Summary Drill Behavior when more than 1000 descendants** option in Defining Application Detail Options.

   A drill from parent with more than 1000 descendants can result inconsistent results. Note the data pattern before allowing users to drill at all levels.

   When the option is set to **Ignore**, then the system completely ignores filtering on that dimension. **Filters** are applied only for the remaining dimensions with level 0 members or descendants less than 1,000. As a result, it is possible you will get members not under the parent you are drilling. When using the Ignore option the only time it works is when the ignored dimensions intersect with unique members of the other dimensions.

   For example in the following scenario, **Location** and **Account** are leaf level members

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 15-2 Location and Account Leaf Level Members

Location	Cost Center	Account
US	110	Expense
US	120	Expense
US	130	Expense
UK	210	Expense
UK	210	Expense

In this case Cost Center intersects with unique rows **Location** and **Account**. If you drill on US, Total Cost Center, and Expense and the option is set to ignore the Cost Center, then the drill returns 3 rows of US.

If on the other hand, lets assume **Location** is a parent with more than 1000 children:

Table 15-3 Cost Center Intersect with Unique Location

Location	Cost Center	Account	Project
US	110	Expense	ABC
US	120	Expense	ABC
US	130	Expense	XYZ
UK	210	Expense	ABC
Uk	210	Expse	XYZ

If you drill on US, Total Cost Center, and Expense, the ABC system ignores **Location** and **Country** and returns all rows that match **Expense** and **ABC**. This results in rows from US and UK.

When enabling drill from Parent, it is recommended to enable drill only from parents with less than 1000 descendants. This can be done by defining drill region manually using Application Drill Details page.

When you select Limit Option, it is possible that all the children do not fall in the 1000 limit resulting in an incomplete result set.

> **Note**
>
> If you select to perform a summary drill on more than 1,000 descendants and include only top-level dimensions but the results are the same as the base-level dimension, then also select base-levels intersections for at least three dimensions.

> **Note**
>
> In order to support a summary drill, your drill query should not include an operator in the WHERE clause of the SQL. For example, your query should be: WHERE COMPANY ~ENTITY~. The system determines appropriate condition (IN, LIKE) based on the number of descendants.

For more information the drill region, see Defining Drill Regions .

9. Click **Save**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

When you save the drill definition, drill regions are automatically created or updated in Essbase. When you delete the Drill Details, they are also deleted in Essbase.

10. Specify a custom drill region on the Drill Region tab if desired.
11. **Optional**: To edit an existing direct drill definition, from the **Drill Details** page, select the definition and click .

    To delete an existing direct drill definition, from the **Drill Details** page, select the definition and click .

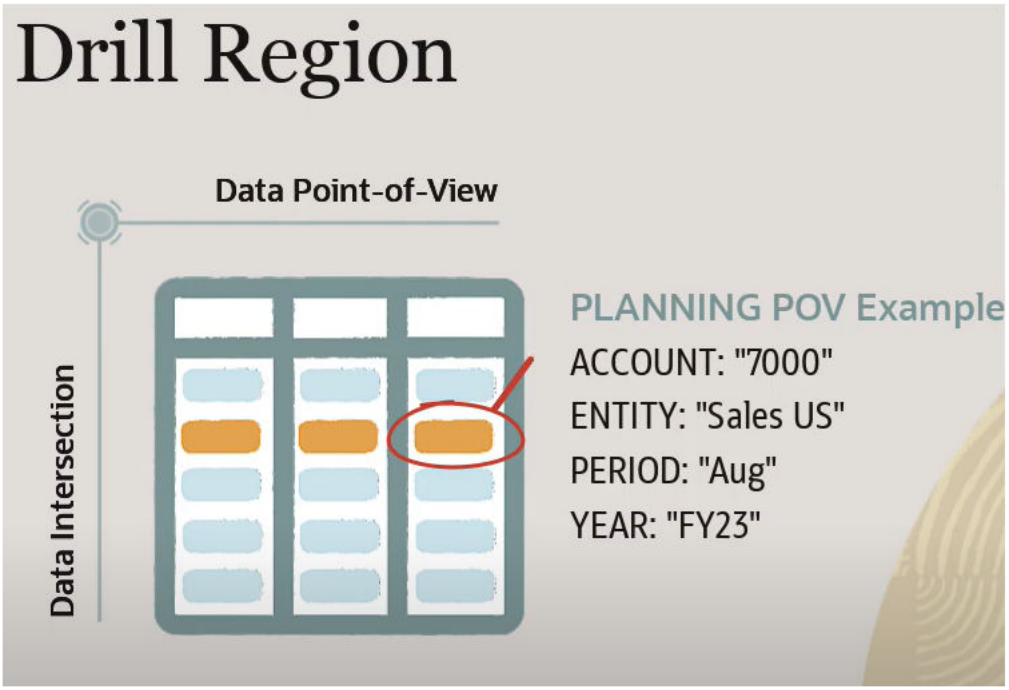
Defining Drill Regions

A drill region is a named region of data intersections in an Oracle Fusion Cloud Enterprise Performance Management application that can be viewed with the drill-through functionality. The data in the drill region is loaded to the Cloud EPM application with Data Integration. In the workspace, you can view the drill region in the data forms of the application.

The drill region stores the uniform resource locator (URL) to get back to Data Integration, and a region name. When you enable drill regions, Data Integration populates them in the Data Integration target application after data is loaded and consolidated. A cell is considered drillable in the target application when it is contained in the drill regions. For Oracle Enterprise Performance Management System applications, the drill region includes: Entity, Account, Scenario, Year, and Period dimensions.

Data Integration creates drill region by scenarios. For any cube (Planning plan types or Planning databases), the name of the drill region is FDMEE_<name of the scenario member>. When creating the drill region, Data Integration checks if a dimension is enabled for the drill.

Drill regions to the source system work only when there is an identical intersection of the data loaded by Data Integration. For example, if you loaded $5000 to Entity2;Account2 intersection, you cannot drill-through from Entity2;ParentAccount2 since no data was loaded for that intersection.



Copyright © 2017, 2026, Oracle and/or its affiliates.

Members of enabled dimensions selected in data loads, are included in the drill region filter. If no dimensions are enabled, the following dimensions are enabled by default: Scenario, Version, Year, and Period. You can enable additional dimensions, and the subsequent data load considers members of newly enabled dimensions. If you disable any dimensions which were previously included in a drill region used for drill creation, members of such dimensions are not deleted during the subsequent data loads. If needed, you can remove obsolete members manually.

Drill regions are defined to identify the cells that are drillable in the Cloud EPM data forms. The drill region can be created using two methods:

1. **Auto create**—If you enable the Create Drill Region in Application Options, then drill regions are created based on the dimensions selected to participate in the Drill Region. The following dimensions are automatically included in the drill region: Scenario, Period, Year, and Version at the application level. A drill region can be defined at the application level in Application Details. A drill region can be enabled at the integration level in Target Options.
2. **Custom drill region**—You can define a Custom drill region using Oracle Essbase member functions. This provides more control and easier maintenance of drill regions. For more information, see Defining Drill Details.

Drill Through to a Landing Page

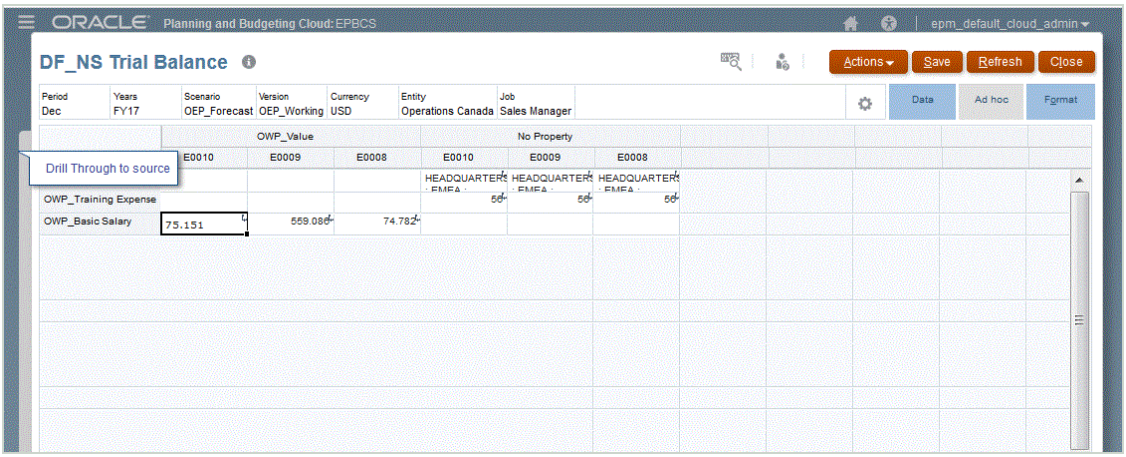
When drilling through to the source from the target application, a separate landing page is displayed that shows all rows that comprise the amount from the selected cell. The landing page displays general ledger accounts and the hyperlinked balances that were used to populate the cells in the Oracle Fusion Cloud Enterprise Performance Management application. When you click a hyperlinked data value, you can drill through to the source system and view the associated journal entries for the selected general ledger account.

When Data Integration is configured to integrate directly from a supported source system, then the drill URL is configured automatically by Data Integration. If you are loading data via a file, then you must configure the URL manually in the import format definition.

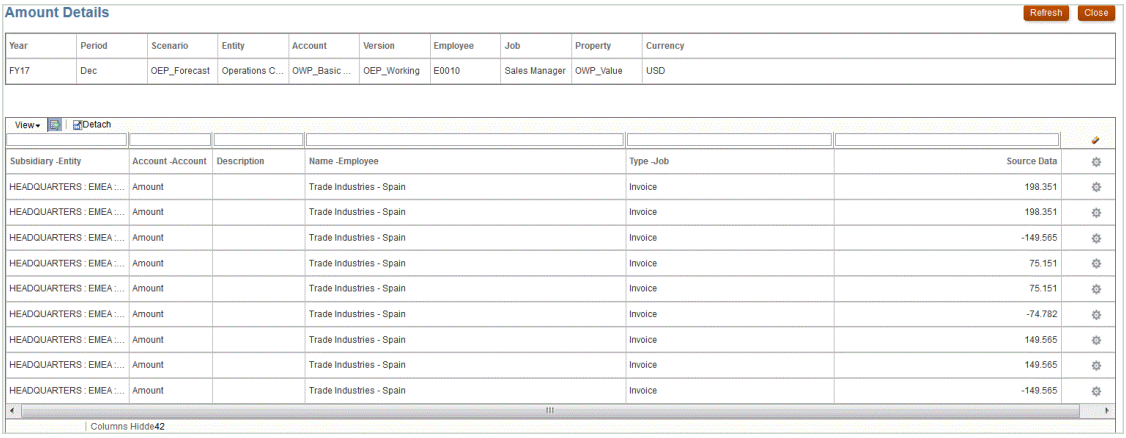
> **Note**
>
> If you map to an alias when loading data to Oracle Essbase, then the drill-through might fail because the filter on the landing page uses the member from Essbase. If this case, use the member name instead of the alias.

In the following example, drill through is available from a data form in Planning:

Copyright © 2017, 2026, Oracle and/or its affiliates.



From this landing page, you can open the source document or continue to drill through to the defined source system landing page.



> **Note**
>
> Drill through functionality is not supported for exchange rates data loaded to Planning applications.

Drill Through to a Custom URL Page

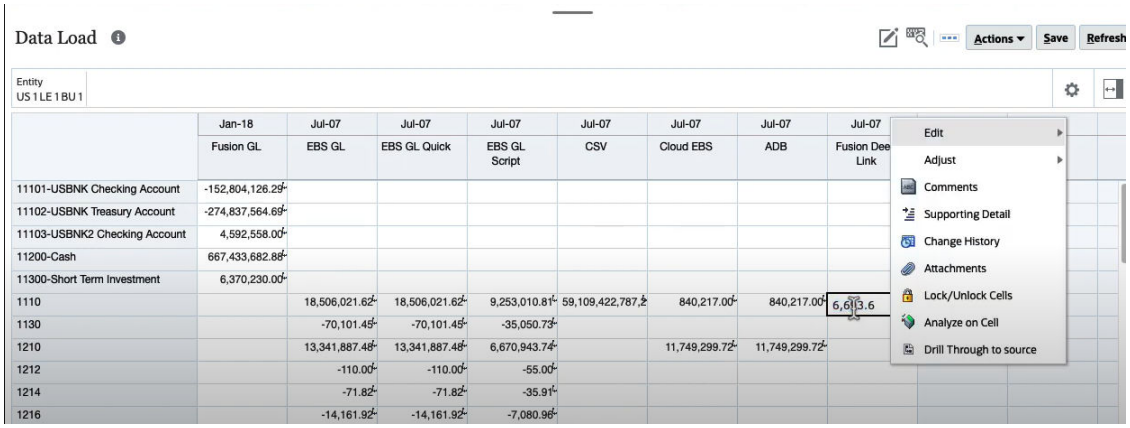
Use a URL associated with a custom direct drill region definition and display the drill through results in a separate browser. For a custom URL Type, the drill URL must include the server, port, and URL parameters, for example, `https://server:port/<URL Parameters>`. Customers must have access to the source data to use a Custom URL.

When you define a custom Drill using this setting for a Plan Type, the default drill creation does not create the drill region for the Plan Type. The create drill region flag is ignored if a custom drill is defined.

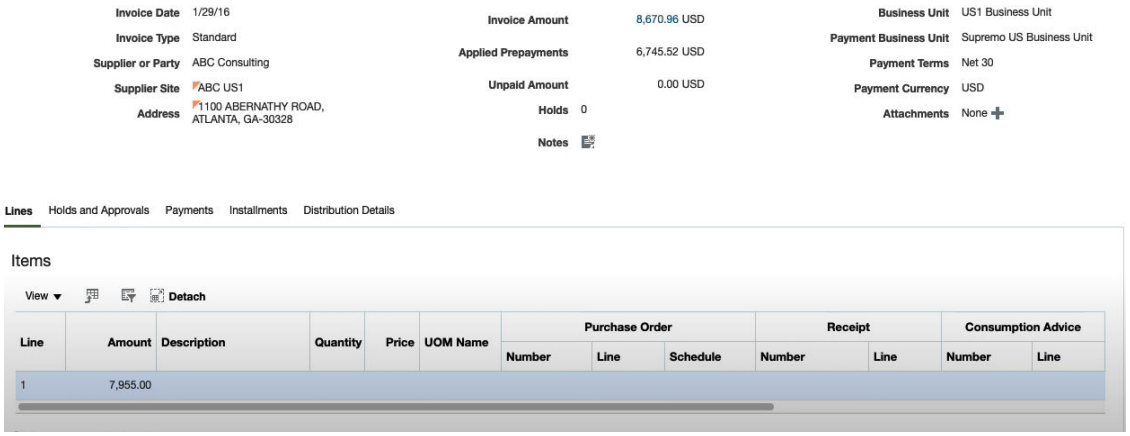
For information about defining a drill region, see Defining Drill Details.

The following examples shows to use a Custom URL to drill through to the source data using a Fusion deep link. Deep Links are URL links that can be distributed outside of Fusion Applications and allow users to open pages without navigating through a menu structure:

Copyright © 2017, 2026, Oracle and/or its affiliates.



When the user selects the Fusion Deep link, the results are shown on a separate tab:



Adding a Custom View to the Drill Through Landing Page

When drilling through to data on the landing page in the Oracle Fusion Cloud Enterprise Performance Management, you can create and select a custom view of columns. This feature enables you to customize the display of the landing page. You can customize the list of the source columns, and order the columns and column titles. The custom view definition can be saved. For subsequent drills, the system uses the last used custom view. If no custom view, the system uses the Source (All) as the default view.

> **Note**
>
> When drilling into Oracle Smart View for Office, Data Integration uses the last used view on the Drill landing page. If no last used view is found, Data Integration uses the default view selection in this setting

To define a custom view for a drill:

1. From the **Home** page, click **Navigator**, and then under **Tasks and Reporting**, select **Data**.
2. Open a form containing loaded source data.

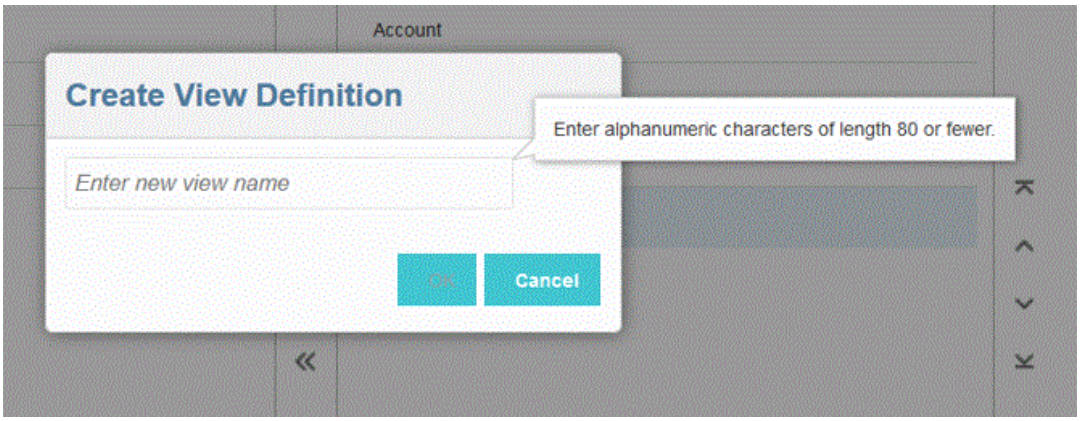
   The icon

   in the right top corner of a cell indicates that it has drill-through data.

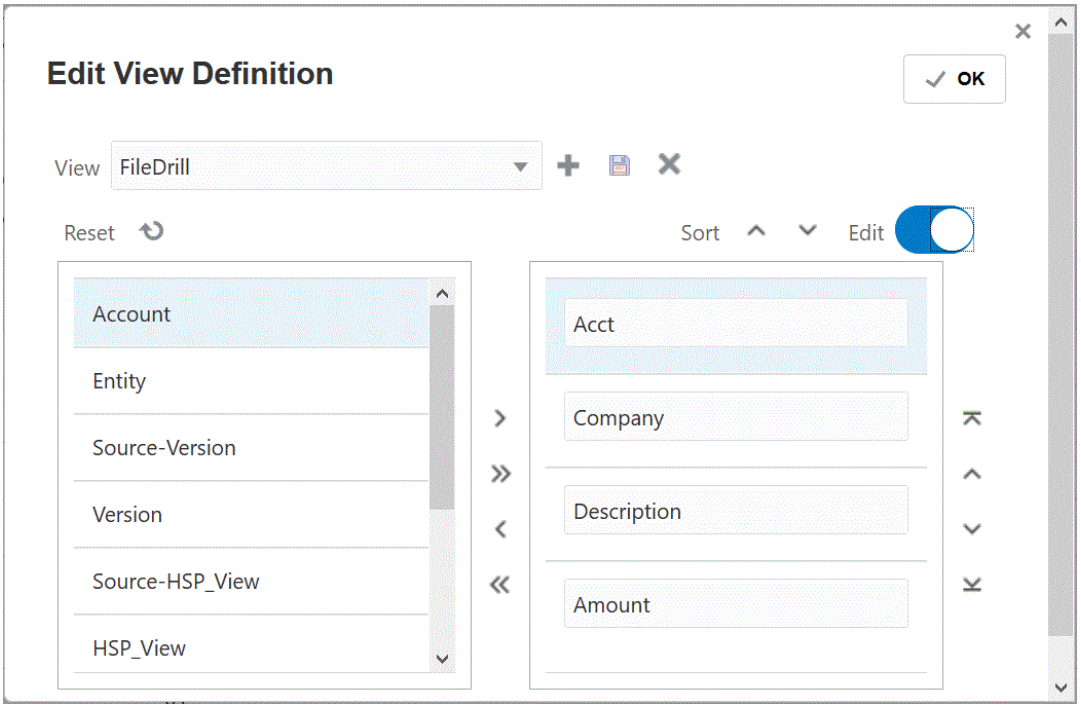
Copyright © 2017, 2026, Oracle and/or its affiliates.

3. From a cell that contains drill-down data, right click the cell, and then select **Drill Through to source**.
4. Select the default view from which to build your custom view definition.
5. Click .
6. From the **Edit View Definition** page, click .
7. In the **Create View Definition** windows, enter the name of the custom view definition and click **OK**.

   The view definition name can contain alphanumeric characters. The view definition name can have up to eighty characters or less.



When the Edit View Definition page is first displayed for the new view, all columns are shown in the (Show Columns) right pane. Move all columns that you do not want to include in the custom view to the (Available Columns) left pane.



8. To include a column in a custom view, select the column from the (Available Columns) left page and then click to move it to the (Show Column) right pane.

Copyright © 2017, 2026, Oracle and/or its affiliates.

To clear the populated values in the view, click **Reset**.

9. To exclude a column in a custom view, select the column from the (Show Columns) right pane and then click to move it to the (Available Column) left pane.

   Additional move options include:

   - To move all columns from the (Available Columns) left pane to the (Show Columns) right pane, click .
   - To move all columns from the (Show Columns) right page to the (Available Columns) left page, click .

10. **Optional**: To rename a column in the (Show Columns) right pane, select the column, then tap the **Edit** slider on, and then type the new column name.
11. **Optional**: To rearrange a column in the (Show Columns) right pane, select the column and then select one of the following:

    - —Move column to the top of the display.
    - —Move column up one position of the display.
    - —Move column down one position of the display.
    - —Move column down to the bottom the display.

12. **Optional**: Click **Sort** to sort columns in the **Shown Columns** pane in ascending or descending order.
13. Click **OK** or to save and update the view definition.
14. **Optional**: To delete a view, click .

Defining Drill Details

In Standard Drill mode you drill from Oracle Fusion Cloud Enterprise Performance Management data form to a landing page and then drill to the source system data. Using this method, you must load data using Data Integration. You must use the Full Workflow mode, which is supported for all business service types. For this mode you can use the auto-created drill region or custom drill region.

Using a Custom Drill Region, You can specify a drill region, which provides an Oracle Essbase drill slice defined using member names and member functions.

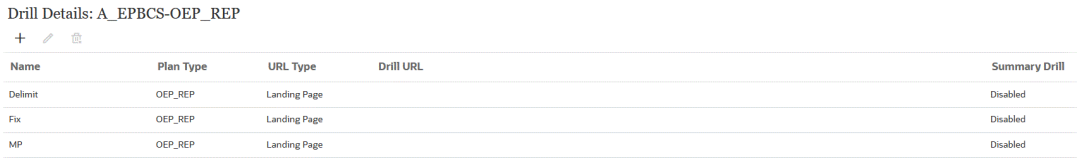
Using Direct Drill, you can drill to source data without staging the data in Data Integration. It also provides the flexibility to simplify the drill bypassing the landing page. Additionally, you can customize the scope of the drill region using Essbase member functions, which simplifies maintenance and improves performance.

With Direct Drill you can load the data using Full Workflow mode, Quick mode or even load directly to the Cloud EPM business process. This mode is not supported for Account Reconciliation and Profitability and Cost Management business processes. For this mode you must define a custom drill region.

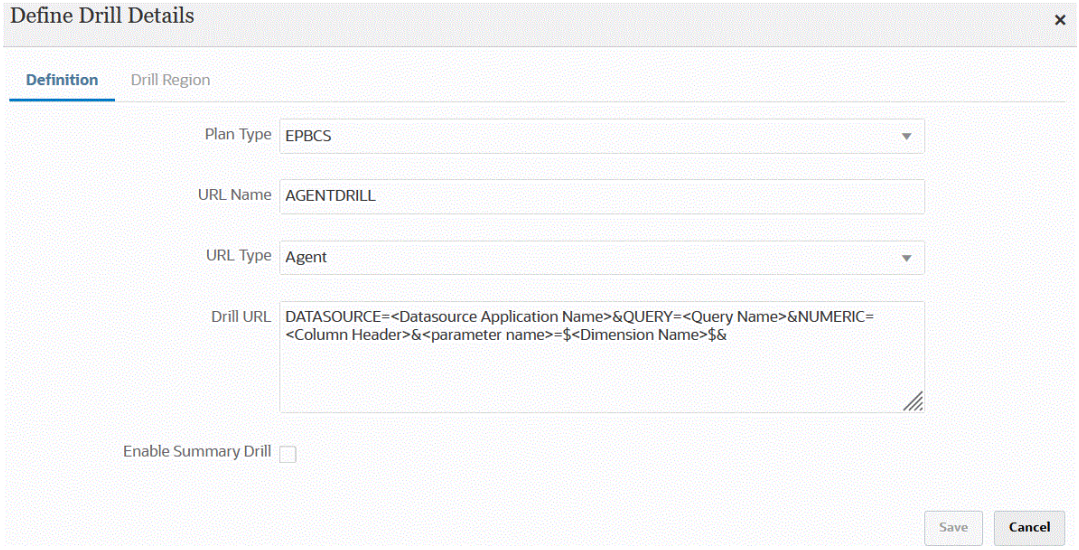
Copyright © 2017, 2026, Oracle and/or its affiliates.

To define a drill definition:

1. From the **Data Integration** home page, then **Actions**, select **Applications**.
2. From the **Application** page, select an application by clicking ••• to the right of a local application, and then select **Drill Details**.



3. From the **Drill Details** page, click + .
4. Select the **Define Drill Detail** page, and then from **Plan Type**, select the source plan type to which you have an integration.



5. In **URL Name**, specify the name of the drill.

   This is the Drill URL Name in Essbase when you open it in Calculation Manager.

   Essbase name restrictions apply to the name of the direct drill including:

   - Use no more than 8 characters when naming a direct drill for a non-Unicode-mode application.
   - Use no more than 30 characters when naming a direct drill Unicode-mode applications.
   - Do not use spaces in the name.
   - Do not use the following characters:

Table 15-4 Essbase Naming Restrictions

Character	Description
*	asterisk
[]	brackets
:	colon
;	semicolon

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 15-4 (Cont.) Essbase Naming Restrictions

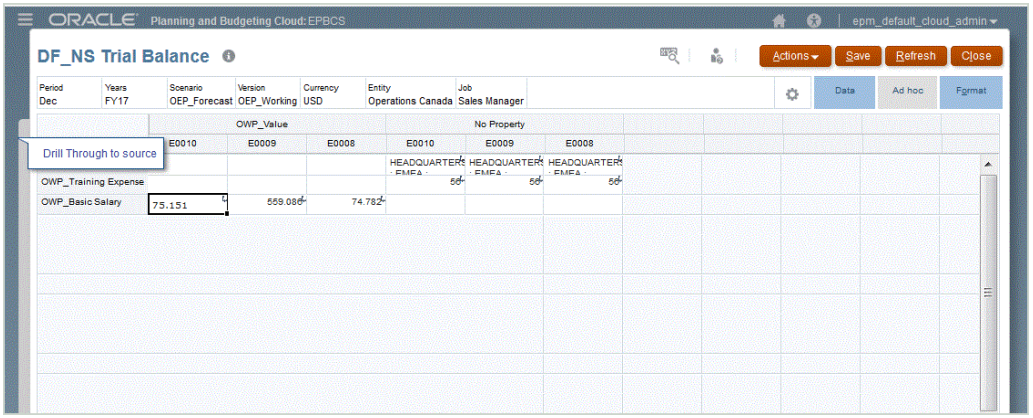
Character	Description
,	comma
=	equal sign
>	greater-than sign
<	less-than sign
.	period
+	plus sign
?	question mark
"	double quotation mark
‘	single quotation mark
/	forward slash
\	backslash
|	vertical bars
	tabs

6. From **URL Type**, select the type of URL to use for the direct drill.

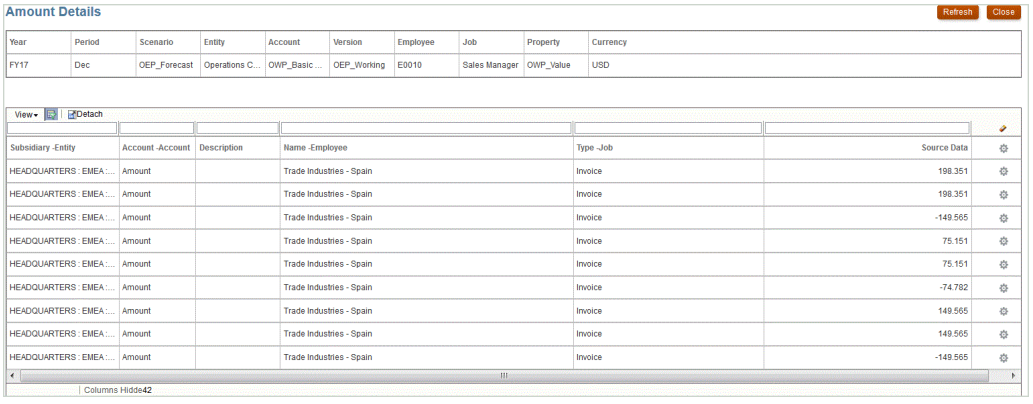
   Available types:

   - **Landing Page**—Have the system automatically launch the landing page. From this landing page, you can open the source document or continue to drill-through to the defined source system landing page as shown below.

     In the following example, drill through is available from a data form in Planning:



From this landing page, you can open the source document or continue to drill through to the defined source system landing page.



Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> Drill through functionality is not supported for exchange rates data loaded to Planning applications.
>
> If you map to an alias when loading data to Essbase, then the drill-through might fail because the filter on the landing page uses the member from Essbase. If this case, use the member name instead of the alias.

- **Custom**—Use a URL associated with a custom direct drill region definition and display the drill through results either in Custom - Dynamic tab or Custom - Browser tab.

  In the **Custom - Dynamic** tab, results are shown in a new tab within the Oracle Fusion Cloud EPM.

  In the **Custom - Browser** tab, results are shown in a new browser tab.

  When you define a custom Drill using this setting for a Plan Type, the default drill creation does not create the drill region for the Plan Type. The create drill region flag is ignored if a custom drill is defined.

  Optional: You can specify a drill region, which provides an Essbase drill slice defined using member names and member functions.

  a. From the **Drill Details** page, select the **Drill Region** tab.

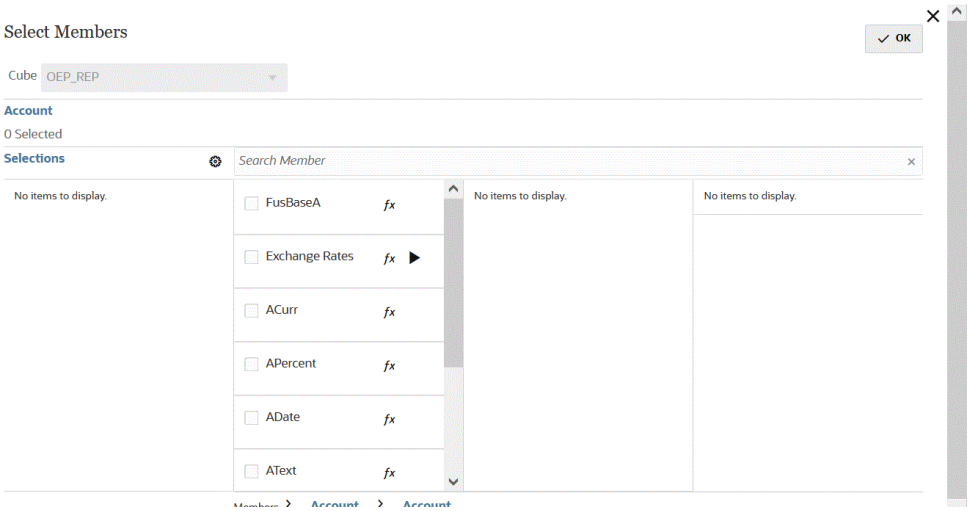
  b. Click + .

  c. From the **Dimension** drop-down, select the dimension(s) to add to the drill region.

  d. From **Filter**, specify a member function for the selected dimension to limit results only to the specified members.

     Specify the member between quotation ("") marks. For example, specify the member July as "Jul".

     Optionally, you can click to display the Member Selector page and navigate to a selected member to drill on. For more information, see Selecting Members from the Member Selector in the *Smart View for Office User's Guide 22.100*.



- **Agent**—Use the parameters based on the SQL query and server information. In this case, the server information comes from the agent but the rest of the URL is derived

Copyright © 2017, 2026, Oracle and/or its affiliates.

from the parameters specified in the SQL query. To use this URL type, the drill URL must include the data source, query, and the parameters to pass.

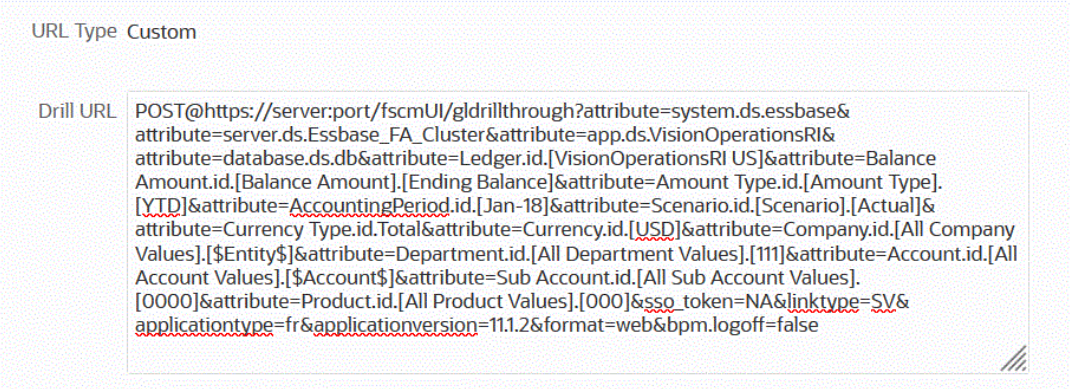
- **OADB**—Use the parameters based on the SQL query and server information. To use this URL type, the drill URL must include the data source, query, and the parameters to pass.

7. In **Drill URL**, specify the URL address for the custom, agent, and OADB direct drill through definitions.

   For a custom URL Type, the drill URL must include the server, port, and URL parameters, for example, `https://server:port/<URL Parameters>`.

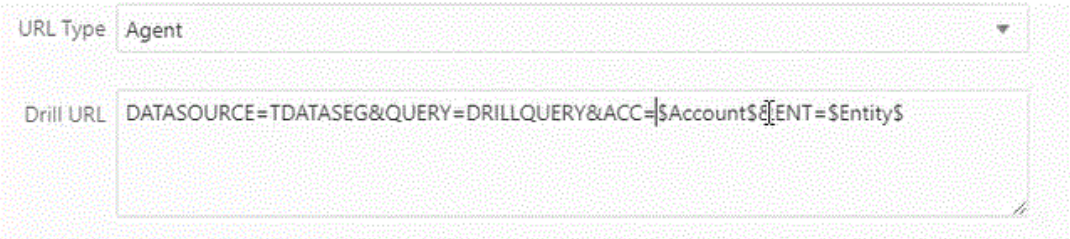
   The following is a custom drill URL. This is a direct drill URL if you want to bypass the landing page:

   > **Note**
   >
   > If a static URL contains non-ascii characters, the non-ascii characters must be encoded in the URL definition.



   For agent and OADB URL Types, the drill URL must include the data source application name, SQL Query, and number column.

   - DATASOURCE—Specify the name of the Data Source Application used to load the data. The source connection information is derived from this application.
   - QUERY—Specify the SQL query to use for performing the drill.
   - NUMERIC—Specify the number column so it is properly aligned.



   For more information about drilling through using the EPM Integration Agent, see Drill through to Source using the EPM Integration Agent.

8. Select the **Enable Summary Drill** option to drill down from summary members in a data form or report and view the detail source data that makes up the number.

Copyright © 2017, 2026, Oracle and/or its affiliates.

After enabling this option and loading the data with the Drill Region option set to **Yes**, the Drill icon is enabled at the summary level. Drill is limited to 1000 descendant members for a dimension.

For a summary drill the 1,000 descendants limit is for each dimension. When there are more than 1,000 descendants, the system behavior is controlled by the **Summary Drill Behavior when more than 1000 descendants** option in Defining Application Detail Options.

A drill from parent with more than 1000 descendants can result inconsistent results. Note the data pattern before allowing users to drill at all levels.

When the option is set to **Ignore**, then the system completely ignores filtering on that dimension. **Filters** are applied only for the remaining dimensions with level 0 members or descendants less than 1,000. As a result, it is possible you will get members not under the parent you are drilling. When using the Ignore option the only time it works is when the ignored dimensions intersect with unique members of the other dimensions.

For example in the following scenario, **Location** and **Account** are leaf level members

Table 15-5 Location and Account Leaf Level Members

Location	Cost Center	Account
US	110	Expense
US	120	Expense
US	130	Expense
UK	210	Expense
UK	210	Expense

In this case Cost Center intersects with unique rows **Location** and **Account**. If you drill on US, Total Cost Center, and Expense and the option is set to ignore the Cost Center, then the drill returns 3 rows of US.

If on the other hand, lets assume **Location** is a parent with more than 1000 children:

Table 15-6 Cost Center Intersect with Unique Location

Location	Cost Center	Account	Project
US	110	Expense	ABC
US	120	Expense	ABC
US	130	Expense	XYZ
UK	210	Expense	ABC
Uk	210	Expse	XYZ

If you drill on US, Total Cost Center, and Expense, the ABC system ignores **Location** and **Country** and returns all rows that match **Expense** and **ABC**. This results in rows from US and UK.

When enabling drill from Parent, it is recommended to enable drill only from parents with less than 1000 descendants. This can be done by defining drill region manually using Application Drill Details page.

When you select Limit Option, it is possible that all the children do not fall in the 1000 limit resulting in an incomplete result set.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> If you select to perform a summary drill on more than 1,000 descendants and include only top-level dimensions but the results are the same as the base-level dimension, then also select base-levels intersections for at least three dimensions.

> **Note**
>
> In order to support a summary drill, your drill query should not include an operator in the WHERE clause of the SQL. For example, your query should be: WHERE COMPANY ~ENTITY~. The system determines appropriate condition (IN, LIKE) based on the number of descendants.

For more information the drill region, see Defining Drill Regions .

9. Click **Save**.

   When you save the drill definition, drill regions are automatically created or updated in Essbase. When you delete the Drill Details, they are also deleted in Essbase.

10. Specify a custom drill region on the Drill Region tab if desired.

11. **Optional**: To edit an existing direct drill definition, from the **Drill Details** page, select the definition and click .

    To delete an existing direct drill definition, from the **Drill Details** page, select the definition and click .

Defining Direct Drill

Direct Drill enables you to drill through directly from any platform-based form to source data, which can be queried and launched using an external URL or an EPM Integration Agent query. Additionally, for standard mode data loads, you can define a custom drill region and continue to use the standard landing page. The custom drill region enables you to define custom drill regions using Oracle Essbase member functions instead of individual members.

Using Direct Drill, you can drill to source data without staging the data in Data Integration. It also provides the flexibility to simplify the drill bypassing the landing page. Additionally, you can customize the scope of the drill region using Essbase member functions, which simplifies maintenance and improves performance.

Drill through to the Oracle ERP Cloud

Drill down to the Oracle General Ledger account balance summary page in Oracle ERP Cloud,

Drill though enables you to display and view the Oracle General Ledger account balance summary page in the Oracle ERP Cloud.

When integrating with the Oracle ERP Cloud, Data Integration determines the drill URL definition based on the connection information automatically such as system and fixed information. You do not need to set up when drilling through to the Oracle ERP Cloud.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> Drill through to the Oracle General Ledger does not support use of multi-byte characters in GL Segment values, Ledger Name, and Period Name.

> **Note**
>
> For information about drilling down from Oracle Transactional Business Intelligence (OTBI) ad hoc reports to transaction details in Oracle Fusion Applications, see Drill Down To Fusion Financial Applications from OTBI ERP Reports.

Drill through to a BI Publisher Report in the Oracle ERP Cloud

When integrating data from the Oracle Business Intelligence Publisher Reporting in the Oracle ERP Cloud, you can drill down to the source associated with account balance data loaded to the Oracle Fusion Cloud Enterprise Performance Management.

Drilling Down to a BI Publisher Report in the Oracle ERP Cloud Process Description

Drill down to the Oracle Business Intelligence Publisher BI Publisher Reporting source associated with account balance data loaded to the Oracle Fusion Cloud Enterprise Performance Management from Oracle ERP Cloud.

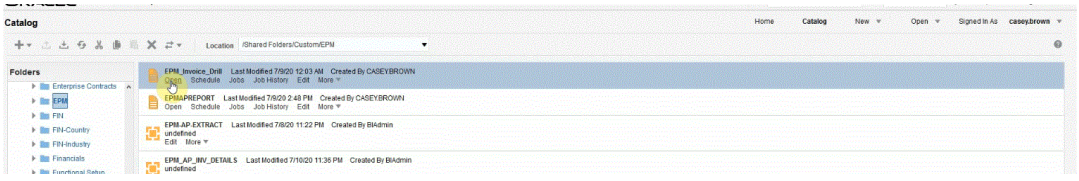
The following is a high-level description on how to drill down to a BI Publisher report from the Cloud EPM

1. Define the BI Publisher report that you want to use as a drill target. This includes the query extract, data model, and any required parameters.

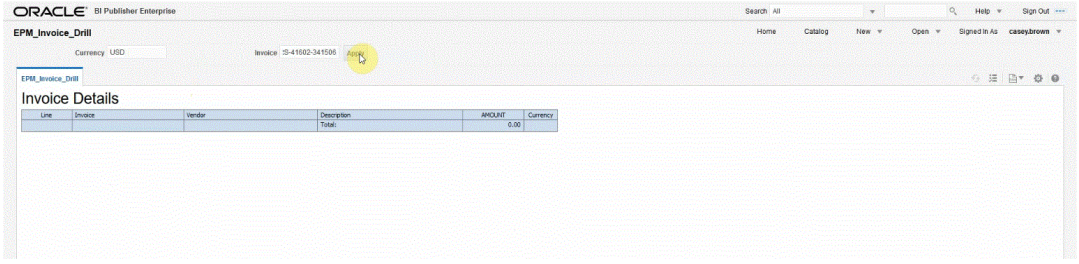
   For more information, see:

   - Defining the Data Extract Process
   - Creating the Data Model for the BI Publisher Report Extract

2. Create a drill down BI Publisher report with the desired parameters based on the original BI Publisher report.

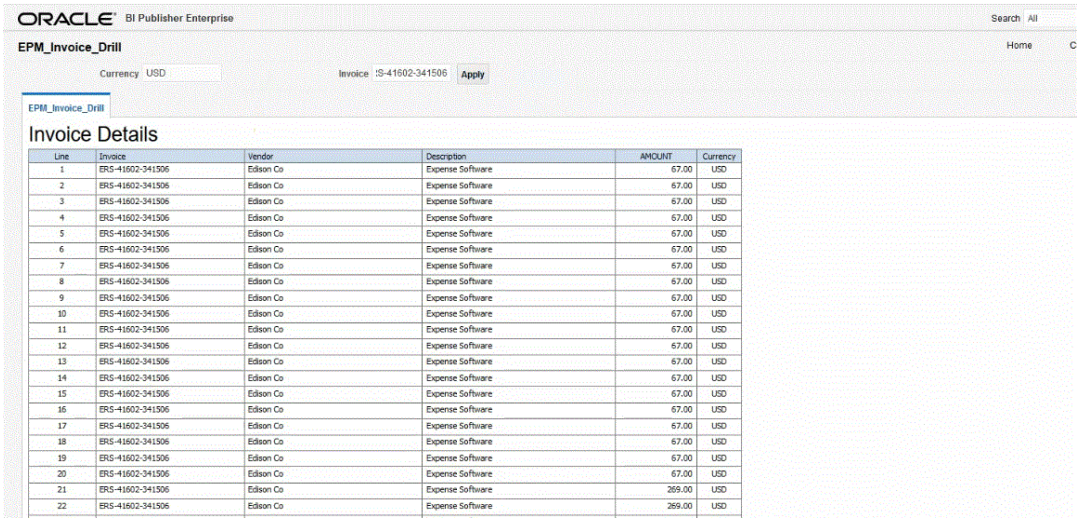


For example, you could include "Currency" and "Invoice".



Copyright © 2017, 2026, Oracle and/or its affiliates.

The results show an Invoice Detail reported used for drill-through from the Cloud EPM using the parameters for currency and invoice detail:

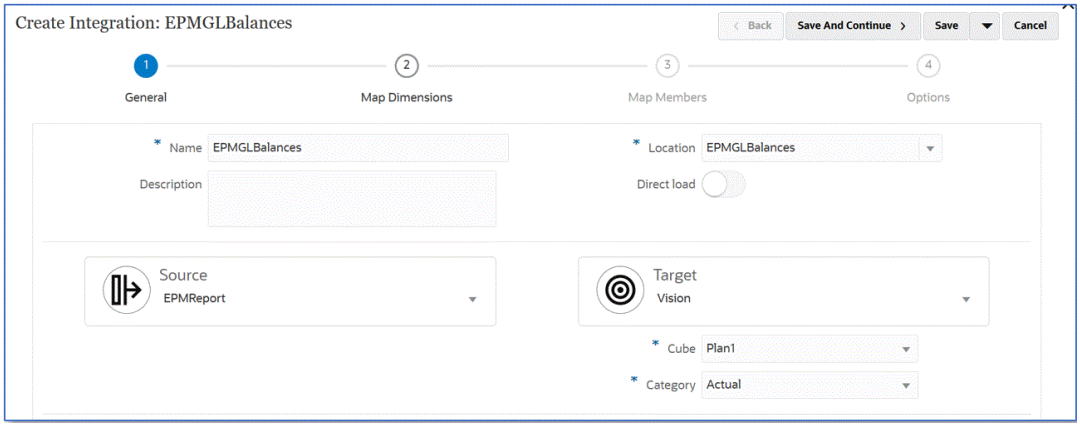


Line	Invoice	Vendor	Description	AMOUNT	Currency
1	ERS-41602-341506	Edison Co	Expense Software	67.00	USD
2	ERS-41602-341506	Edison Co	Expense Software	67.00	USD
3	ERS-41602-341506	Edison Co	Expense Software	67.00	USD
4	ERS-41602-341506	Edison Co	Expense Software	67.00	USD
5	ERS-41602-341506	Edison Co	Expense Software	67.00	USD
6	ERS-41602-341506	Edison Co	Expense Software	67.00	USD
7	ERS-41602-341506	Edison Co	Expense Software	67.00	USD
8	ERS-41602-341506	Edison Co	Expense Software	67.00	USD
9	ERS-41602-341506	Edison Co	Expense Software	67.00	USD
10	ERS-41602-341506	Edison Co	Expense Software	67.00	USD
11	ERS-41602-341506	Edison Co	Expense Software	67.00	USD
12	ERS-41602-341506	Edison Co	Expense Software	67.00	USD
13	ERS-41602-341506	Edison Co	Expense Software	67.00	USD
14	ERS-41602-341506	Edison Co	Expense Software	67.00	USD
15	ERS-41602-341506	Edison Co	Expense Software	67.00	USD
16	ERS-41602-341506	Edison Co	Expense Software	67.00	USD
17	ERS-41602-341506	Edison Co	Expense Software	67.00	USD
18	ERS-41602-341506	Edison Co	Expense Software	67.00	USD
19	ERS-41602-341506	Edison Co	Expense Software	67.00	USD
20	ERS-41602-341506	Edison Co	Expense Software	67.00	USD
21	ERS-41602-341506	Edison Co	Expense Software	269.00	USD
22	ERS-41602-341506	Edison Co	Expense Software	269.00	USD

3. In Data Integration, create the application that uses the BI Publisher report data source.

   For more information, see Registering Oracle ERP Cloud Applications.

4. Create a new integration and select the data source application as the source and the target application. Then complete the remaining steps to create the integration.



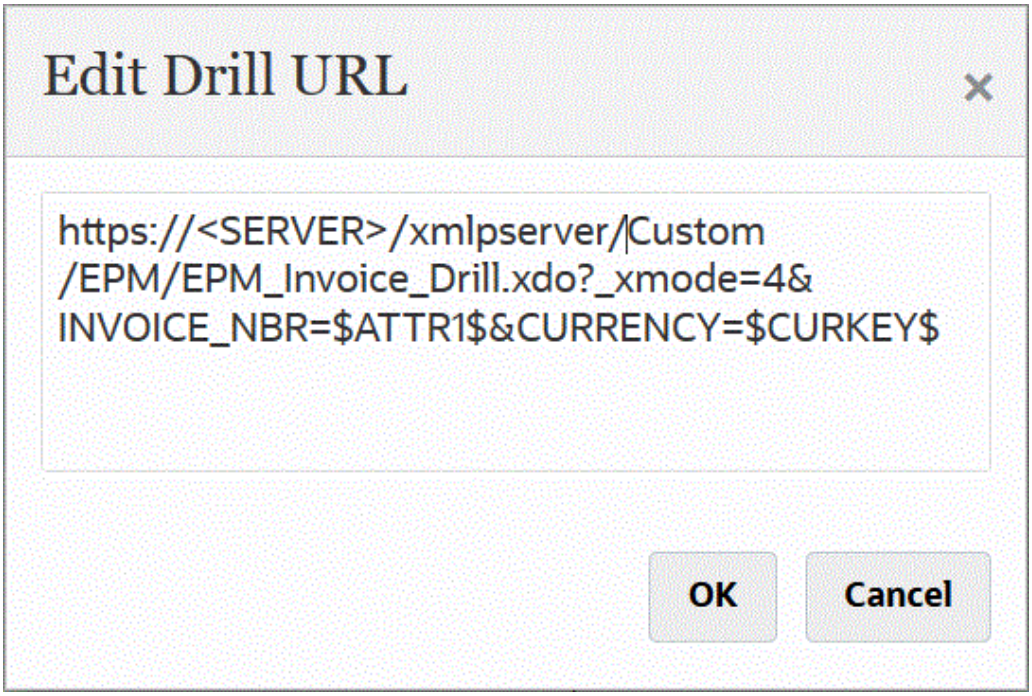
   a. Set up the integration mapping between the Oracle ERP Cloud data source and the target application by building an import format, location, and dimension mappings.

      See Mapping Dimensions.

   b. On the **Map Dimensions** page, then **Drill URL**, enter the parameters for the drill through report.

      These parameters are referenced from the Workbench making the drill through dynamic.

Copyright © 2017, 2026, Oracle and/or its affiliates.



> **Note**
>
> You can also drill through to ERP BIP Report using a custom drill region. To do this, specify the drill URL on the Drill Details page (select the application from the **Applications** page, then from **Actions**, select **Drill Detail**). In this case, select a Custom URL type and the Drill URL where you specify the dimension name instead of TDATASEG Table column name. For example, https://%3CServer%3E/xmlpserver/Custom/EPM/EPM_Invoice_Drill.xdo?-xmode=4&INVOICE_NVR=$Customer$&CURRENCY=$Currency$ Note that dimension names are case sensitive. For example, `&P_COMPANY=$ENTITY$` should be mentioned as `&P_COMPANY=$Entity$`

c. Map members from the source to target.
See Mapping Members.

d. Select any source and target options.

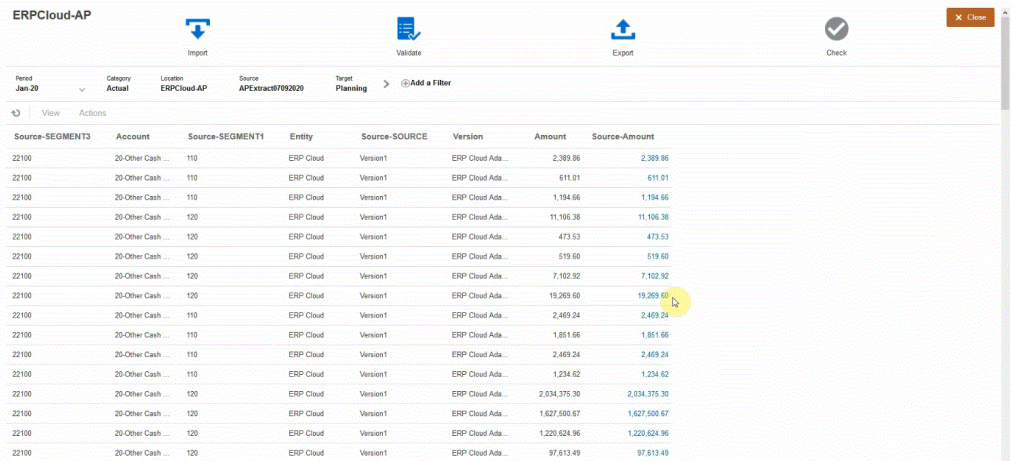
See Setting Data Integration Options.

e. Run the integration.

See Running an Integration.

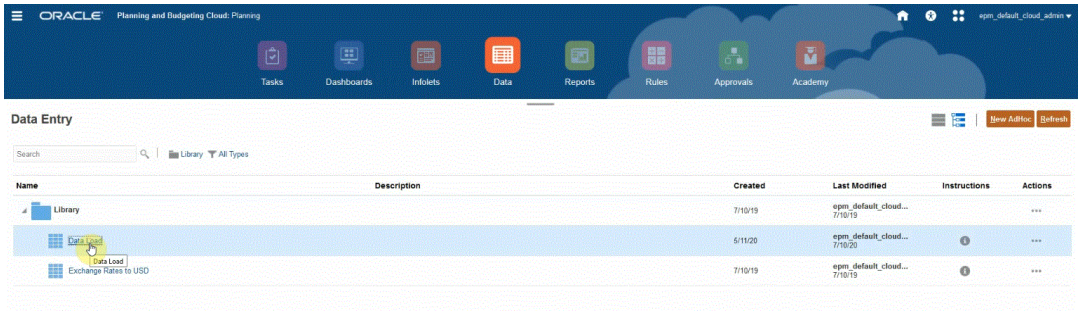
After running the integration, the source data is shown in the Workbench.

Copyright © 2017, 2026, Oracle and/or its affiliates.

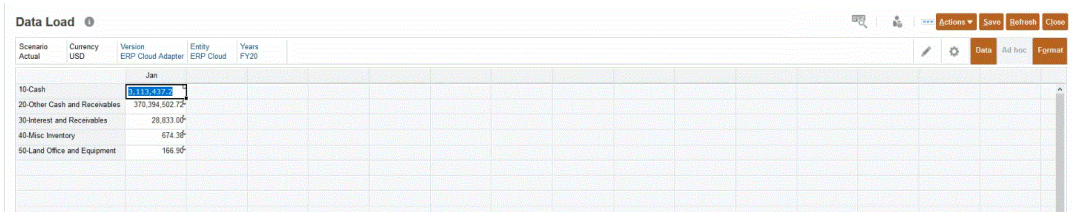


5. From the **Home** page, click **Navigator**, and then under **Tasks and Reporting**, select **Data** (Data).

6. From the **Data Entry** page, under **Library**, open a form containing loaded source data.



7. Select a cell that contains drill-down data.

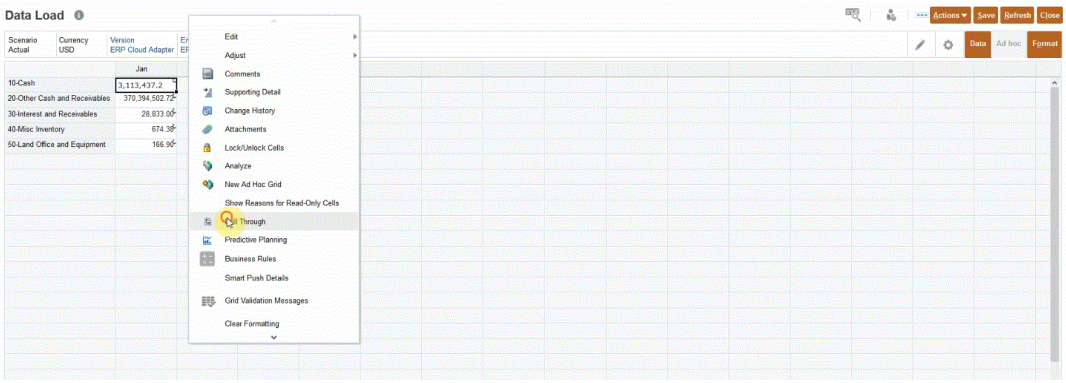


The icon

in the right top corner of a cell indicates that it has drill-through data.

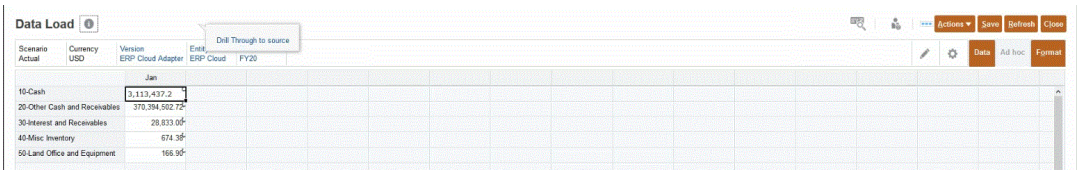
8. Right click the cell that contains drill-down data and select **Drill Through**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

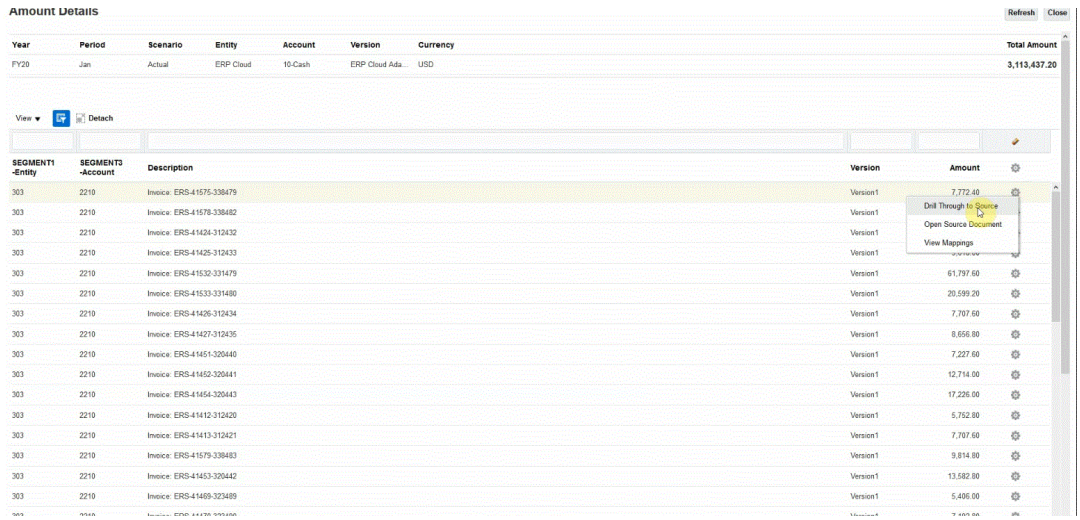


By selecting **Drill Through**, you can drill from a data form in the Cloud EPM to the BI Publisher report in the Oracle ERP Cloud.

9. On the **Data Load** page, click the **Drill Through to source** link.

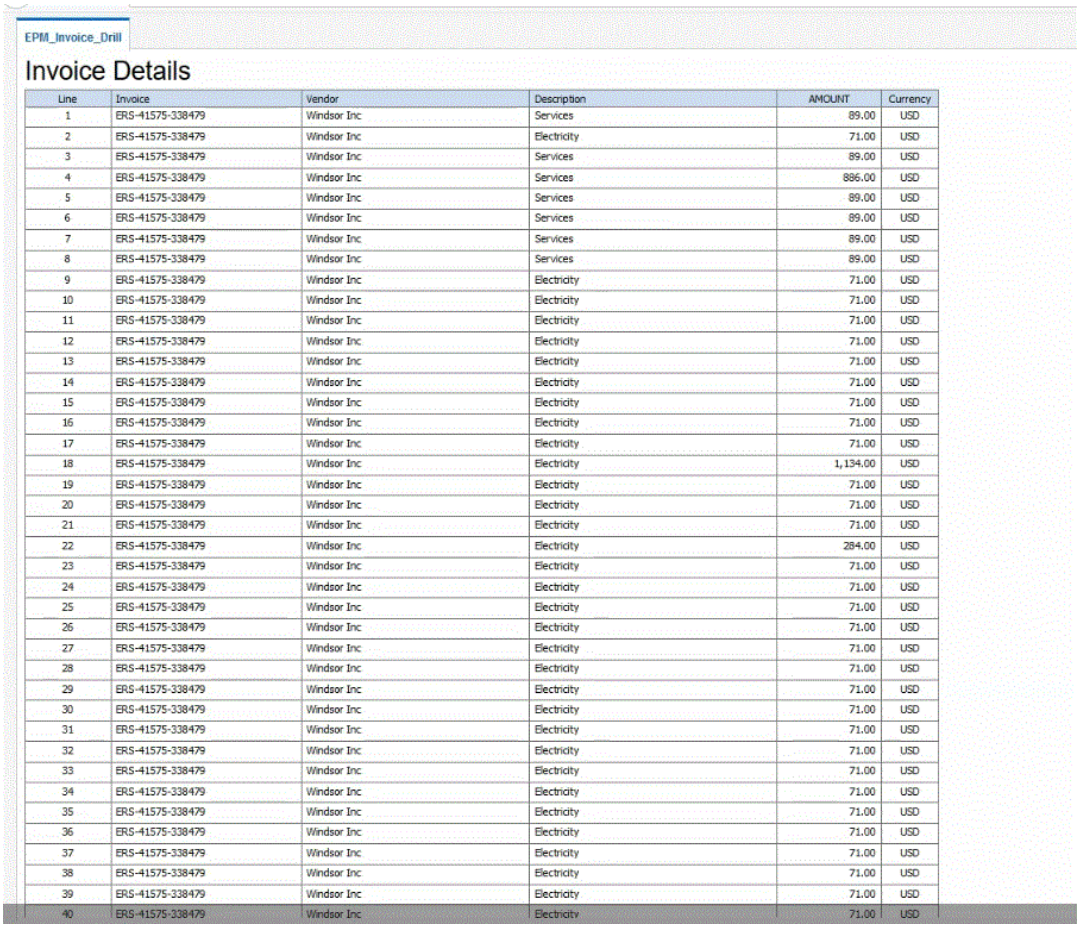


10. On the **Amount Details** page, right click the amount and then select **Drill Through to source**.



11. On the **Details** page, review the results of the drill through.

Copyright © 2017, 2026, Oracle and/or its affiliates.



EPM_Invoice_Drill

Invoice Details

Line	Invoice	Vendor	Description	AMOUNT	Currency
1	ERS-41575-338479	Windsor Inc	Services	89.00	USD
2	ERS-41575-338479	Windsor Inc	Electricity	71.00	USD
3	ERS-41575-338479	Windsor Inc	Services	89.00	USD
4	ERS-41575-338479	Windsor Inc	Services	886.00	USD
5	ERS-41575-338479	Windsor Inc	Services	89.00	USD
6	ERS-41575-338479	Windsor Inc	Services	89.00	USD
7	ERS-41575-338479	Windsor Inc	Services	89.00	USD
8	ERS-41575-338479	Windsor Inc	Services	89.00	USD
9	ERS-41575-338479	Windsor Inc	Electricity	71.00	USD
10	ERS-41575-338479	Windsor Inc	Electricity	71.00	USD
11	ERS-41575-338479	Windsor Inc	Electricity	71.00	USD
12	ERS-41575-338479	Windsor Inc	Electricity	71.00	USD
13	ERS-41575-338479	Windsor Inc	Electricity	71.00	USD
14	ERS-41575-338479	Windsor Inc	Electricity	71.00	USD
15	ERS-41575-338479	Windsor Inc	Electricity	71.00	USD
16	ERS-41575-338479	Windsor Inc	Electricity	71.00	USD
17	ERS-41575-338479	Windsor Inc	Electricity	71.00	USD
18	ERS-41575-338479	Windsor Inc	Electricity	1,134.00	USD
19	ERS-41575-338479	Windsor Inc	Electricity	71.00	USD
20	ERS-41575-338479	Windsor Inc	Electricity	71.00	USD
21	ERS-41575-338479	Windsor Inc	Electricity	71.00	USD
22	ERS-41575-338479	Windsor Inc	Electricity	284.00	USD
23	ERS-41575-338479	Windsor Inc	Electricity	71.00	USD
24	ERS-41575-338479	Windsor Inc	Electricity	71.00	USD
25	ERS-41575-338479	Windsor Inc	Electricity	71.00	USD
26	ERS-41575-338479	Windsor Inc	Electricity	71.00	USD
27	ERS-41575-338479	Windsor Inc	Electricity	71.00	USD
28	ERS-41575-338479	Windsor Inc	Electricity	71.00	USD
29	ERS-41575-338479	Windsor Inc	Electricity	71.00	USD
30	ERS-41575-338479	Windsor Inc	Electricity	71.00	USD
31	ERS-41575-338479	Windsor Inc	Electricity	71.00	USD
32	ERS-41575-338479	Windsor Inc	Electricity	71.00	USD
33	ERS-41575-338479	Windsor Inc	Electricity	71.00	USD
34	ERS-41575-338479	Windsor Inc	Electricity	71.00	USD
35	ERS-41575-338479	Windsor Inc	Electricity	71.00	USD
36	ERS-41575-338479	Windsor Inc	Electricity	71.00	USD
37	ERS-41575-338479	Windsor Inc	Electricity	71.00	USD
38	ERS-41575-338479	Windsor Inc	Electricity	71.00	USD
39	ERS-41575-338479	Windsor Inc	Electricity	71.00	USD
40	ERS-41575-338479	Windsor Inc	Electricity	71.00	USD

Optionally, you can scroll down to the bottom of the Details page to review the detail that comprise the balance in the Cloud EPM.

Copyright © 2017, 2026, Oracle and/or its affiliates.

EPM_Invoice_Drill

			creation date of 2020-04-12.		
76	ERS-41575-338479	Windsor Inc	The receipt invoice ERS-41575-338479 has a creation date of 2020-04-12.	14.20	USD
77	ERS-41575-338479	Windsor Inc	The receipt invoice ERS-41575-338479 has a creation date of 2020-04-12.	14.20	USD
78	ERS-41575-338479	Windsor Inc	The receipt invoice ERS-41575-338479 has a creation date of 2020-04-12.	14.20	USD
79	ERS-41575-338479	Windsor Inc	The receipt invoice ERS-41575-338479 has a creation date of 2020-04-12.	14.20	USD
80	ERS-41575-338479	Windsor Inc	The receipt invoice ERS-41575-338479 has a creation date of 2020-04-12.	14.20	USD
81	ERS-41575-338479	Windsor Inc	The receipt invoice ERS-41575-338479 has a creation date of 2020-04-12.	14.20	USD
82	ERS-41575-338479	Windsor Inc	The receipt invoice ERS-41575-338479 has a creation date of 2020-04-12.	14.20	USD
83	ERS-41575-338479	Windsor Inc	The receipt invoice ERS-41575-338479 has a creation date of 2020-04-12.	14.20	USD
84	ERS-41575-338479	Windsor Inc	The receipt invoice ERS-41575-338479 has a creation date of 2020-04-12.	14.20	USD
85	ERS-41575-338479	Windsor Inc	The receipt invoice ERS-41575-338479 has a creation date of 2020-04-12.	14.20	USD
86	ERS-41575-338479	Windsor Inc	The receipt invoice ERS-41575-338479 has a creation date of 2020-04-12.	14.20	USD
87	ERS-41575-338479	Windsor Inc	The receipt invoice ERS-41575-338479 has a creation date of 2020-04-12.	14.20	USD
88	ERS-41575-338479	Windsor Inc	The receipt invoice ERS-41575-338479 has a creation date of 2020-04-12.	14.20	USD
89	ERS-41575-338479	Windsor Inc	The receipt invoice ERS-41575-338479 has a creation date of 2020-04-12.	14.20	USD
90	ERS-41575-338479	Windsor Inc	The receipt invoice ERS-41575-338479 has a creation date of 2020-04-12.	14.20	USD
91	ERS-41575-338479	Windsor Inc	The receipt invoice ERS-41575-338479 has a creation date of 2020-04-12.	14.20	USD
92	ERS-41575-338479	Windsor Inc	The receipt invoice ERS-41575-338479 has a creation date of 2020-04-12.	14.20	USD
93	ERS-41575-338479	Windsor Inc	The receipt invoice ERS-41575-338479 has a creation date of 2020-04-12.	14.20	USD
94	ERS-41575-338479	Windsor Inc	The receipt invoice ERS-41575-338479 has a creation date of 2020-04-12.	14.20	USD
95	ERS-41575-338479	Windsor Inc	The receipt invoice ERS-41575-338479 has a creation date of 2020-04-12.	14.20	USD
96	ERS-41575-338479	Windsor Inc	The receipt invoice ERS-41575-338479 has a creation date of 2020-04-12.	14.20	USD
97	ERS-41575-338479	Windsor Inc	The receipt invoice ERS-41575-338479 has a creation date of 2020-04-12.	14.20	USD
98	ERS-41575-338479	Windsor Inc	The receipt invoice ERS-41575-338479 has a creation date of 2020-04-12.	14.20	USD
99	ERS-41575-338479	Windsor Inc	The receipt invoice ERS-41575-338479 has a creation date of 2020-04-12.	56.80	USD
100	ERS-41575-338479	Windsor Inc	The receipt invoice ERS-41575-338479 has a creation date of 2020-04-12.	56.80	USD
101	ERS-41575-338479	Windsor Inc	The receipt invoice ERS-41575-338479 has a creation date of 2020-04-12.	28.40	USD
102	ERS-41575-338479	Windsor Inc	The receipt invoice ERS-41575-338479 has a creation date of 2020-04-12.	28.40	USD
103	ERS-41575-338479	Windsor Inc	The receipt invoice ERS-41575-338479 has a creation date of 2020-04-12.	14.20	USD
104	ERS-41575-338479	Windsor Inc	The receipt invoice ERS-41575-338479 has a creation date of 2020-04-12.	14.20	USD
			Total:	7,772.40	

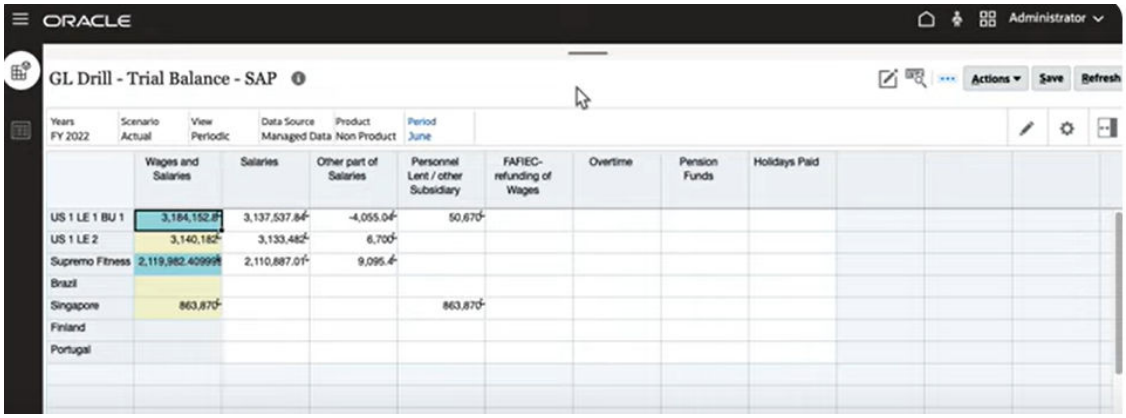
Drill through to the SAP® General Ledger in Cloud EPM

After loading data into your Oracle Fusion Cloud Enterprise Performance Management from your source SAP General Ledger using Data Integration, you can drill through the source SAP system using a URL with parameters. Required parameters not loaded to the target application may be loaded into the workbench for reference when drilling. Based on how you defined drill details, you can drill through to the Data Integration landing page, to a custom URL like a deep link, or drill directly using the EPM Integration Agent.

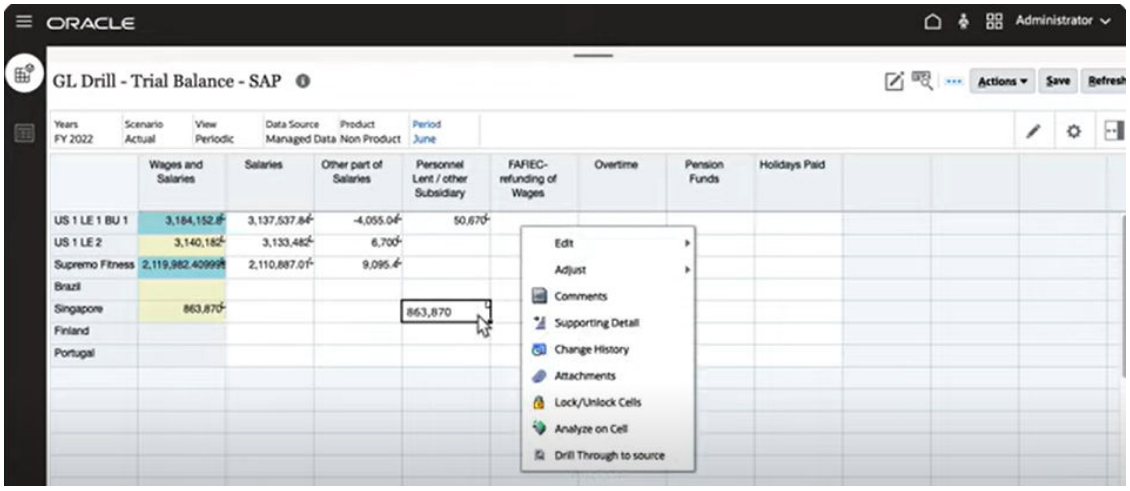
Drill through is activated after you define the drill region. For more information, see: Defining Drill Regions.

In the example below, you have a form where trial balance data from the general ledger for several entities, which was loaded into the Singapore entity using Data Integration data from an SAP GL:

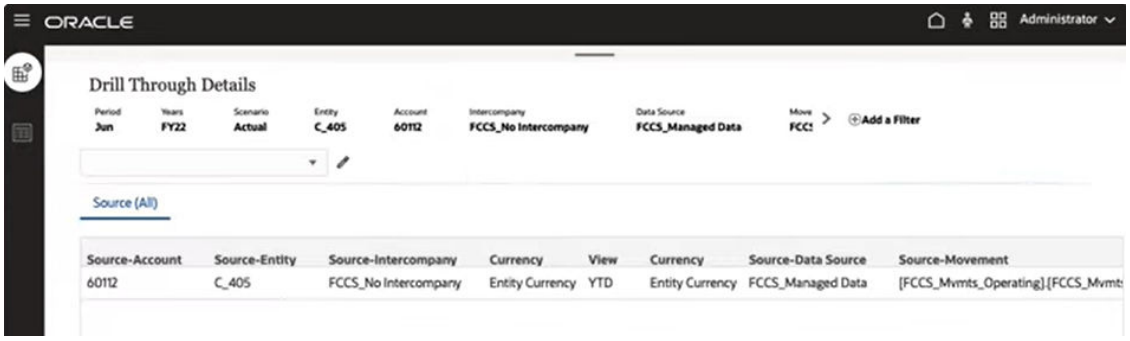
Copyright © 2017, 2026, Oracle and/or its affiliates.



To drill through to the source data in the form, right-click a drillable cell and select Drill Through to source.



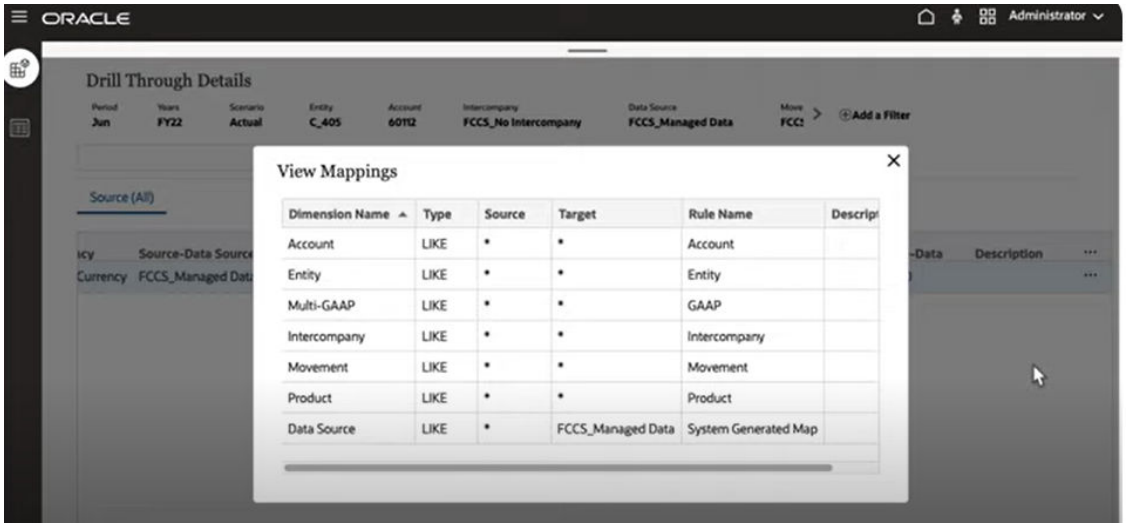
The landing page shows the details that were loaded to the target Cloud EPM application.



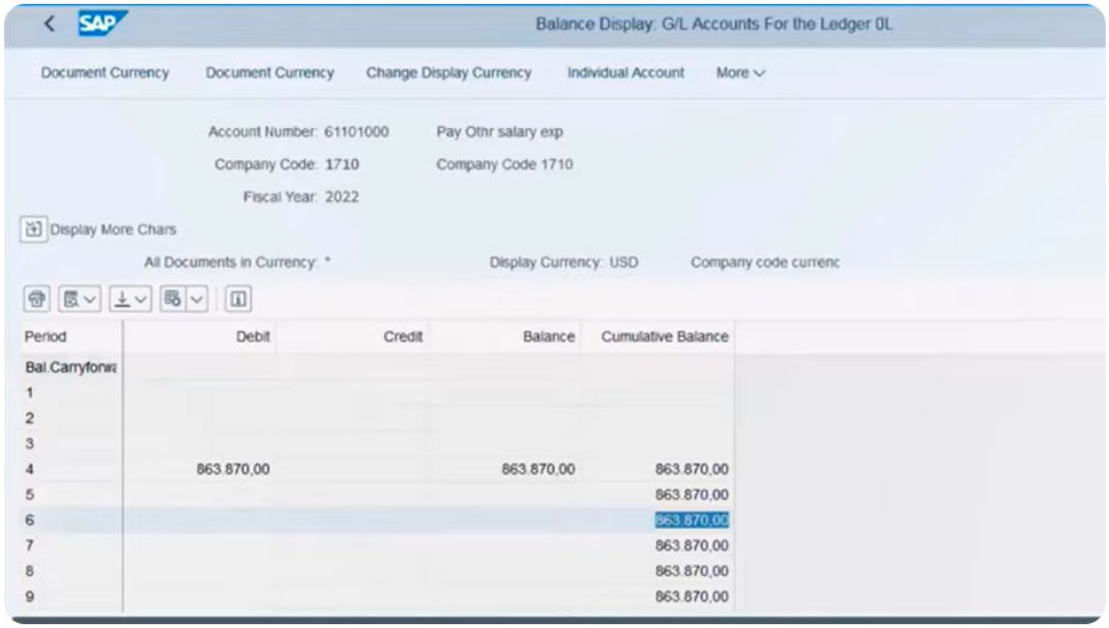
The system displays the data row that comprises the amount from the selected cell. View mappings to see how data was mapped to the dimensions from the source GL across the chart of account segments.

You can view mappings to see how data was mapped to the dimensions from the source GL across the chart of account segments.

Copyright © 2017, 2026, Oracle and/or its affiliates.



From here, you can drill through and open the source system. On the SAP General Ledger page, you can see the account details and validate the numbers displayed.

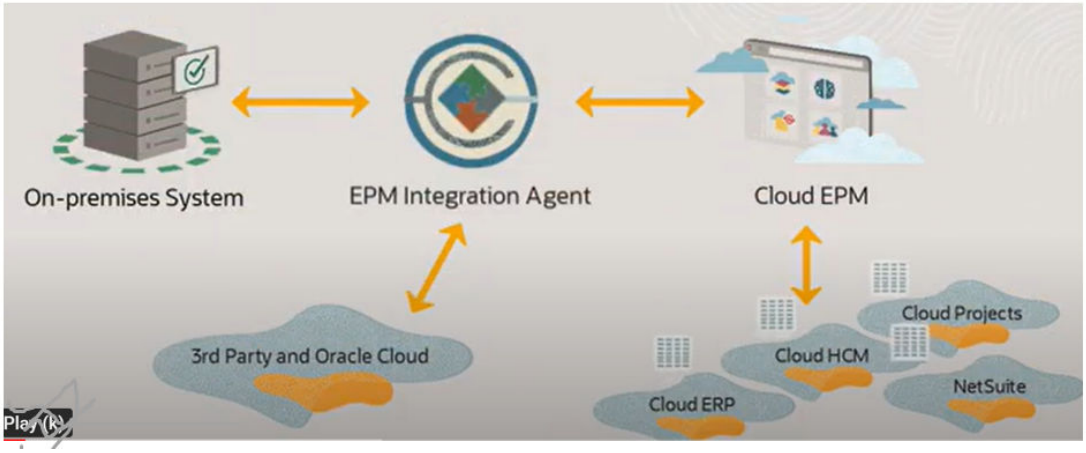


For additional information, see: Drill-Through to SAP.

Drill through to Source using the EPM Integration Agent

When integrating to an on-premises data source, you can drill down to the source to view the details associated with account balance data loaded to the Oracle Fusion Cloud Enterprise Performance Management.

Copyright © 2017, 2026, Oracle and/or its affiliates.



You can perform two types of drill depending on the availability of a landing page:

- If the source system provides a landing page that displays the detail source data, you access the landing page using an http URL and passing contextual parameters to this page. To use this method, you define a http drill URL similar to other source integrations on the Map Dimensions page and use the substitution variables to pass the URL parameters. .
- If the source system does not provide a landing page to view the detail source data, execute an SQL query in the source database and display the results in a pop-up window inside Data Integration using the EPM Integration Agent. To use this method, you need a web server running in https mode. The web server accepts the drill request and routes it to the agent. For more information, see Creating a Drill Down to a Source System Without a Landing Page.

Additional Considerations:

- Available for SYNC and ASYNC modes
- HTTPS server required
- If inside network, an open is port not required
- Oracle Smart View for Office drill inside network doesn't require an HTTPS server.
- drill from public internet outside of network does require a public facing HTTPS server/port
- Configure re-write rules to redirect drill request to Agent URL.
- Test with `https://<server>/epmagent/rest/details`

Drill through to source using the EPM Integration Agent is supported irrespective of how the EPM Integration Agent is configured for the asynchronous or synchronous mode option.

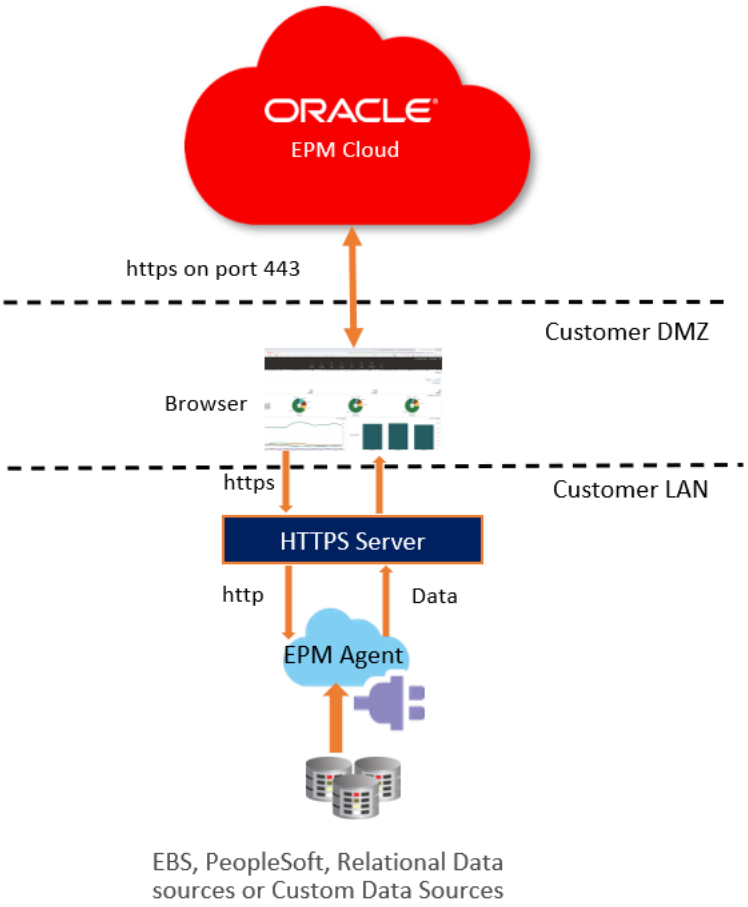
When using web forms and you drill through to source using the EPM Integration Agent, the browser invokes the agent passing the query and other parameters using the HTTP protocol. The agent processes the query request and returns the data to the browser to be displayed to the user. The browser session to the Cloud EPM uses the HTTPS protocol so the communication from the browser to the agent uses HTTPS.

Browsers do not allow mixed HTTP and HTTPS connections in the same session. In order to support drill, perform the following configuration steps:

1. Install and configure an HTTPS web sever inside the on-premises network.
2. Configure the rewrite rules in the web server to redirect the request for drill to the Agent URL (same as in the synchronous agent definition. See Configuring Synchronous Mode).
3. Configure the Web URL of the agent to point to the HTTPS web server.

Copyright © 2017, 2026, Oracle and/or its affiliates.

4. Optional: To test your agent drill-through setup, use the following URL to make sure that your webserver re-write rules are defined correctly: `https://server/epmagent/rest/details`.
   When testing the setup, replace "server" with the web url that is defined in your agent setup.



Note

- In the preceding configuration, the drill is supported only when the client computer running the browser is inside the on-premises network. If you want the drill to be available even when the browser is not in an internal network, enable the HTTPS web server to be publicly accessible.
- When using Smart View and you perform a drill through to source, the additional HTTPS web server configuration is not required. In Smart View, set the drill through launch to open in a new sheet.

Creating a Drill Down to a Source System Without a Landing Page

When you want to drill down to a source system without a landing page, execute a SQL query in the source database and display the results in a pop-up window inside Data Integration.

To create a query for the drill through:

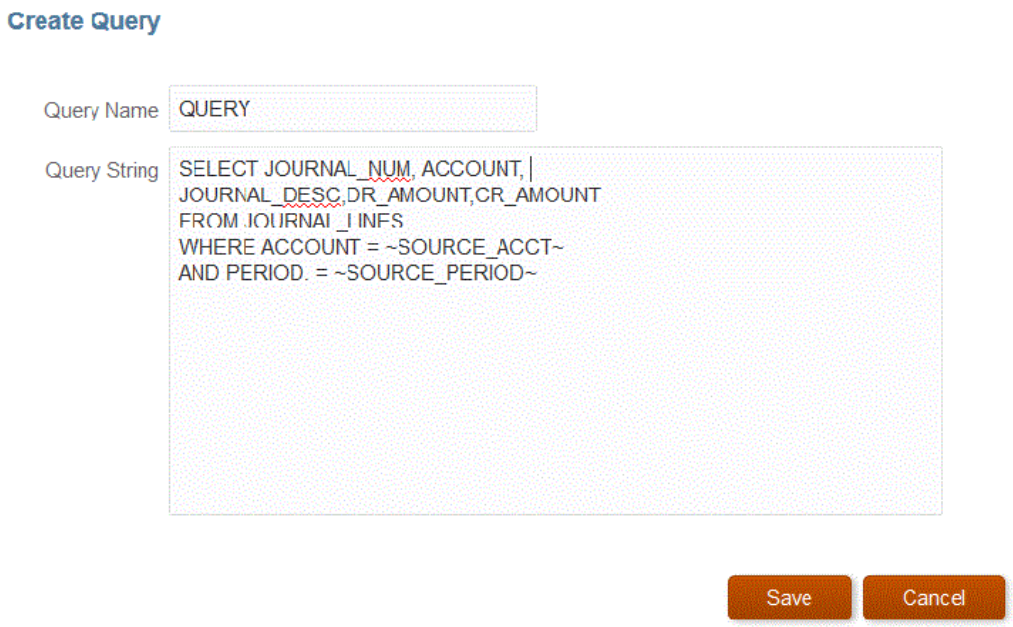
Copyright © 2017, 2026, Oracle and/or its affiliates.

1. From the **Data Integration** home page, then from the **Actions** menu, then from the **Setup** actions drop-down select **Query**.
2. On the **Query** page, click ➕
3. From the **Create Query** page, and then in **Query Name**, specify the name of the query to query the data from the source system.
4. In **Query String**, specify the SQL statement used to extract the data based on the header row in the file.

   You can use an alias for a column name in the SQL query to the dimension name.

   When specifying the string for a header row value and the string is mixed case, enclose the string in double quotation marks. When a string with mixed case characters is not enclosed within double quotation marks, it is resolved in uppercase.

   The query can include filter conditions. The bind variable for the filter condition must be enclosed inside ~~ characters.



5. Click **Save**.
6. From the **Data Integration** home page, click ⋯ to the right of the integration to which to add a drill through, and then select **Map Dimensions**.
7. On the **Map Dimension** page, and then in **Drill URL**, click the drill pen icon.

   For more information on the Map Dimension page, see Mapping Dimensions.
8. In the **Edit Drill URL** page, specify the **#agent** tag and then specify parameters for the drill through URL.

   For example, using the query shown in step 4, enter: **#agent?QUERY=Source Drill Query&SOURCE_ACCOUNT=$ACCOUNT$&SOURCE_PERIOD=$ATTR1$** where:

   - QUERY is the name of the query you defined in step 3
   - SOURCE_ACCOUNT is derived from the ACCOUNT column
   - SOURCE_PERIOD is derived from the ATTR1 column

Copyright © 2017, 2026, Oracle and/or its affiliates.

9. Map the columns in the source column to the dimensions in the target application to which to drill through and click **Save**.
10. From the Data Integration home page, click **Actions**, then from the **Configure** actions drop-down select **Agent**.
11. From the **Agent Cluster** screen, select the name of the agent to use with the drill through.
12. From the **Agent** tab, and then in **Web URL**, select the web server gateway that enables inbound communications from the Oracle Fusion Cloud Enterprise Performance Management.

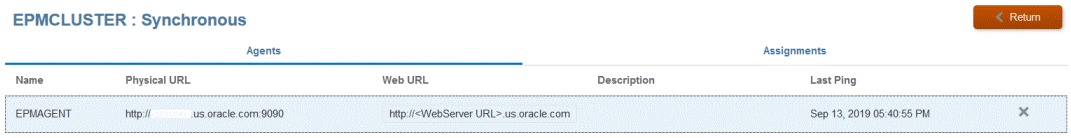
    If you are using synchronous mode, the Web Server URL you defined to perform the data load operations is used to perform the drill. No additional setup is required.

    If you are using agent in asynchronous mode, define the URL using one of the following methods:

    - Configure a https web server similar to how you configured the synchronous mode. See Configuring Synchronous Mode.
    - Assign the Agent URL as the web server URL. As a result of accessing the http URL from a https page, you need to define an Exception in the browser security settings.

      For Chrome, select **Settings**, then **Privacy and security** , then **Insecure content**, and then under **Allow**, add the following site ***.oraclecloud.com**.

> **Note**
>
> You must be in your local network where the agent is running to drill when you are running in asynchronous mode.



Drill through to Oracle Autonomous Database Source Without Using the EPM Integration Agent

When integrating to an Oracle Autonomous Database data source, you can drill down to the source to view the details associated with account balance data loaded to the Oracle Fusion Cloud Enterprise Performance Management. You must execute a SQL query in the source database and display the results in a pop-up window inside Data Integration.

Creating a Query for Drill Through

To create a query for the drill through:

1. On the **Data Integration** home page, from the **Actions** menu, click **Setup** and then click **Query**.
2. On the **Query** page, click ➕
3. From the **Create Query** page, in **Query Name**, specify the name of the query to query the data from the source system.

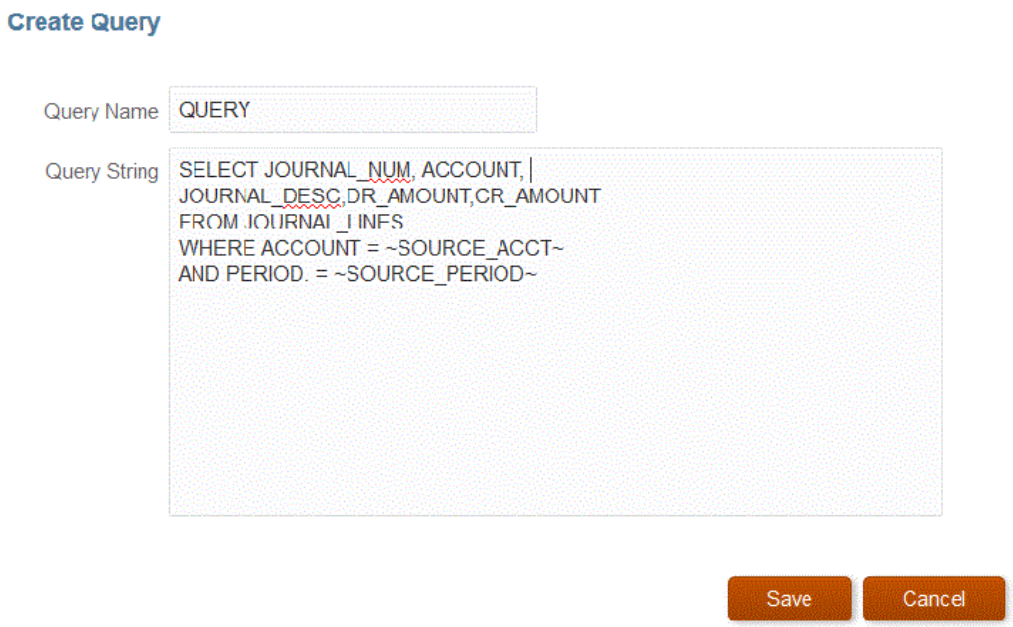
Copyright © 2017, 2026, Oracle and/or its affiliates.

4. In **Query String**, specify the SQL statement used to extract the data based on the header row in the file.

   You can use an alias for a column name in the SQL query to the dimension name.

   When specifying the string for a header row value and the string is mixed case, enclose the string in double quotation marks. When a string with mixed case characters is not enclosed within double quotation marks, it is resolved in uppercase.

   The query can include filter conditions. The bind variable for the filter condition must be enclosed inside ~~ characters.



5. Click **Save**.

6. From the **Data Integration** home page, click ... to the right of the integration for which you want to add a drill through, and then select **Map Dimensions**.

7. On the **Map Dimension** page, in **Drill URL**, click the drill pen icon.

   For more information about the Map Dimension page, see Mapping Dimensions.

8. On the **Edit Drill URL** page, specify the **#oadb** tag and then specify parameters for the drill through URL.

   For example, using the query shown in step 4, enter: `#oadb?QUERY=Source Drill Query&SOURCE_ACCOUNT=$ACCOUNT$&SOURCE_PERIOD=$ATTR1$`

   Where:

   - QUERY is the name of the query you defined in step 3
   - SOURCE_ACCOUNT is derived from the ACCOUNT column
   - SOURCE_PERIOD is derived from the ATTR1 column

9. Map the columns in the source column to the dimensions in the target application to which to drill through and click **Save**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Viewing Oracle Autonomous Database Source Using Drill Through

To view the Oracle Autonomous Database source using drill through:

1. On the Data Integration page, click (Action icon) and then click **Workbench**. The integration details window is displayed.
2. Click any of the links in the **Source Amount** column and then click **Drill Through Source**. The details of the source amount is displayed.

Drilling Through to Legacy On-premises ERP Systems

When drilling from the Oracle Fusion Cloud EPM, you can drill to an on-premises system like the Oracle E-Business Suite suite using a URL. This is the same URL that was used in the legacy Oracle Hyperion Financial Data Quality Management, Enterprise Edition and can also be used when drilling from the cloud to on-premises. See the support link below that provides information on formatting the drill URL for the following on-premises ERPs:

- E-Business Suite
- PeopleSoft
- JD Edwards
- SAP

For more information on the drill through URL details for the above legacy ERP systems, see the "FDMEE" pdf available from Oracle Support at: https://support.oracle.com/support/?kmContentId=1636621&page=sptemplate&sptemplate=km-article .

Copyright © 2017, 2026, Oracle and/or its affiliates.

16

Synchronizing Data

Synchronize and move data between cubes in a single or multiple environments in different environment.

Data synchronization enables you to move data between cubes (plan types) in a single Oracle Fusion Cloud Enterprise Performance Management business process or move data between two Cloud EPM business processes in different environments.

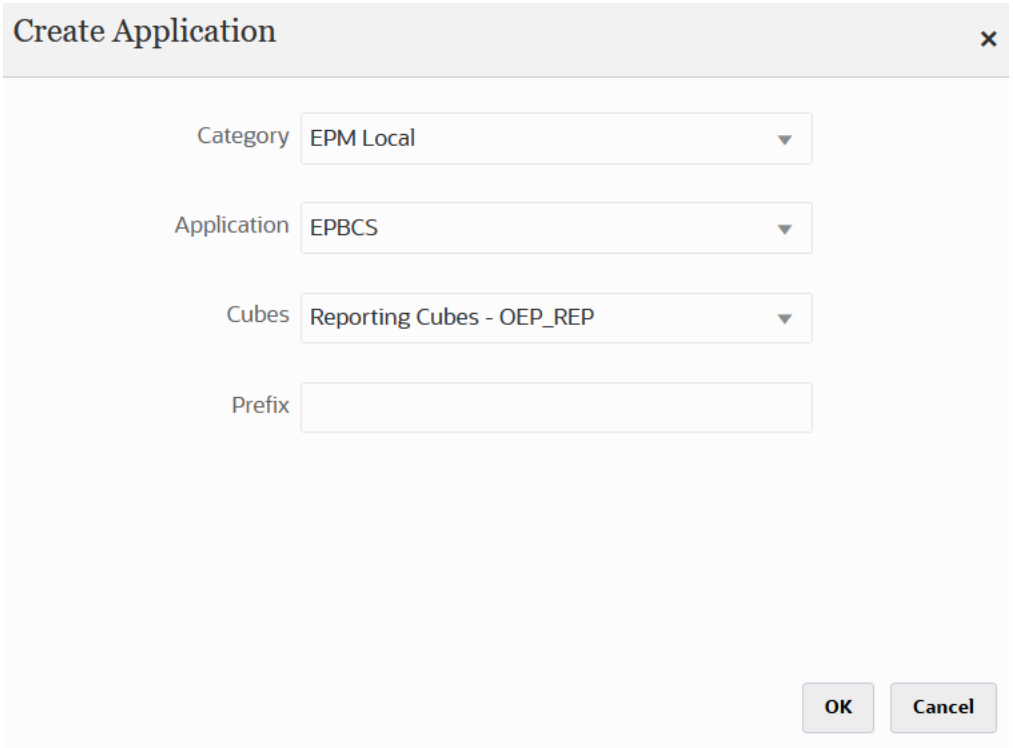
For example, use data synchronization to move data from:

- Planning input cubes to reporting cubes,
- Actuals from Financial Consolidation and Close to Planning reporting cube for variance reporting.

Data Synchronization Process Description

At a high level here are steps to synchronize data in order to move data to a single Oracle Fusion Cloud Enterprise Performance Management business process:

1. To move data to reporting cubes in the same Cloud EPM business process:
   a. From the **Applications** page, add an **EPM Local** application, and select the reporting cube.
   b. Click **OK** to register the application.

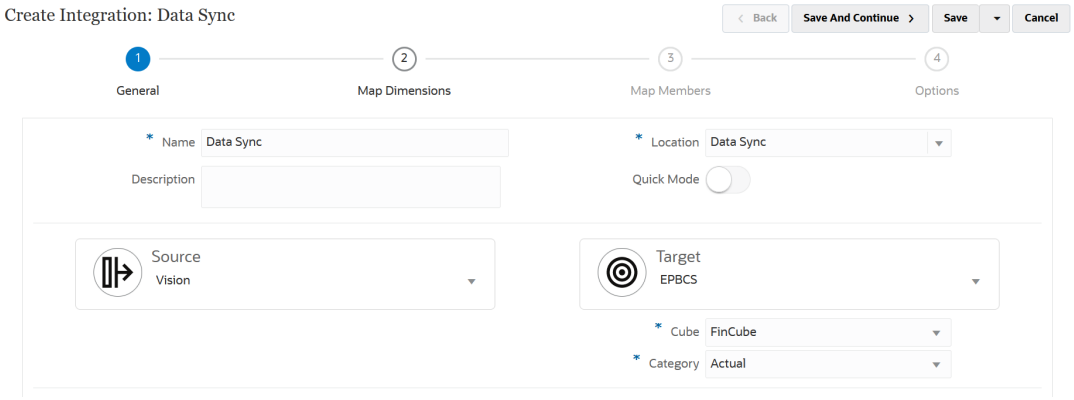


For more information, see Registering Applications.

Copyright © 2017, 2026, Oracle and/or its affiliates.

2. On the **General** page, create the integration between the source and the target applications to be synchronized.

   When synchronizing data to an ASO or BSO application and the target is an input cube, from **Cube**, select the target input cube.



   For more information, see Creating Direct Integrations.

3. On the **Map Dimension** page, map the dimensions between the source and target.

   You map the dimensions that comprise each source and target so that the synchronization can recognize all relevant elements.

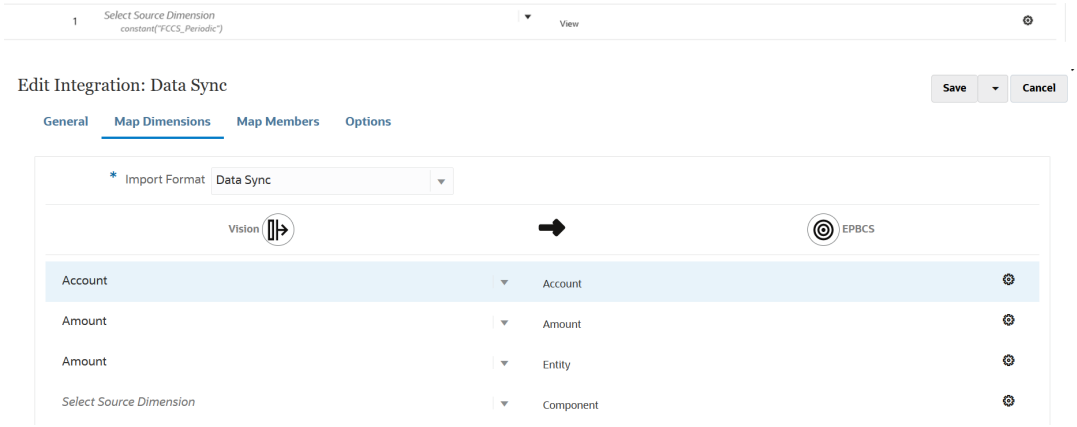
   If the source and target members are the same, define a target expression and select **copySource()** as the expression. For more information, see Copy Source Value.

   Additionally, you can define any other expressions for transforming the data.

   For more information about source expressions, see Using Source Expressions.

   For more information about target expressions, see Using Target Expressions.

   If you want to load periodic data to Financial Consolidation and Close, you should map the FCCS_Periodic member to the View dimension. This is possible using the constant source expression with FCCS_Periodic as the value as shown below:



   For more information about mapping dimensions, see Mapping Dimensions.

4. On the **Map Members** page, map members for each dimension to use target system members based on existing source values from the source.

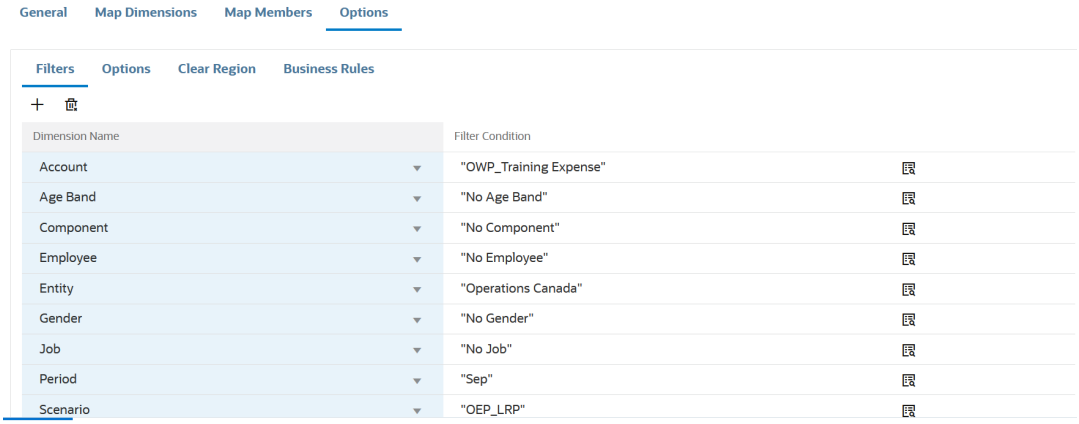
   For more information, see Mapping Members.

5. On the **Options** page, define any integration parameters such as source filters, and general and target options.

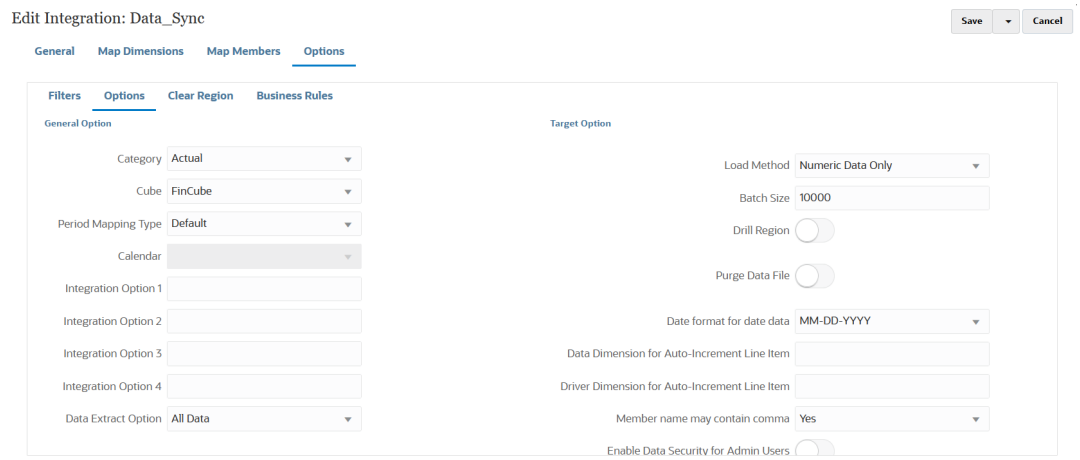
Copyright © 2017, 2026, Oracle and/or its affiliates.

For more information about general options, see Defining Direct Integration Options.

From the **Filters** tab, select a dimension, and then enter filter criteria for it. For example, you can specify the subset of budget data to extract from your Planning source and load it to the target. For more information, see Defining Planning Filters.



From the **Options** tab, then under **General Option**, select the source cube, period mapping type (default or explicit), and the Data Extract option.



For more information about Data Extract Options, see Step 7 Data Extract Option in Defining Direct Integration Options.

Under **Target Option**, select any options to manage how data is loaded to the target such as how you load data including load methods, date formats, batch, and purge options

For more information about target options, see Defining Target Options.

6. From the **Run Integration** page, run the integration to execute the synchronization of the source and target.

   For more information, see Running an Integration.

Quick Mode Data Synchronization

The Quick Mode data synchronization feature enables customers to move data within or between Oracle Fusion Cloud Enterprise Performance Management business processes. This method provides better performance for data movement within your Cloud EPM business processes.

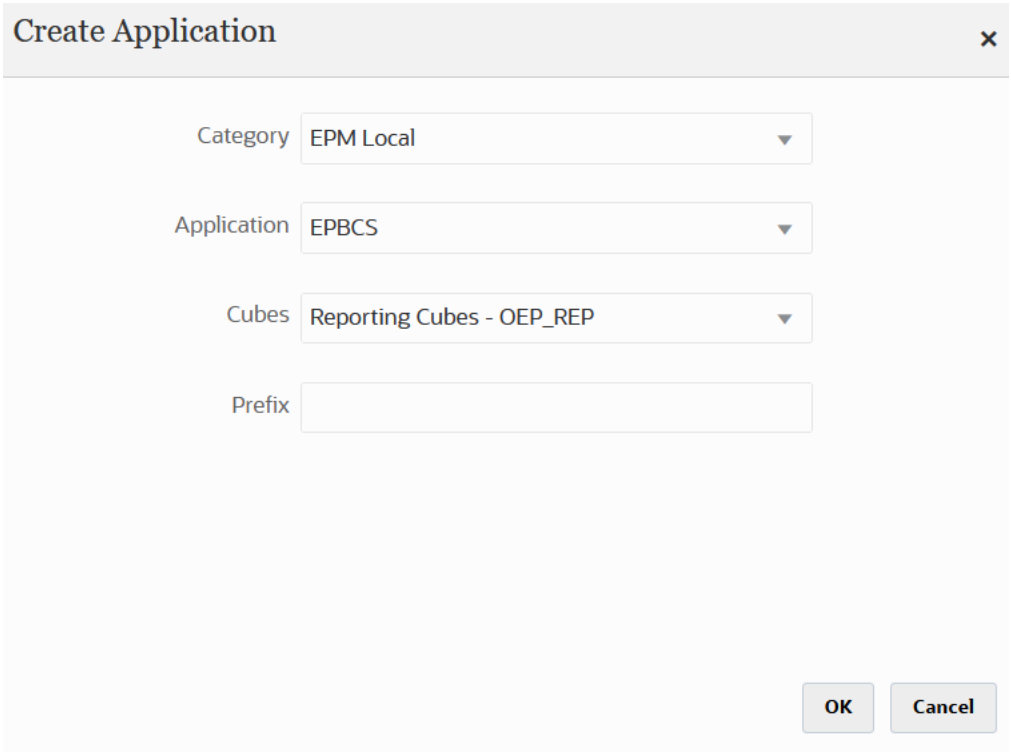
Copyright © 2017, 2026, Oracle and/or its affiliates.

Quick Mode Data Synchronization Process Description

At a high level here are steps to synchronize data in Quick Mode to move data to a single Oracle Fusion Cloud Enterprise Performance Management business process or move data between two Cloud EPM business processes in different environments.

To synchronize data using Quick Mode:

1. From the **Applications** page, add an **EPM Local** application, and select the reporting cube.
2. Click **OK** to register the application.



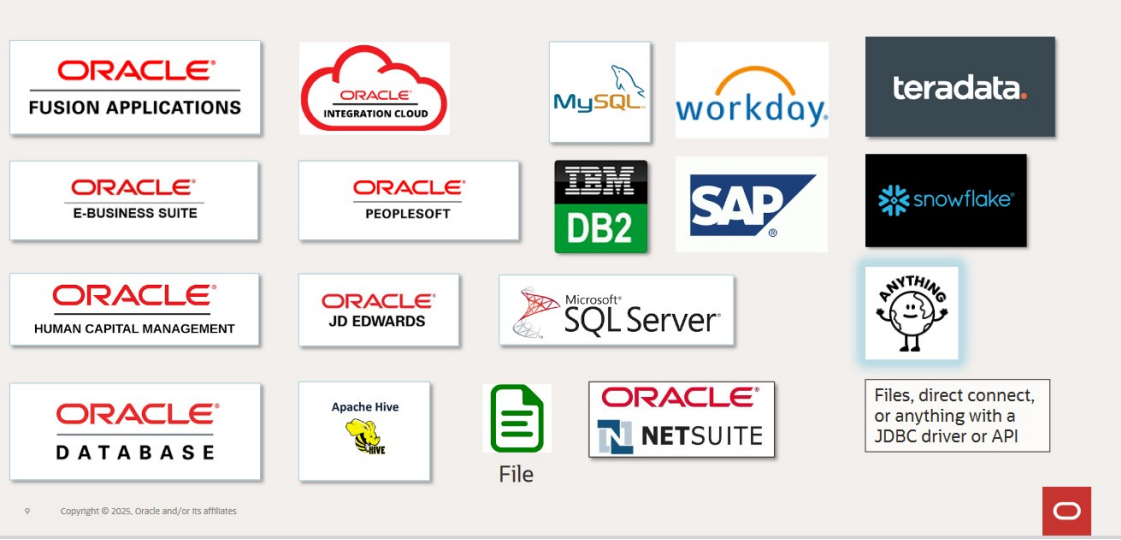
Copyright © 2017, 2026, Oracle and/or its affiliates.

17

Integrating Data

Create an integration definition to determine how to extract or write back data from source and target systems.

Data Integration provides enhanced integration options that allows you to integrate any source and source and target combinations.



Integrating Oracle General Ledger Balances from the Oracle ERP Cloud

Push Oracle General Ledger balances data from the Oracle ERP Cloud to your Oracle Fusion Cloud Enterprise Performance Management application.

You can integrate Oracle General Ledger balances data from the Oracle ERP Cloud with your Cloud EPM application. This integration enables you to simply pick the desired source ledger from the Oracle ERP Cloud, set up a few simple mappings and then push a button to pull the data into the Cloud EPM applications. This integration can be run manually or scheduled for a specific time.

> **Note**
>
> The Average Daily Balances (ADB) ledger is not supported in the current integration.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> Data Integration also supports the Financials Accounting Hub (FAH) and the Financial Accounting Hub Reporting Cloud Service (FAHRCS) as part of its integration with the Oracle General Ledger.

Data Integration sets up the drill definition automatically to an Oracle General Ledger application.

Data Integration facilitates not only data loads but write-backs to the Oracle ERP Cloud.

Videos

Your Goal	Learn How
For administrators, this overview highlights the requirements and tasks you need to complete when integrating Oracle Fusion Cloud ERP with Oracle Cloud EPM using prepackaged extracts.	Overview: Integrating Oracle Cloud ERP with Oracle Cloud EPM

Integration Process Description

At a high level, this is how you integrate Oracle General Ledger data from the Oracle ERP Cloud with your Oracle Fusion Cloud Enterprise Performance Management application:

1. In Data Integration, configure the Oracle ERP Cloud connection information, and then register the Oracle ERP Cloud as a data source.

   For more information, see Configuring a Source Connection.

2. Register the Cloud EPM application that requires the data from the Oracle ERP Cloud (GL Balances).

   For more information, see Registering Cloud EPM Applications.

3. Register the Oracle General Ledger application in the Oracle ERP Cloud from which to load data to one or more Cloud EPM applications and import the Oracle General Ledger application.

   The import application process brings over the Oracle General Ledger data into the Cloud EPM system as Oracle Essbase cubes. Each Essbase application represents a chart of accounts definition from the source Oracle ERP Cloud.

   For more information, see Registering Oracle Cloud ERP Applications.

4. Create the integration between the source application and the target Cloud EPM application.

   For more information, see Creating an Integration with the Oracle ERP Cloud (GL Balances).

5. Map the dimensions between the Cloud EPM application and the dimensions in the segments in the Oracle General Ledger segments.

   For more information, see Mapping Dimensions.

6. Map members to convert the chart of accounts values from the Oracle General Ledger to dimension members in the Cloud EPM during the transfer.

   For more information, see Mapping Members.

Copyright © 2017, 2026, Oracle and/or its affiliates.

7. On the **Options** page, select any filters and application options.

   A default filter is provided that includes all dimensions of the Essbase cube. The cube may have duplicate members so fully qualified member names are required. The Essbase cubes work off the Oracle General Ledger segments, and there is a one-to-many relationships of Chart of Accounts to ledgers in the Oracle General Ledger.

   Data Integration creates filters when an integration is created. You can modify the filters as needed but cannot delete them. (If the filters are deleted, Data Integration recreates the default values). For information about these filters, see Defining Oracle General Ledger Filters.

8. **Optional**: Define adjustment periods from the source Oracle General Ledger when loading balances to a Cloud EPM application.

9. Run the integration job using any filters.

   This process extracts and loads data from the Oracle General Ledger to the Cloud EPM.

10. **Optional**: Write back Cloud EPM data to the Oracle General Ledger in the Oracle ERP Cloud.

    To write back data to the Oracle General Ledger from a Cloud EPM source system, set up and run the integration. In this case, the filters are applied against the Cloud EPM application.

    Optionally, you can write back budget data from the Cloud EPM to a flat file using a custom target application. This output file may be used to load data to any other application.

Configuring a Source Connection

To begin integrating the Oracle General Ledger with the Oracle Fusion Cloud Enterprise Performance Management, you first create a connection to the "Oracle ERP Cloud."

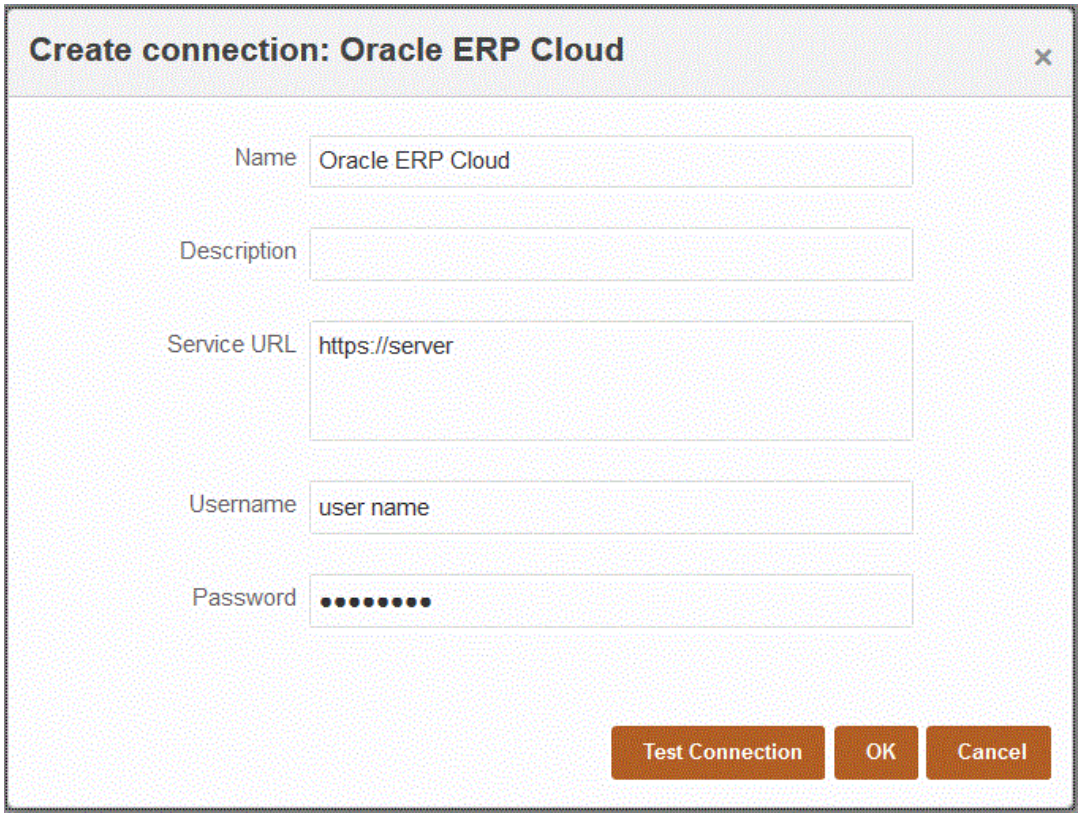
To define a connection to the Oracle ERP Cloud:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.

2. On the **Applications** page, click (Configure Connections icon).

3. On the **Connections** page, select **Oracle ERP Cloud** from the (Add drop-down icon) drop-down.

Copyright © 2017, 2026, Oracle and/or its affiliates.



4. In **Name**, enter the source system name.
5. In **Description**, enter a description of the source system.
6. In **Service URL**, enter the server information for the web services.
7. In **Username**, enter the Oracle ERP Cloud user name.

   Enter the name of the Oracle ERP Cloud user who launches the process requests to send information between Cloud EPM and the Oracle ERP Cloud. This user must have an assigned Oracle General Ledger job role such as "Financial Analyst," "General Accountant," or "General Accounting Manager."
8. In **Password**, enter the Oracle ERP Cloud password.

   You must update this password anytime you change your Oracle ERP Cloud password.
9. Click **Test Connection**.

   When the connection has been tested correctly, the information message "Connection to [*source system name*] successful" is displayed.
10. Click **OK**.

Registering an Oracle General Ledger Application

When integrating Oracle General Ledger data from the Oracle ERP Cloud with your Oracle Fusion Cloud Enterprise Performance Management application, you use Applications to specify the Oracle ERP Cloud as a data source application type and then import the applications so that they can be used as the source from which to integrate Oracle General Ledger balances with Cloud EPM target applications. When you import the applications, the system brings over the source Oracle General Ledger data into the Cloud EPM system as Essbase cubes. Each Essbase application represents a chart of accounts definition from the source Oracle General Ledger.

Copyright © 2017, 2026, Oracle and/or its affiliates.

To define an Oracle ERP Cloud as a data source:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, click (Add icon).
3. From **Category**, select **Data Source**.
4. From **Type**, select **Oracle ERP Cloud**.
5. From **Connection**, specify the name of the connection.

   For example, you might specify: **Fusion**.
6. In **Application Filter**, specify any filters for the load.

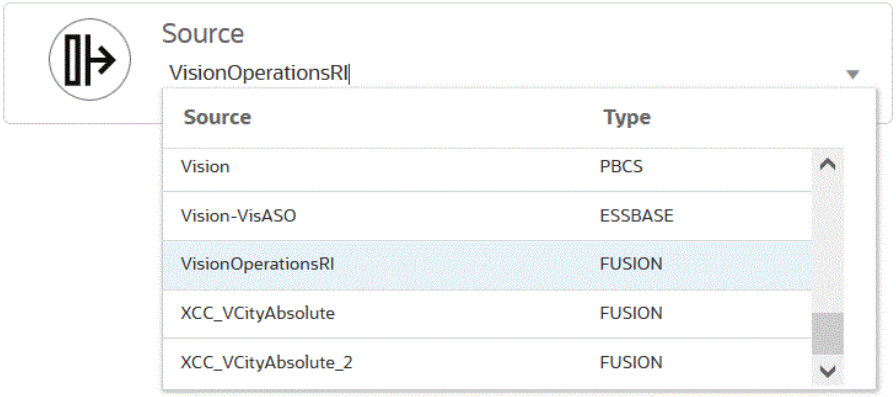
   An application filter is used to filter the names of application to import from the Oracle ERP Cloud. For example, you can filter on a specific application name such as Vision USA or wild card Vision%.
7. Click **Import Applications** to register and initialize the Oracle General Ledger application.

Creating an Integration with the Oracle ERP Cloud (GL Balances)

To load General Leger balances from the Oracle ERP Cloud, create an integration between the source Oracle General Ledger application and the Oracle Fusion Cloud Enterprise Performance Management application.

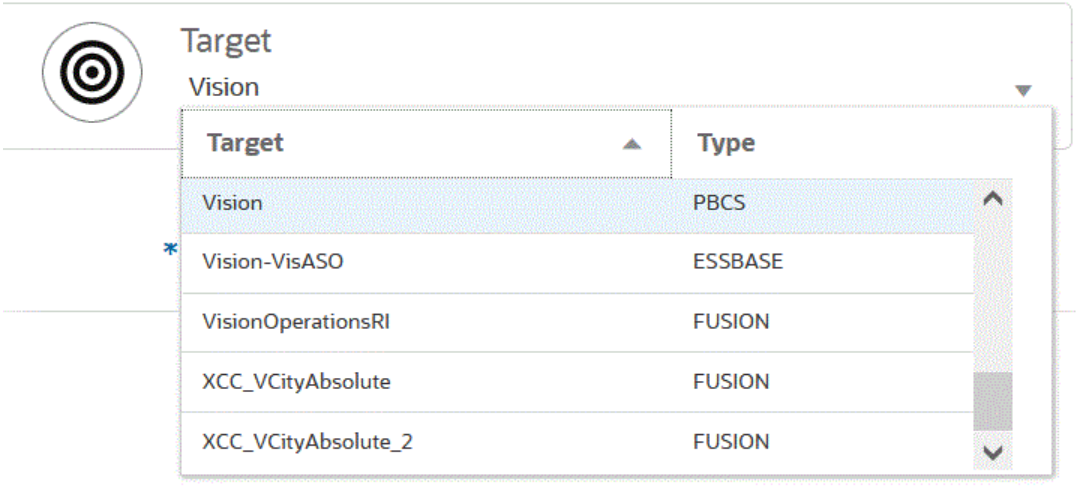
To create an integration between Oracle General Ledger application and the Cloud EPM target application:

1. From the **Data Integration** home page, click (Add icon) to create a new integration.
2. On the **Create Integration** page, then **Name** and **Description**, enter a name and description for the new integration.
3. In **Location**, enter a new location name, or pick an existing location to specify where to load data.
4. From the **Source** ( ) drop-down, select the Oracle ERP Cloud source.



Copyright © 2017, 2026, Oracle and/or its affiliates.

5. From the **Target** ( ) drop-down, select the Cloud EPM target application.



6. From **Cube**, select the plan type of the target application.
7. From **Category**, select the category mappings for categorizing and mapping source system data to a target Scenario dimension member.

   The categories listed are those that you created in the setup, such as "Actual." For more information, see Using Category Mappings.
8. **Optional**: Select any applicable location attributes for the integration. For more information, see Selecting Location Attributes.
9. Click **Save**.

Defining Location Attributes

Location attributes enable you to refine which type of data is loaded by location. For example, you can select a functional currency such as the US dollar.

To edit location attributes:

1. From the **Data Integration** home page, click ... to the right of an integration to which to define location attributes and then select **General** from the drop-down.
2. On the **Edit Integration** page, click **Location Attributes**.
3. From **Functional Currency**, specify the currency of the location.
4. From **Parent Location**, enter the parent assigned to the location.

   Parent mappings are used to share mappings with other locations. Enter mappings at the parent location, and the related locations can use the same mappings. Multiple locations can share a parent. This feature is useful when multiple locations use one chart of accounts. Changes to a child or parent mapping table apply to all child and parent locations.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> If a location has a parent, the mappings are carried over to the child. However; changes to mapping can only be performed on the parent location.

5. From **Logic Account Group**, specify the logic account group to assign to the location.

   A logic group contains one or more logic accounts that are generated after a source file is loaded. Logic accounts are calculated accounts that are derived from the source data.

   The list of values for a logic group is automatically filtered based on the Target Application under which it was created.

6. From **Check Entity Group**, specify the check entity group to assign to the location.

   When a check entities group is assigned to the location, the check report runs for all entities that are defined in the group. If no check entities group is assigned to the location, the check report runs for each entity that was loaded to the target system. Data Management check reports retrieve values directly from the target system, Data Integration source data, or Data Integration converted data.

   The list of values for a check entity group is automatically filtered based on the target application under which it was created.

7. From **Check Rule Group**, specify the check rule group to assign to the location.

   System administrators use check rules to enforce data integrity. A set of check rules is created within a check rule group, and the check rule group is assigned to a location. Then, after data is loaded to the target system, a check report is generated.

   The list of values for a check rule group is automatically filtered based on the target application under which it was created.

8. Click **Save**.

Using Category Mappings

When integrating Oracle ERP Cloud data, you can select category mappings that categorize and map source system data to a target Oracle Fusion Cloud Enterprise Performance Management Scenario dimension member. For example, you may have a Scenario dimension member called Actuals for storing actual balances from an Oracle ERP Cloud application. In a Planning application, the same source system data is stored using the Scenario dimension member "Current". In Data Integration, you can create one category mapping to give both one name to represent their respective scenarios. To do this, use the Category Mappings option. For more information on defining categories, see Managing Category Mappings.

Mapping Dimensions

Mapping dimensions enables you to define how source dimensionality translates to the target dimensionality between the Oracle Fusion Cloud Enterprise Performance Management application and the dimensions in the Oracle General Ledger application.

> **Note**
>
> Oracle General Ledger creates one Essbase cube per Chart of Account/Calendar combination. In this case, you can use the same import format to import data from Ledgers sharing this Chart of Accounts. Ledgers can be specified as a filter in the data load rule.

Copyright © 2017, 2026, Oracle and/or its affiliates.

To map dimensions between the source Oracle General Ledger application and the Cloud EPM target application:

1. From the **Data Integration** home page, click to the right of the integration, and then select **Map Dimensions**.
2. On the **Map Dimensions** page, then **Import Format**, select the name of the import format to use for the integration.

   You can also add a user defined import format name.
3. In the mappings grid, map the source columns in the source to the dimensions in the target application.

   The dimensions from the target application are populated automatically.

   If the import format has already been defined for the integration, then the source and target columns are mapped automatically.

   If you are adding a new import format or editing an existing import format, complete the following:

   - In **Column**, specify the field number from the file to import.
   - In **Select Source Dimension**, specify the name of the source dimension to assign to the target application.

     Multiple source columns of the same dimension can be mapped to target dimensions. For example, you can map four "Account" source columns.
   - Add a source or target expression by assigning an expression that operates on values directly from the source or target.

     See Using Source Expressions and Using Target Expressions.

     Optional: For a comma delimited file, select an additional row to map in the import format by clicking to the right of a row and selecting the row to add from the drop-down.

     Available rows:

     – Source Period
       * Year
       * Period
       * Period Number
     – Currency
     – Attribute
     – Description
     – Dimension Row
       * Account
       * Version
       * Entity
       * View

     You can also skip a row.
4. Click **Save**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

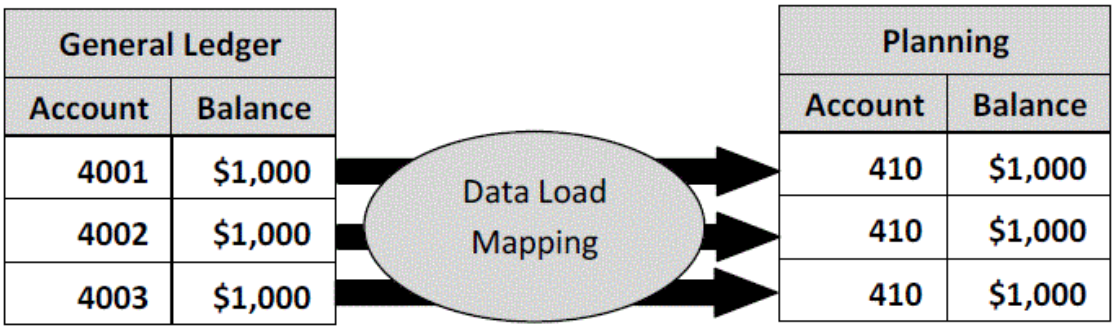
Mapping Members

When you map members, the system converts the Chart of Accounts values from the Oracle General Ledger to the Oracle Fusion Cloud Enterprise Performance Management dimension members during the transfer. This allows Data Integration to categorize Oracle General Ledger balances.

In the following example, based on the Chart of Accounts segment, the Oracle General Ledger breaks down administrative expenses as a range of accounts from 4001-4003.

In Planning, budgeting for administrative expenses is done for dimension value 410, Administrative Expenses.

The example shows how Oracle General Ledger accounts in the range 4001-4003 actual amounts are mapped to 410 administrative expenses in Planning.



Differences in Planning dimension values and Oracle General Ledger Chart of Accounts values may occur. Additionally, segments of the Chart of Accounts may not be used when budgeting.

For example, you can map zero values (such as "0000" for sub-account) for the Chart of Accounts segments that are not used for budgeting.

> **Note**
>
> When there is no update to the Oracle General Ledger value prior to the load, it is still necessary to create the member mapping for the dimensions to instruct Data Integration to create the target values.

To define map members:

1. From the **Data Integration** home page, click ... to the right of the integration, and then select **Map Members**.
2. On the **Map Members** page, from the **Dimensions** drop-down, select the dimension that you want to map.

   At a minimum, map values for the "Account" and "Entity" dimensions since those are transferred from Oracle General Ledger.

   If you are transferring additional chart segments you must provide a mapping for each destination dimension.
3. Select the member mapping type from the **Mapping Type** drop down, and then specify the source value.

   Types of member mappings:

Copyright © 2017, 2026, Oracle and/or its affiliates.

Type of Mapping	Description	See Also
== Explicit	Match and replace the source value exactly with the target value. Explicit mappings are one to one mappings, for example, source value "ABC" is replaced with target value "123."	Using Explicit Mappings
<> Between	Replace a continuous range of source values with a single target value. For example, a range from "001" to "010" is replaced as one value. In another example, you may need to map accounts 300000 through 3001999 to Retained Earnings, but accounts 310000 might be Capital contributions or dividends.	Using Between Mappings
IN In	List non-sequential (non-continuous) source values to be mapped to one target value. In this case, multiple values are mapped to one value within one mapping, eliminating the need to create multiple rules (as is required for an Explicit map). For example, you could have source accounts 1503, 1510, and 1515 map to the target account 15000010.	Using In Mappings
Is Like	Use special characters to match a string in the source value and map it to a target value. Like mappings use wildcard characters: asterisks (*) and question marks (?). Asterisks are placeholders for any number of characters. For example, 1190* maps the accounts 1190, 1190100, and 1190-200 to the Cash target account. Question marks are placeholders for one character. For example, the source account of 119? maps only to source accounts that contain four characters and that begin with 119.	Using Like Mappings
Is Multi Dimensional	Multi-dimension mapping enables you to assign a target value for a specific combination of source column values. This functionality provides you with the ability to load data into dimensions unavailable in the target application. For example, the mapping for the Account dimension can be based on source values of Entity, Product, and Project. In addition, Lookup dimensions added in the target application registration can be selected. These dimensions contain source dimensions that do not exist in the target application. They provide even more flexibility in creating multi-dimension filters. It is a way to facilitate conditional data loading.	Using Multi-Dimensional Mappings

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> When processing the source values for transformations, multiple mappings may apply to a specific source value. The order of precedence is Explicit, Between, In, Multi-Dimensional, and Like. Within Between and Like types mappings can overlap.

4. Click + to add a new member mapping.

   You can also just edit existing mappings as needed without adding a new mapping by selecting a mapping and clicking the Edit icon .

5. From **Add Member Mapping** page, then **Source**, select the source dimension member to map to the target dimension member.

   Enter the values from the Oracle General Ledger. Enter the values directly.

   To map all members to then Cloud EPM business process "as is" without any modification, in **Source**, enter: ***, and in **Target**, enter: ***.

6. In **Target**, enter the value for the accounting scenario to use to load the budget information.

   Enter the values that should be used in the Cloud EPM business process to store the Oracle General Ledger actual balances that are transferred.

7. In **Processing Order**, specify the order of the mapping.

   The processing order determines the level of precedence within a mapping type. Mappings are processed in alphabetical order by the name within a mapping type. Numbers may also be used to help with ordering. For example, if you use Number for processing, note that the processing order is in an alphanumeric sort order. If have 10, 20, 30, 100 as the order, the processing order will be 10, 100, 20, 30. When using numbers for the processing order, use the same number of digits for all maps.

8. In **Description**, enter a description of the member mapping.
   For example, enter a description such as "Map to the General Ledger".

9. In **Change Sign**, select to reverse the sign of the target account specified.

   The Change Sign option is often used with Oracle General Ledger source data when the trial balance has negative signs for Revenue and Liability/Equity source accounts. In Cloud EPM applications, positive numbers are often loaded as credits and all negative numbers are loaded as debits. Consequently, you can reverse the sign.

   > **Note**
   >
   > The Change Sign option is not supported for Quick Mode loads.

10. **Optional:** From the **Apply To** drop-down, select the name of the integration to which to apply the member mapping.

11. On the **Add Member Mapping** page, click **OK**.

12. On the **Member Mapping** page, click **Save**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Defining Oracle General Ledger Filters

For data integrations used to import data from the Oracle General Ledger, use filters to limit the results.

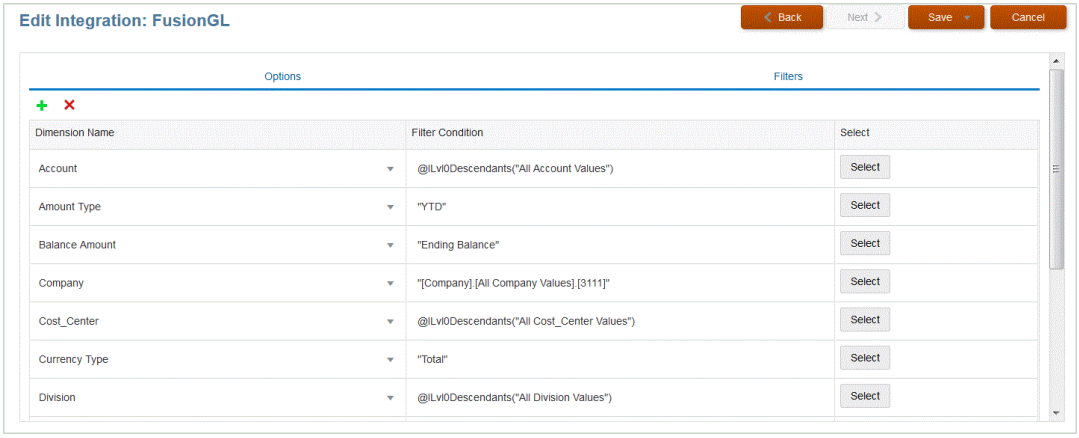
When an integration is created, filters are defined automatically. You can modify the filters as needed but cannot delete them. (If filters are deleted, the default value is recreated.)

Oracle General Ledger Dimension	Filter
Scenario	Actual
Balance Amount	Ending Balance
Amount Type	YTD
Currency Type	Total
All Other Dimensions	'@ILvl0Descendants("All '|| TARGET_DIMENSION_NAME||' Values")'

To add an Oracle General Ledger filter:

1. From the **Data Integration** home page, click ... to the right of the Oracle General Ledger integration, and then select **Options**.
2. Click the **Filter** tab.
3. Click + .

   Optionally, you can simply select another dimension already assigned to a filter and select either another dimension from the **Dimension Name** drop-down or just change the filter condition.



4. From the **Dimension Name** drop-down, specify the name of the dimension to which to add as a filter.
5. In **Filter Condition**, specify the filter.
6. Click **Save**.

Processing Oracle General Ledger Adjustment Periods

You can include adjustment periods from an Oracle General Ledger source system in the Oracle ERP Cloud when loading balances to an Oracle Fusion Cloud Enterprise Performance Management application.

Copyright © 2017, 2026, Oracle and/or its affiliates.

You can include adjustment periods from an Oracle General Ledger source system from the Oracle ERP Cloud when loading balances to a Cloud EPM application.

Adjustment periods are additional periods that are related to regular periods from the source. An "adjustment period" refers to any accounting period set up to adjust balances prior to the closing period of the year. These periods are adjusted to "per12" and consequently are referred to as "per13." Typically, dates within the adjustment period overlap regular accounting periods. A customer might use a "Year Open Period" that refers to the first period in the accounting calendar to adjust last year's balance carried forward amount. Additionally, the customer can set up the last period of the accounting calendar as the "Year Close Period" to adjust transactions made in the current accounting calendar.

In Data Integration, adjustments are processed in Period Mappings where you indicate how the adjustment period maps to the period in the target application. The method for how adjustments get processed is specified in the integration. This feature enables you to map Oracle General Ledger source periods in Data Integration simply by pointing to the calendar and periods from the Oracle General Ledger application to the period in the Oracle ERP Cloud application.

When setting up the integration, you can either load to regular and adjustment periods when an adjustment period mapping exists or loads an adjustment period only when an adjustment period mapping exists.

For example, when you map period 13 to December/Period 12, and select the Include Adjustment Period option, then the following occurs:

- For YTD balances, period 13 becomes the ending balance.
- For PTD balances, period 13 and December/Period12, are added.

To include adjustment periods from an Oracle General Ledger source system:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Period Mapping**.
2. Select the **Source Mapping** tab.
3. From the **Source Type** drop-down, select **Oracle ERP Cloud**.
4. From the **Connection** drop-down, select the name of the connection used to connect to the Oracle ERP Cloud.
5. From the **Source Application** drop-down, select the Oracle ERP Cloud source application from which the adjustment is based.
6. From **Target Application**, select the Cloud EPM application to which the adjustment applies.
7. From **Mapping Type**, select **Adjustment**.
8. Click **Add**.
9. In **Source Period Key**, specify the last day of the month to be mapped from the Oracle General Ledger source system.

   Use the date format based on the locale settings for your locale. For example, in the United States, enter the date using the **MM/DD/YY** format.

   You can also click and browse to and select the source period key.

   When you select the **Source Period Key**, Data Integration populates the **Source Period** and **Source Period Year** fields automatically.

Copyright © 2017, 2026, Oracle and/or its affiliates.

10. In **Adjustment period**, specify the name of the adjustment period from the Oracle General Ledger source.

    For example, if the adjustment period from the Oracle General Ledger is Adj-Dec-16, then enter: **Adj-Dec-16** in this field.

11. In **Target Period Key**, specify the last day of the month to be mapped from the target system.

    Use the date format based on the locale settings for your locale. For example, in the United States, enter the date using the **MM/DD/YY** format.

    You can also click and browse to and select the target period key.

    When you select the **Target Period Key**, Data Integration populates the **Target Period Name**, **Target Period Month**, and **Target Period Year** fields automatically.

12. Click **Save**.

13. From the **Data Integration** home page, click to the right of the direct integration, and then select **Options**.

14. In **Category**, specify the default category value.

    The categories listed are those that you created in the Data Integration setup.

15. From **Calendar**, select the name of the source period mapping calendar.

16. In **Period Mapping Type**, select the period mapping type for each data rule.

    Valid options:

    - Default—The Data Rule uses the Period Key and Prior Period Key defined in Data Integration to determine the source General Ledger periods mapped to each Data Integration period included in a Data Rule execution.
    - Explicit—The Data Rule uses the Explicit period mappings defined in Data Integration to determine the source General Ledger periods mapped to each Data Integration period included in a data load rule execution. Explicit period mappings enable support of additional Oracle General Ledger data sources where periods are not defined by start and end dates.

17. From **Include Adjustment Period**, select one of the following options for processing adjustment periods:

    - No—Adjustment periods are not processed. The system processes only regular period mappings (as setup for "default" and "explicit" mappings). **No** is the default option for processing adjustments.
    - Yes—If **Yes** is selected, then the regular period and adjustment period are included. If the adjustment period does not exist, then only the regular period is processed.
    - Yes (Adjustment Only)—If **Yes (Adjustment Only)** is selected, the system processes the adjustment period only. However, if the adjustment period does not exist, the system pulls the regular period instead.

18. Click **Save**.

Running an Integration

You run an integration to extract the data from the source system, and then view and verify the results. If the data has been transformed correctly, you can push it to the target system.

The Run Integration page consists of two tabs: Options and Filters.

Copyright © 2017, 2026, Oracle and/or its affiliates.

The Options tab enables you to select general and target options for the selected integration at the transformation stage. Note that the period selections on this page depend on the target system.

The Filters tab enables you to quickly add or modify filters at runtime and execute them rather than modifying the integration definition every time. Filters are used to query the data from the data sources. The changes to the filters are not saved and used for that execution only.

To run the integration:

1. From the **Data Integration** home page, select the integration, and then click .
2. Select the **Options** tab.
3. Select **Import Source** to import the data from the source system, perform the necessary transformation, such as import, map, and validate the data.

   Select this option only when:

   - you are running an integration for the first-time.
   - the data in the source system has changed. For example, if you reviewed the data in the workbench after the export, and it was necessary to modify data in the source system.

   In many cases, source system data may not change after you import the data from the source the first time. In this case, it is not necessary to keep importing the data if it has not changed.
4. Select **Recalculate** to skip importing the data, but re-process the data with updated mappings.
5. From **Import Mode**, select the method for importing data.

   Available import modes:

   - **Append**—Keep existing rows for the POV but append new rows to the POV. For example, a first-time load has 100 rows and second load has 50 rows. In this case, 50 rows are appended. After this load, the row total for the POV is 150.
   - **Replace**—Clears all data for the POV in the target, and then loads from the source or file. For example, a first-time load has 100 rows, and a second load has 70 rows. In this case, 100 rows are removed, and 70 rows are loaded to TDATASSEG. After this load, the row total is 70.

     For a Planning application, Replace clears data for Year, Period, Scenario, Version, and Entity that you are loading, and then loads the data from source or file. Note that when you have a year of data in the Planning application, but are only loading a single month, this option clears the entire year before performing the load.

> **Note**
>
> When running an integration in Replace mode to an ASO cube, if the scenario member is a shared member, then only a Numeric data load is performed. Be sure to specify the member name with fully qualified name including complete hierarchy. The All Data Type load methods does not work when the scenario member is a shared member.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> Replace Mode is not supported for the load method "All data types with auto-increment of line item."

- **Merge**—Merge adds existing cells and create new cells if they do not exist. It does not add to the data.

  For example, a customer might have 100 rows of existing balances for one number account IDs, each which has an amount of $100.00. If the customer runs the integration in merge mode and the source has one row for one account ID with an amount of $80, then after running integration, there are one hundred rows of balances, 99 each of which have a balance of $100.00, and 1 which has a balance of $80.00.

- **No Import**—Skip the import of data entirely.
- **Map and Validate**—Skip importing the data but reprocess the data with updated mappings.

6. From **Export Mode**, select the method for exporting data to the target application.

   Available options:

   - **Merge**—Overwrite existing date with the new data from the load file. (By default, all data load is processed in the Merge mode.) If data does not exist, create new data.
   - **Replace**—Clears all data for the POV in the target, and then loads from the source or file. For example, a first-time load has 100 rows, and a second load has 70 rows. In this case, 100 rows are removed, and 70 rows are loaded to the staging table. After this load, the row total is 70.

     For a Planning application, Replace clears data for Year, Period, Scenario, Version, and Entity that you are loading, and then loads the data from source or file. Note that when you have a year of data in the Planning application, but are only loading a single month, this option clears the specific month before performing the load.

   - **Accumulate**—Accumulate the data in the application with the data in the load file. For each unique point of view in the data file, the value from the load file is added to the value in the application.
   - **Subtract**—Subtract the value in the source or file from the value in the target application. For example, when you have 300 in the target, and 100 in the source, then the result is 200.
   - **Dry Run**—(Financial Consolidation and Close and Tax Reporting only) Scan a data load file for invalid records without loading data it to the target application. The system validates the data load file and lists any invalid records into a log, which lists 100 errors or less. For each error, the log indicates each record in error with its corresponding error message. Log details are available in Process Details.
   - **No Export**—Skip the export of data entirely.
   - **Check**—After exporting data to the target system, display the Check report for the current POV. If check report data does not exist for the current POV, a blank page is displayed.

7. From **Start Period**, select the first period for which data is to be loaded.

   You can filter periods by simply typing the character(s) to filter by. For example, type **J** to filter by months beginning J such as June or July. You can also click the drop down and specify additional filter criteria in the edit box shown below **More results available, please filter further.**

Copyright © 2017, 2026, Oracle and/or its affiliates.

This period name must be defined in period mapping.

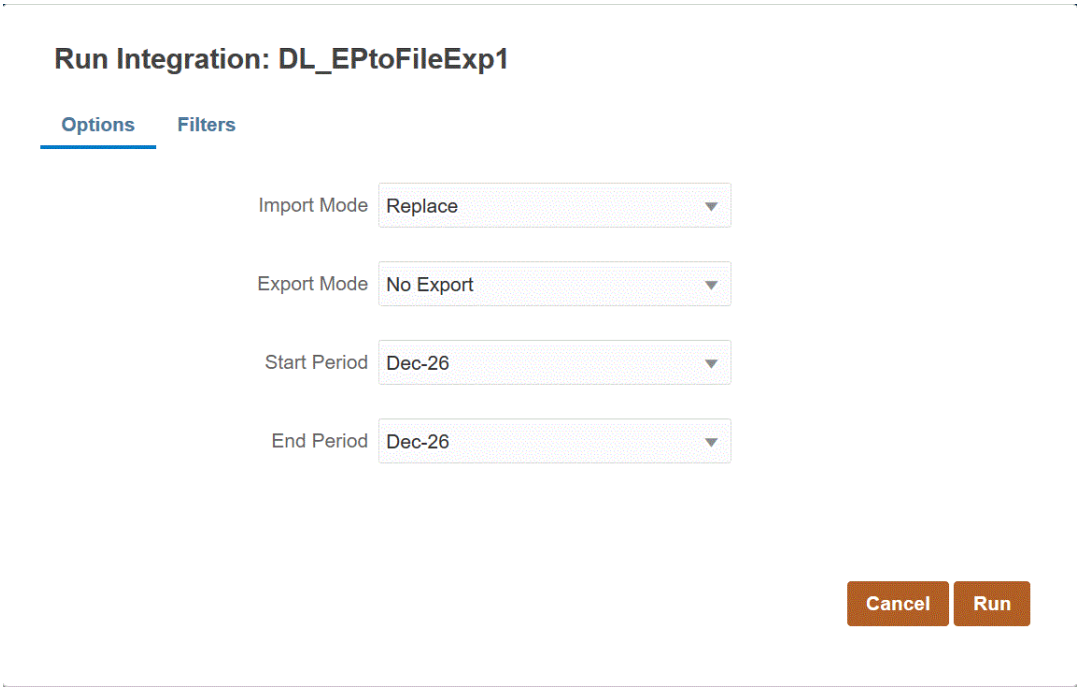
8. From **End Period**, select the last period for which data is to be loaded.

   This period name must be defined in period mapping.

> **Note**
>
> When a POV period has been selected on the Home page, then the selected period defaults here. If no POV period is selected, then the system defaults to the last used period based on browser cache.
> You can select another period when an unlock icon
> (
>
> appears next to it.
>
> When the Global POV mode is enabled in System Settings in , then the period defaults to the Global POV Period and a customer is not allowed to select another period. In this case, a lock icon
> (
>
> appears next to the Period drop-down.

9. Select **Export to target** to export the data to the target application.
10. Select **Execute Check** to generate the data, and then run the Check Report.
11. **(Optional)**: Click **Filters**.
12. Add or modify the filter(s) to execute at runtime.

    Filters are used to query the data from the data sources. Filter specified on the Run Integration page are not saved and used for that execution only.



Copyright © 2017, 2026, Oracle and/or its affiliates.

13. Click **Run**.

Writing Back to the Oracle ERP Cloud

Write-back enables you to write back your budget and actuals to the Oracle General Ledger.

For example, you may need to report budget-to-actual from the Oracle General Ledger.

Writing Back Actuals to the Oracle General Ledger

When actual information is complete in your Oracle Fusion Cloud Enterprise Performance Management application, you can define the Cloud EPM application as a source and then write back data to the Oracle General Ledger from the Oracle ERP Cloud target application.

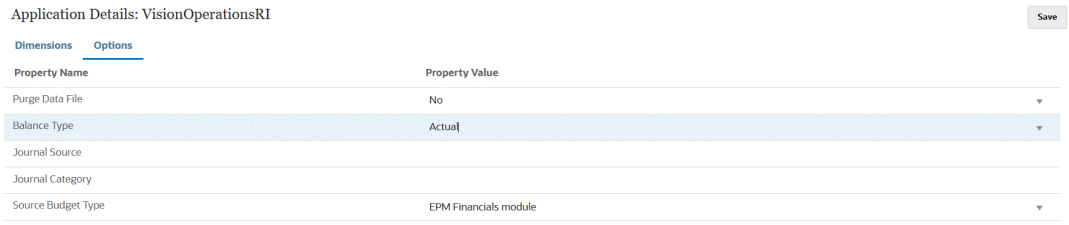
After specifying any necessary filters, you can then extract actual values from Cloud EPM and write them to Oracle General Ledger. In the Run Integration step, the data is written to a flat file, which in turn is copied to a file repository. When data is written back, journal entries are created in the Oracle General Ledger.

On the Oracle ERP Cloud side when configuring the ERP system, make sure the Oracle Fusion ERP Essbase cube has been created using the "Create General Ledger Balances Cube." In addition, scenarios must already be set up in the Oracle Fusion ERP Essbase cube using the "Create Scenario Dimension Members" job.

An Oracle ERP Cloud /Cloud EPM integration requires that you have the privileges or user role and data access to work with all ERP ledgers to be integrated.

To write back to the Oracle ERP Cloud:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. From the **Application** page, click ... to the right of the Cloud EPM source application, and then select **Application Detail**.
3. Select the **Options** tab.
4. From the **Balance Type** drop-down, select **Actual**.
5. In **Journal Source**, enter a description of the journal source that matches the journal source defined in Oracle ERP Cloud.
6. In **Journal Category**, enter a description of the journal category that the matches the journal category in Oracle ERP Cloud.



7. Click **Save**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

8. From the **Data Integration** home page, click , then on the **Create Integration** page, create the integration between the source data to the target, and then click **Save and Continue**.

   a. In **Name** and **Description**, enter a name and description for the new integration.

   b. In **Location**, enter a new location name, or pick an existing location to specify where to load data.

   c. From the **Source** ( ) drop-down, select the name of the Cloud EPM application from the drop-down.

   d. From the **Target** ( ), select the Oracle ERP Cloud application.

   e. From **Cube**, select the plan type of the target system.

   f. From **Category**, select the category mappings for categorizing and mapping source system data to a target Scenario dimension member.

      The categories listed are those that you created in the setup, such as "Actual." For more information, see Managing Category Mappings.

   g. **Optional**: Select any applicable location attributes for the integration. For more information, see Selecting Location Attributes.

   h. Click **Save and Continue**.

9. On the **Map Dimensions** page, map the dimensions in the Oracle ERP Cloud source application to the dimensions in the Cloud EPM application.

   If you are adding new dimensions or editing an existing dimension, complete the following:

   - In **Column**, specify the field number from the file to import.
   - In **Select Source Dimension**, specify the name of the source dimension to assign to the target application.

     Multiple source columns of the same dimension can be mapped to target dimensions. For example, you can map four "Account" source columns.
   - Add a source or target expression: assign an expression that operates on values directly from the source or target.

     See Using Source Expressions and Using Target Expressions.

   Be sure to map a source for the target dimension "Ledger."

   You can map a dimension like "Entity" to the ledger and define any necessary data load mapping to convert to the Oracle General Ledger name. If you are writing back to a single ledger, enter the name of the ledger in the expression column.

   Optional: If you want to populate any additional reference data and/or attribute data for each journal, use the Attribute columns to map the columns.

   Attribute columns, Attribute1 to Attrbute10, are reserved for REFERENCE1 TO REFERENCE10. In this case, you need to also add the REFERENCE column as a dimension and map it to the ATTR column in the target application. For example, to populate REFERENCE3, then insert dimension details and give it an appropriate name, assign the type of Attribute and then assign data column ATTR3. (ATTR11 to ATTR30 are reserved for ATTRIBUTE1 TO ATTRIBUTE20. Attribute1 is stored in ATTR11, Attribute2 is stored in ATTR12 and so on.)

   For more information, see Mapping Dimensions.

Copyright © 2017, 2026, Oracle and/or its affiliates.

10. From **Map Members** page, map any members from the source to target.

    To map all members to the Oracle ERP Cloud "as is" without any modification, select **All** for the **Mapping Type**, click **Add**, and on the **Add Map Member** page, in **Source**, enter: *****, and in **Target**, enter: *****.

    For more information, see Mapping Members.

11. Click **Save and Continue**.
12. From the **Data Integration** home page, and then from the **Actions** menu, select **Period Mapping**.

    The period mapping is used to convert periods to Oracle General Ledger accounting calendar periods for the transfer.

13. Select the **Application Mapping** tab.
14. From **Application Mapping** page, then **Target Application**, select the Oracle General Ledger application from the Oracle ERP Cloud to which to write back data.
15. Click to add a separate row for each period that is to receive actual amounts and complete the following.

    > **Note**
    >
    > When specifying the period, the starting and ending periods should be within a single fiscal year. Providing date ranges that cross fiscal year results in duplicate data.

16. Define a **Period Key**, **Target Period Month** and **Target Period Year**.
    - **Period Key**—Specify the last day of the month to be mapped from the target system.

      Use the date format based on the locale settings for your locale. For example, in the United States, enter: the date using the MM/DD/YY format.
    - **Target Period Month**—The values in this field need to match the accounting calendar for the ledger in the Oracle General Ledger, which receives the transferred amounts.
    - **Target Period Year**—Use values that corresponds to the accounting period (as defined in the Target Period Month column).

    Once you select a value, information about the period key, prior period key, period name, and the target period month are populated automatically.

17. From the **Data Integration** home page, click to the right of the file-based integration, and then select **Options**.
18. Complete the following:
    a. In **File Name**, select the data file name from which you are loading data. It may be the same one from which you created the data source application, or another file that has data as well as the appropriate header.

       When only the file name is provided, then data must be entered for a single period on the Rules Execution window.

       To load multiple periods, create a file for each period and append a period name or period key to the file name. When you execute the rule for a range of periods, the process constructs the file name for each period and uploads it to the appropriate POV.

Copyright © 2017, 2026, Oracle and/or its affiliates.

b. From **Directory**, specify the directory to which the file has been assigned.

   To navigate to a file located in a Data Integration directory, click **Select**, and then choose a file on the **Select** page. You can also select **Upload** on the **Select** page and navigate to a file on the **Select a file to upload** page.

   If you do not specify a file name, then Data Integration prompts you for the file name when you execute the rule.

c. To load data into multiple periods, in the **File Name Suffix Type** drop-down, select **Period Name** or **Period Key**.

   A suffix is appended to the file name, and Data Integration adds the file extension after adding the suffix. When you leave the file name blank, then the system looks for a file with a suffix. When the file name suffix type is provided, then the file name is optional in this case, and it is not required on the Rule Execution window.

   If the file name suffix type is a period key, the suffix indicator and period date format are required (as the suffix set) in the file name and must be validated as a valid date format. In this case, when you run the rule, **enter 1_.txt** in the file name field and select "Period Name" for the suffix indicator. Then run the rule for the January to March periods.

   For example, specify:

   i. 1_Jan-2019.txt

   ii. 1_Feb-2019.txt

   iii. 1_Mar-2019.txt

d. In **Period Key Date Format**, specify the data format of the period key that is appended to the file name in JAVA date format. (SimpleDateFormat).

e. Click **Save**.

19. Click **Save**.

20. Run the integration.

    In the Run Integration step, data is written to a flat file, which in turn is copied to a file repository. When data is written back, journal entries are created in the Oracle General Ledger of the Oracle ERP Cloud.

    For more information, see Running an Integration.

Writing Back Budgets to the Oracle General Ledger

If you want to report budget-to-actual from the Oracle General Ledger, you need to write back your budget to the Oracle General Ledger. If you want to validate spending online, you need to write back your budget to Budgetary Control.

Use this procedure to write back original and revised budgets prepared using the Planning feature to the Oracle General Ledger.

This procedure is not for writing back budget revisions prepared using the Budget Revisions feature in the Oracle Fusion Cloud Enterprise Performance Management, which automatically updates budget in both General Ledger and EPM type control budget in the Budgetary Control through another procedure.

The write back to Oracle General Ledger is also automatically performed for you when you write back the budget to the Budgetary Control for the EPM type control budget, but obviously only for the portion of your enterprise-wide budget that you write back to Budgetary Control.

For more information, see Using Financials for the Public Sector.

Copyright © 2017, 2026, Oracle and/or its affiliates.

To write back to the Oracle General Ledger:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. From the **Application** page, click ... to the right of the Cloud EPM source application, and then select **Application Detail**.
3. Select the **Options** tab.
4. From the **Balance Type** drop-down, select **Budget**.
5. In **Journal Source**, enter a description of the journal source that matches the journal source defined in Oracle ERP Cloud.
6. In **Journal Category**, enter a description of the journal category that the matches the journal category in Oracle ERP Cloud.
7. From the **Data Integration** home page, click +, then on the **Create Integration** page, create the integration between the source data to the target and then click **Save and Continue**.
   a. In **Name** and **Description**, enter a name and description for the new integration.
   b. In **Location**, enter a new location name, or pick an existing location to specify where to load data.
   c. From the **Source** ( ) drop-down, select the name of the Cloud EPM application from which to write back data.
   d. From the **Target** ( ) drop-down, select the Oracle ERP Cloud application.
   e. From **Category**, select the category mappings for categorizing and mapping source system data to a target Scenario dimension member.

      The categories listed are those that you created in the setup, such as "Actual." For more information, see Managing Category Mappings.
   f. **Optional**: Select any applicable location attributes for the integration. For more information, see Selecting Location Attributes.
   g. Click **Save and Continue**.
8. On the **Map Dimensions** page, map the dimensions in the Oracle ERP Cloud source application to the dimensions in the Cloud EPM application.

   If you are adding new dimensions or editing an existing dimension, complete the following:

   - In **Column**, specify the field number from the file to import.
   - In **Select Source Dimension**, specify the name of the source dimension to assign to the target application.

     Multiple source columns of the same dimension can be mapped to target dimensions. For example, you can map four "Account" source columns.
   - Add a source or target expression: assign an expression that operates on values directly from the source or target.

     See Using Source Expressions and Using Target Expressions.

   Be sure to map a source for the target dimension "Ledger."

Copyright © 2017, 2026, Oracle and/or its affiliates.

You can map a dimension like "Entity" to the ledger and define any necessary data load mapping to convert to the Oracle General Ledger name. If you are writing back to a single ledger, enter the name of the ledger in the expression column.

Optional: If you want to populate any additional reference data and/or attribute data for each journal, use the Attribute columns to map the columns.

Attribute columns, Attribute1 to Attrbute10, are reserved for REFERENCE1 TO REFERENCE10. In this case, you need to also add the REFERENCE column as a dimension and map it to the ATTR column in the target application. For example, when you want to populate REFERENCE3, then insert dimension details and give it an appropriate name, assign the type of Attribute, and then assign data column ATTR3. (ATTR11 to ATTR30 are reserved for ATTRIBUTE1 TO ATTRIBUTE20. Attribute1 is stored in ATTR11, Attribute2 is stored in ATTR12 and so on.)

For more information, see Mapping Dimensions.

9. From **Map Members** page, map any members from the source to target.

   To map all members to the Oracle ERP Cloud "as is" without any modification, select **All** for the **Mapping Type**, click **Add**, and on the **Add Map Member** page, in **Source**, enter: *****, and in **Target**, enter: *****.

   For more information, see Mapping Members.

10. Click **Save and Continue**.

11. From the **Data Integration** home page, and then from the **Actions** menu, select **Period Mapping**.

    The period mapping is used to convert periods to Oracle General Ledger accounting calendar periods for the transfer.

12. Select the **Application Mapping** tab.

13. From **Application Mapping** page, then **Target Application**, select the Oracle General Ledger application from the Oracle ERP Cloud to which to write back data.

14. Click + to add a separate row for each period that is to receive actual amounts and complete the following.

    > **Note**
    >
    > When specifying the period, the starting and ending periods should be within a single fiscal year. Providing date ranges that cross fiscal year results in duplicate data.

15. Define a **Period Key**, **Target Period Month** and **Target Period Year**.

    - **Period Key**—Specify the last day of the month to be mapped from the target system.

      Use the date format based on the locale settings for your locale. For example, in the United States, enter: the date using the MM/DD/YY format.

    - **Target Period Month**—The values in this field need to match the accounting calendar for the ledger in the Oracle General Ledger, which receives the transferred amounts.

    - **Target Period Year**—Use values that corresponds to the accounting period (as defined in the Target Period Month column).

    When you select a value, information about the period key, prior period key, period name, and the target period month are populated automatically.

16. Click **Save**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

17. Run the integration.

    For more information, see Running an Integration.

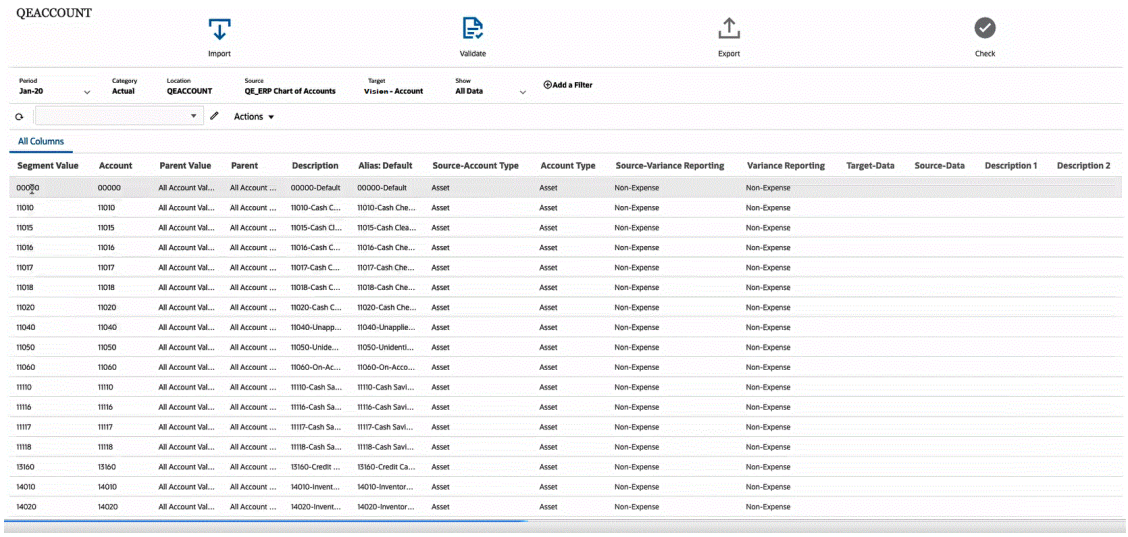
Integrating Metadata from the Oracle ERP Cloud

Push Oracle ERP CloudOracle General Ledger metadata to your Oracle Fusion Cloud Enterprise Performance Management applications

Oracle General Ledger metadata from the can be loaded to your Cloud EPM applications using Data Integration. The feature enables you to load the Oracle General Ledger hierarchy top node, segment value, description, parent, child, account type, etc.

To load the metadata, you simply pick the desired source ledger from the Oracle ERP Cloud using an Oracle ERP Cloud (Chart of Accounts) source adapter, set up a few simple mappings, and then push a button to pull the data into your Cloud EPM applications.

The following shows how Oracle General Ledger is loaded to a target in the Workbench.



Loading Metadata Process Description

At a high level, this is how you load metadata from a file to a Planning application:

1. Generate a metadata load file for each dimension to be loaded.

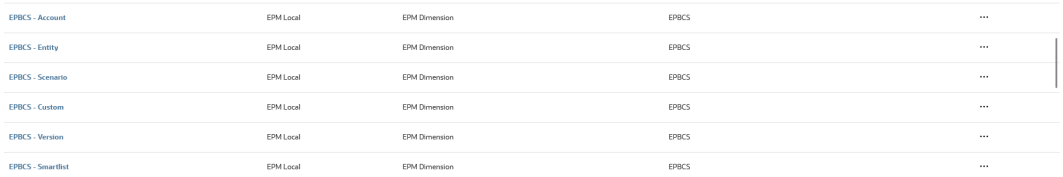
   A metadata load is only available to applications built on the Planning platform only.

   For information on how to construct a metadata load file, see Metadata Load File Considerations.

2. In **Application**, register a target application for the class of dimension or dimension type.

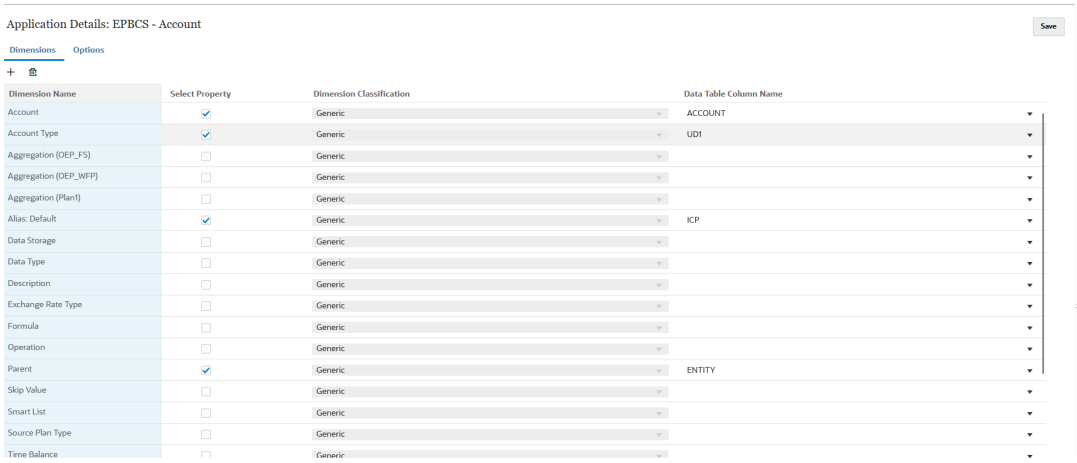
   Data Integration creates six-dimensions applications automatically: Account, Entity, Custom, Scenario, Version, and Smartlist.

Copyright © 2017, 2026, Oracle and/or its affiliates.



For more information on adding a dimension class or type as a target application, see Registering a Target Application for the Class of Dimension or Dimension Type.

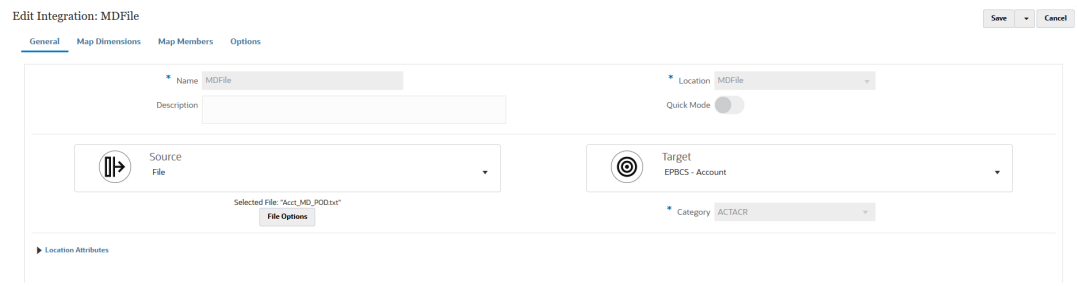
3. In **Application Details**, the system creates a record for each dimension property. Only the member name and parent are required fields.



To enable additional properties, add a row to the dimension metadata application. The name of the row is the property or attribute name used in the Planning application.

4. **Optional**: To add a custom dimension (one designated as Generic in the Planning application), in the target application, select the property name and enable the **Select Property** field, and then map it to a **Data Table Column Name** value. Next create a separate import format for each generic dimension. Then, in the dimension's data rule, specify the dimension name (for example, Product, Movement) in the Dimension name of the data rule's target options.

5. In **Create Integration**, create an integration between the metadata file and the target application.



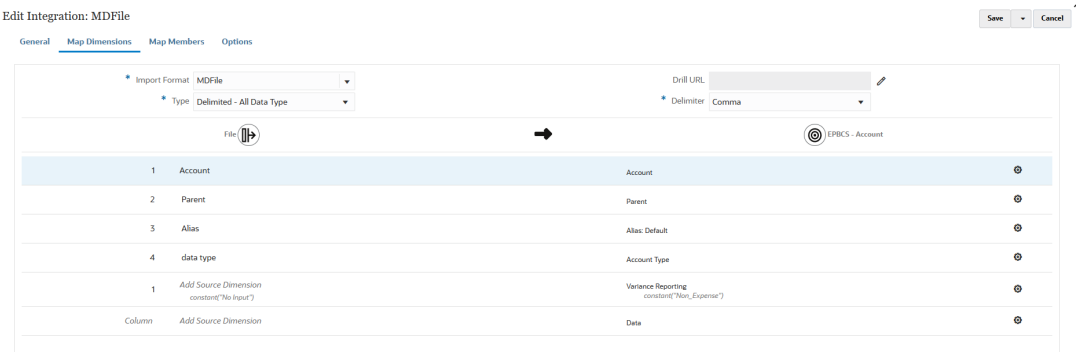
For more information, see Creating File-Based Integrations.

6. In **Map Dimensions**, map the data from the metadata load file to the properties of the dimensions in the Oracle Fusion Cloud Enterprise Performance Management application.

Copyright © 2017, 2026, Oracle and/or its affiliates.

This allows users to import dimension members from any file format. (The file must be "delimited - all data type" file type.)

Properties are added as "dimensions" of a dimension application. For example, the Two Pass Calculation property of Entity is added as a dimension and the flat file adds the "yes" or "no" property on the load.



> **Note**
>
> Checked dimensions in the dimension "application" are the ones loaded. If you do not map them, the load will fail. There is no default if a mapping is missing. To avoid loading a "field" such as alias, uncheck the check box in the target application. To supply a single value for all loaded rows, specify the value in the Expression field and map *to* for that dimension.

For more information, see Creating the Dimension Maps.

7. On the **Run Integration** page, from **Export Mode**, select **Merge** and then select all other required parameters.

   You can execute the integration or one or more periods. You then verify that the data was imported and transformed correctly, and then export the data to the target application.

   Running an Integration.

8. You can also to import and export metadata using the Pipeline. For more information, see Using an Import Metadata Job Type and Using an Export Metadata Job Type.

Integrating Oracle ERP Cloud Data

Pull a subset of data from the Oracle ERP Cloud without connecting directly to Oracle Fusion Financials sources.

You can pull a subset of data from the Oracle ERP Cloud using Data Integration as the integration mechanism without connecting directly to Fusion Financials sources. Using this feature, you can import from data sources from the Oracle Financials Cloud or Supply Chain.

To do this, you use a data source adapter-based framework that queries data from Oracle Business Intelligence Publisher reports as a data source. BI Publisher extracts Oracle ERP Cloud data directly from the Fusion Financials database tables. Any kind of record in the Oracle ERP Cloud can be included in or considered as the basis of the query. When the system brings over the data into Data Integration, the data and metadata can be subsequently mapped and loaded to the Oracle Fusion Cloud Enterprise Performance Management.

Copyright © 2017, 2026, Oracle and/or its affiliates.

You can use either pre-packaged queries or customized BI Publisher reports to define your own report parameters for extracting the data from the Oracle ERP Cloud.

For more information about integrating Oracle ERP Cloud with Cloud EPM using Data Integration, see Overview: Integrating Oracle Cloud ERP with Oracle Cloud EPM.

Process Description for Integrating Oracle ERP Cloud Data Using Prepackaged Queries

Data Integration ships with prepackaged queries that leverage seeded data extracts provided by Oracle ERP Cloud as data sources.

For more information about Oracle Business Intelligence Publisher, see Oracle Business Intelligence Publisher 12.2.1.3.0.

These are the steps for loading data from the Oracle ERP Cloud using the prepackaged queries shipped with Data Integration.

1. An Oracle ERP Cloud integration requires that you have the privileges or user role and data access to the Oracle ERP Cloud. For more information, see Security Role Requirements for Oracle ERP Cloud Integrations.
2. Register the source system for the source system type **Oracle ERP Cloud** and specify your user credentials.

   This step includes specifying the connection details and testing the connection.

   For more information, see Configuring an Oracle ERP Cloud Connection.
3. Register the application as an Oracle ERP Cloud data source application type and save it.

   The source columns are populated automatically from the uploaded file extract.

   See Registering Oracle ERP Cloud Applications.
4. Provide any input values in integration options in Application Filters.

   > **Note**
   >
   > Define any necessary filters to limit the amount of data returned by the BI Publisher extract. Filters ensure the best load performance. Oracle recommends you use the fully Qualified name for a Ledger in the Source Filter.
5. Set up the integration mapping between the Oracle ERP Cloud data source and the target application by building an import format.

   See Mapping Dimensions.
6. Define the location used to associate the import format.
7. Map dimensions between the source and target.

   See Mapping Dimensions.
8. Map members from the source to target.
   See Mapping Members.
9. Select any source and target options.

   See Setting Data Integration Options.
10. Run the integration.

Copyright © 2017, 2026, Oracle and/or its affiliates.

See Running an Integration.

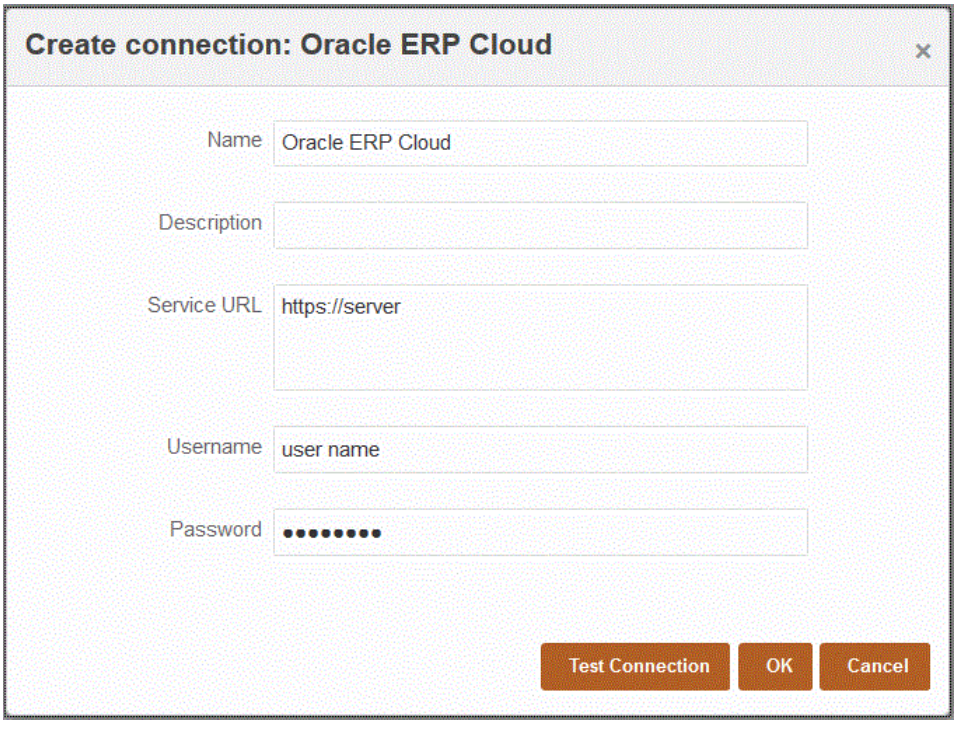
Configuring an Oracle ERP Cloud Connection

Oracle ERP Cloud source connections can be used to register and manage the following data sources:

- Oracle ERP Cloud
- Oracle ERP Cloud (Receivables Transactions)
- Oracle ERP Cloud (Trial Balance Average)
- Oracle ERP Cloud (Custom)
- Oracle ERP Cloud (Payables Transactions)
- Oracle ERP Cloud (Trial Balance)
- Project Management
- Budgetary Control commitments, obligations, expenditures, and write-back data sources. Budget Review data sources

To create an Oracle ERP Cloud connection:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, click (Configure Connections icon).
3. On the **Connections** page, click (Add drop-down icon) drop-down, select **Oracle ERP Cloud**.



4. In **Name**, enter the source system name.
5. In **Description**, enter a description of the source system.

Copyright © 2017, 2026, Oracle and/or its affiliates.

6. In **Service URL**, enter the server information for the web services.
7. In **Username**, enter the Oracle ERP Cloud user name.

   Enter the name of the Oracle ERP Cloud user who launches the process requests to send information between Oracle Fusion Cloud Enterprise Performance Management and the Oracle ERP Cloud. This user must have an assigned Oracle General Ledger job role such as "Financial Analyst," "General Accountant," or "General Accounting Manager."
8. In **Password**, enter the Oracle ERP Cloud password.

   You must update this password anytime you change your Oracle ERP Cloud password.
9. Click **Test Connection**.

   When the connection has been tested correctly, the information message "Connection to [*source system name*] successful" is displayed.
10. Click **OK**.

Registering Oracle ERP Cloud Applications

You can extract non-General ledger data from Oracle ERP Cloud data sources and load it to the Oracle Fusion Cloud Enterprise Performance Management. These types of data sources let you define an Oracle Business Intelligence Publisher report extract that is referenced by Data Integration to pull data directly from Oracle ERP Cloud sources like payables, receivables, fixed assets, and supply chain.

Data Integration provides prepackaged queries to the Oracle ERP Cloud sources including:

- Oracle ERP Cloud (Payable Transactions)
- Oracle ERP Cloud (Receivable Transactions)
- Oracle ERP Cloud (Trial Balance - Average)
- Oracle ERP Cloud (Trial Balance)

In addition, you can create and run a custom query against a BI Publisher data extract. In this case, use an Oracle ERP Cloud (Custom) data source adapter in Data Integration to import the source CSV file from BI Publisher, and then set the report parameters to use.

> **Note**
>
> An integration with the Oracle ERP Cloud requires that you have the privileges or user role and data access to all ERP ledgers. For more information, see Security Role Requirements for Oracle ERP Cloud Integrations.
> You must register source system for the Data Integration data sources using the system type **Oracle ERP Cloud** and specify your user credentials. This step includes specifying the connection details and testing the connection. For more information, see Configuring an Oracle ERP Cloud Connection.

To register an Oracle ERP Cloud application:

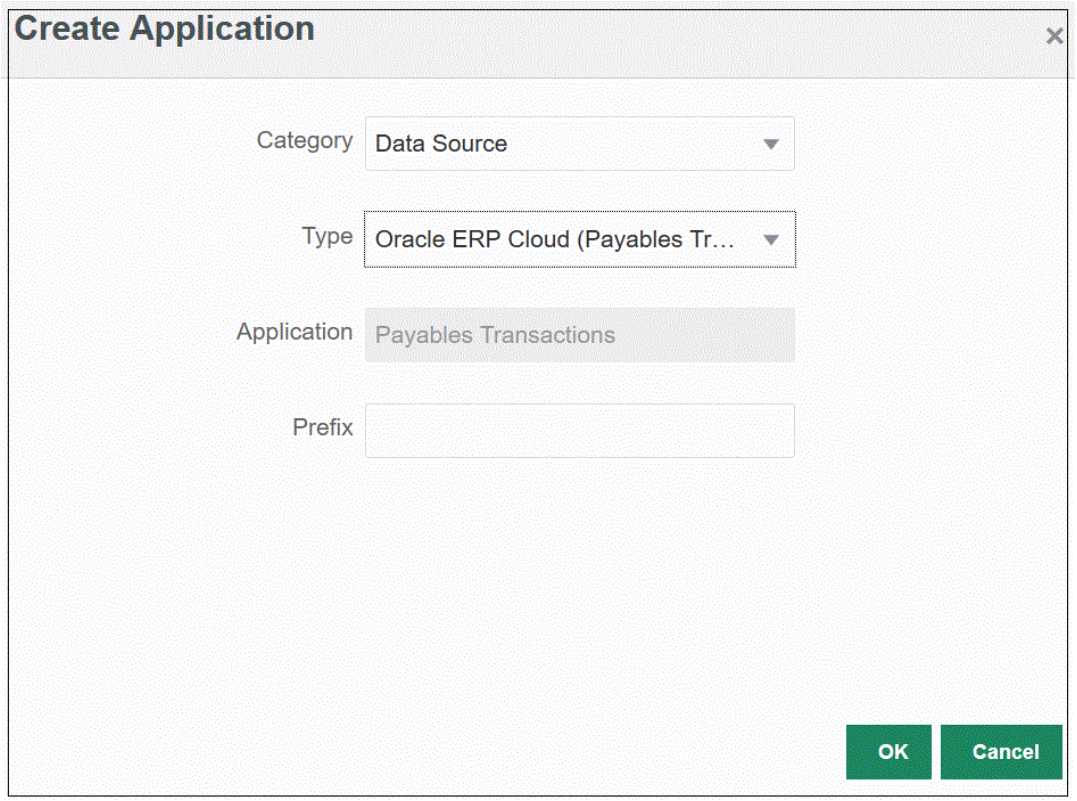
1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, click (Add icon).
3. From **Category**, select **Data Source**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

4. From **Type**, select the type of prepackaged Oracle ERP Cloud query.

   Available Oracle ERP Cloud types:

   - Oracle ERP Cloud (Payable Transactions)
   - Oracle ERP Cloud (Receivable Transactions)
   - Oracle ERP Cloud (Trial Balance - Average)
   - Oracle ERP Cloud (Trial Balance)



   The name of the prepackaged application that displays in the **Application** depends on the type of prepackaged Oracle ERP Cloud query selected in this field.

5. **Optional**: In **Prefix**, specify a prefix to make the application name unique.

   The prefix is concatenated with the application name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.

6. Click **OK**.

Applying Application Filters to an Oracle ERP Cloud Data Source

All application filters are predefined when you select any of the prepackaged queries to the Oracle ERP Cloud sources. However, you can provide input values in integration options.

You can select dynamic filters to define as report parameters from the Oracle ERP Cloud data source when the actual parameter value needs to be defined at the run integration or application level.

Copyright © 2017, 2026, Oracle and/or its affiliates.

An example of a dynamic filter is "Currency Type" where you can select either: Entered, Statistical, or Total.

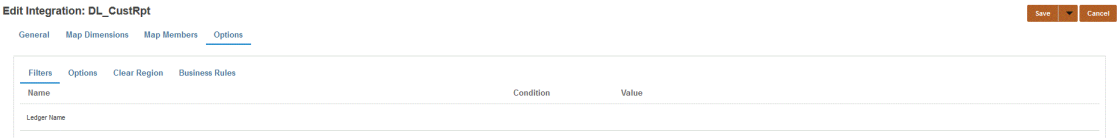
You can specify a single filter condition or multiple filters conditions, and additionally specify the exact values that you want returned.

In some cases, you can change a static parameter value in the Report parameter list by replacing it with a parameter value enclosed within $$ notations. This type of filter applies to the Ledger ID and Period parameters.

For example, you can add the static parameter value `argument1 = $LEDGER_NAME$` to the Report Parameter List as a parameter.

On the Edit Options page, a display name has been entered for the parameter. This is the name as it is shown on the Options page:

This is how the parameter shows on the Options tab in the integration definition:



Two predefined parameters, $START_PERIODKEY$ and $END_PERIODKEY$, can be used to select specific periods using a data format from the POV. For information on using these two parameters, see Selecting Period Report Parameters from the Oracle ERP Cloud.

To add a filter for the Oracle ERP Cloud data source.

1. From the **Application** page, click ... to the right of the Oracle ERP Cloud data source and then select **Edit Options**.
2. From the **Edit Options** page, click **Add** .

   A blank entry line display.
3. Select any parameters to pass by completing the following:
   a. In the **Name** field, specify the name of the parameter.
   b. In **Display Prompt**, enter the name of the display prompt for the filter on the **Source Options** tab in Data Management or **Edit Integration** page in Data Integration.
   c. In **Display Order**, specify the display order of the filter on the **Source Options** or **Edit Integration** page.

      For example, enter **99** to show the filter in the 99th position sequence or position in a list of filters. Display orders are listed from lowest to highest.

      If this field is blank, the custom filter cannot be displayed, and the default value is used as the filter value.
   d. In **Property Level** drop-down, select display level of the parameter (application, integration or both application and integration) to indicate the level at which the filter is displayed.

Copyright © 2017, 2026, Oracle and/or its affiliates.

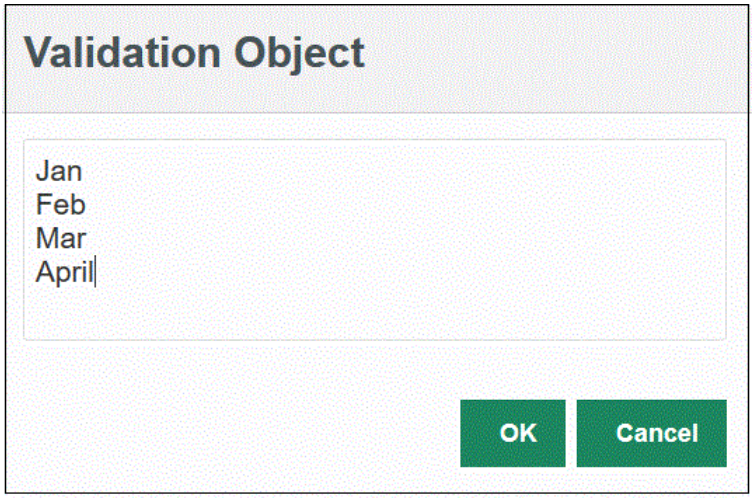
e. To provide a drop-down of values listed using the lookup type, in **Validation Type**, select one of the following

- None
- Number
- Lookup Validate
- Lookup No Validate
- Yes/No
- Date
- Query
- Choice list—You can enter a list of values in the Validation Object field. Each value is entered on a new line. The list of values (LOV) on the Option page shows the values as a list.

For example, to set the Canceled Invoices Only option to "No," select **No** in the parameter field for **Canceled Invoice Only**.

f. In **Validation Object**, enter a list of lookup objects to be validated by type.

To enter a list of values if you chose the **Choice list** validation type above, click , then enter the list of values on the Validation Object page, and then click **OK**.



4. In **Condition List**, provide a condition list based on one or all the following:

- EQ (Equal)
- IN
- Like

The condition values can be `Equal`, `Like`, or `In`. The condition values are stored as a drop-down list in CONDITION_LIST field. If the value is `EQ,IN`, then show only `Equal` and `In` in the Condition drop down. If the value is `EQ LIKE`, then show only Equal, Like in the drop down. Any combination of `EQ,IN,LIKE` are stored in the field. If only one value is provided in the Condition List, then the condition cannot be changed. The default value is `EQ`. When specifying multiple conditions, prefix the condition with a comma. For example, to use the `IN` and `LIKE` conditions, enter: `,IN,LIKE`

Copyright © 2017, 2026, Oracle and/or its affiliates.

5. Click **Save**.

Selecting Period Report Parameters from the Oracle ERP Cloud

When importing data from the Oracle ERP Cloud, you can select the period from which to pull the data by specifying a date (using a date specific format) in the $START_PERIODKEY$ and $END_PERIODKEY$ notations.

Valid values for date format include:

- dd: Day of the Month
- MM or MMM: Month
- yy or yyyy: Year

Specify the From Entered Date and To Entered Date in the format: yyyy-MM-dd. To import data from January 1, 2021 to January 31, 2021, you enter **2021-01-01** as the report parameter in the From Entered Date field and **2021-01-31** as the report parameter in the To Entered Data field.

If you want to specify a single-period load, select the $START_PERIODKEY$ notation to indicate the period key from which the data load rule is run. The data is imported for the period specified in the $START_PERIODKEY$ notation. Source period mappings are not needed for single-period loads.

You can also change the "ACCOUNTING PERIOD NAME" by selecting the actual period using the $START_PERIODKEY[MM-yy]$ format.

When you a run multi-period load, data is imported for the range and must be specified in the START_PERIODKEY and END_PERIODKEY parameter list. In order for the system to load the data into the right periods, source period mappings must exactly match the Year and Period columns in the data extract.

Multi-period imports are available if the report accepts a period as a range. If the report accepts only the period name (START_PERIODKEY parameter), no multi-period imports are available.

Process Description for Integrating Oracle Cloud ERP Data Using a Custom Query

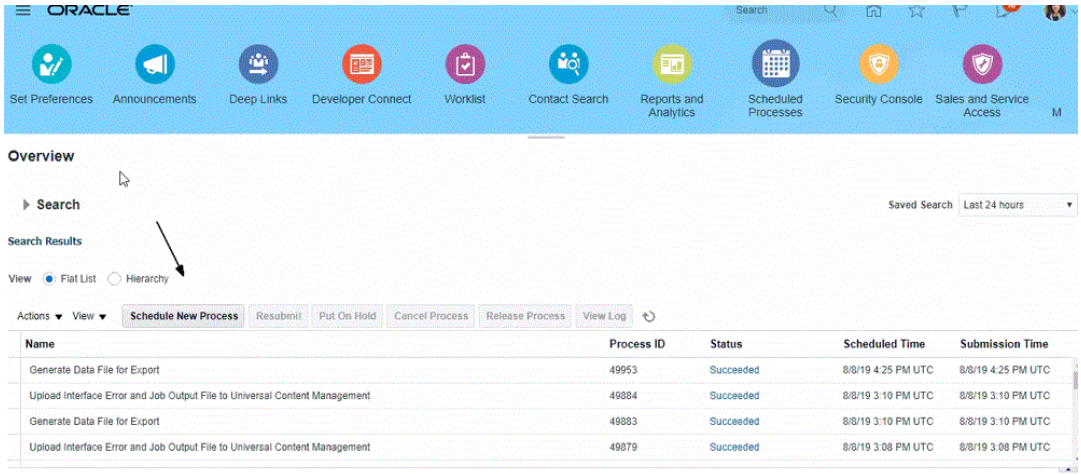
You can fetch data from the Oracle ERP Cloud using a custom query and then load it into Oracle Fusion Cloud Enterprise Performance Management. The custom query is run against any report from Oracle Business Intelligence Publisher that creates a CSV formatted data file and then loads the data to the Cloud EPM. In this case, Data Integration executes the report to extract the data and loads it to Cloud EPM.

These are the steps for loading data from the Oracle ERP Cloud into Cloud EPM using BI Publisher report extracts using a custom query.

> **Note**
>
> To create a custom Oracle ERP Cloud integration definition with the Cloud EPM and register it as an ESS job, see Registering a BI Publisher Report as an Oracle Enterprise Scheduler (ESS) Job.

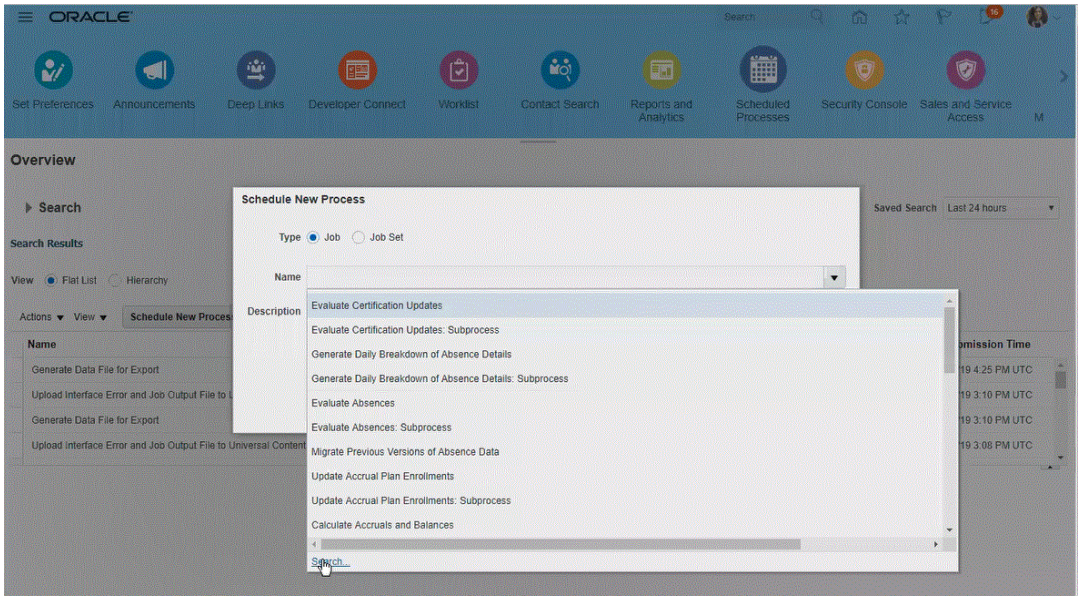
Copyright © 2017, 2026, Oracle and/or its affiliates.

1. An Oracle ERP Cloud integration requires that you have privileges or user role and data access to all ERP ledgers to be integrated. For more information, see Oracle ERP Cloud security role requirements, see Oracle ERP Cloud Securing ERP.
2. Navigate to the Oracle ERP Cloud and execute a BI Publisher report by clicking **Schedule New Process**.



3. From the **Search and Select** page, in **Name**, select a report or extract, and click **OK**.

   You can select any BI Publisher report if it produces an output file in a CSV format file. Not all reports in Fusion produce a CSV format file.



For example, enter **Trial** to search for a Trial Balance Report.

Copyright © 2017, 2026, Oracle and/or its affiliates.

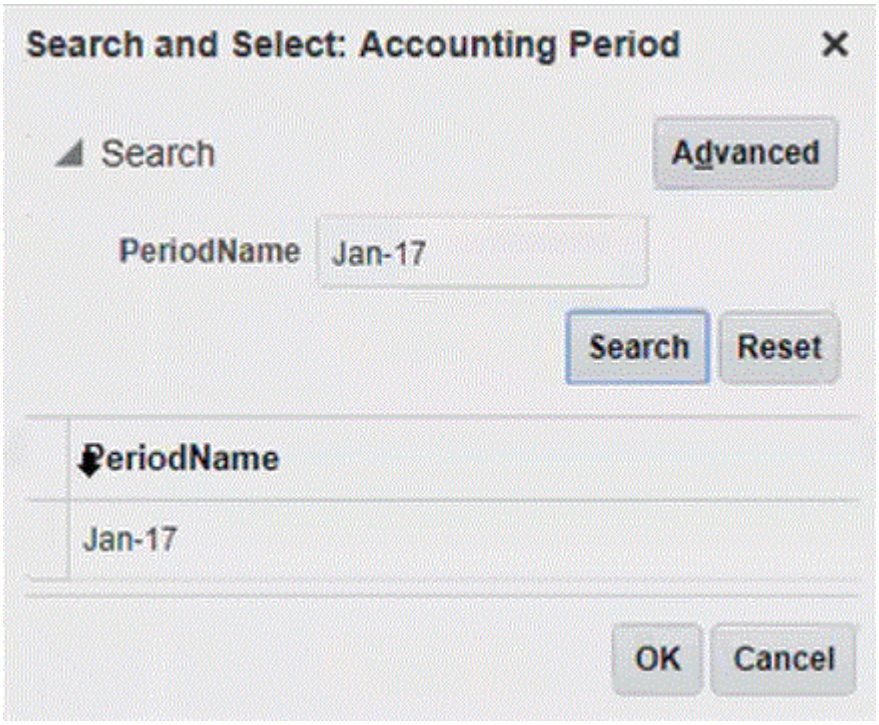
4. From **Process Details**, select the parameters for the extract or report, and click **Submit**.

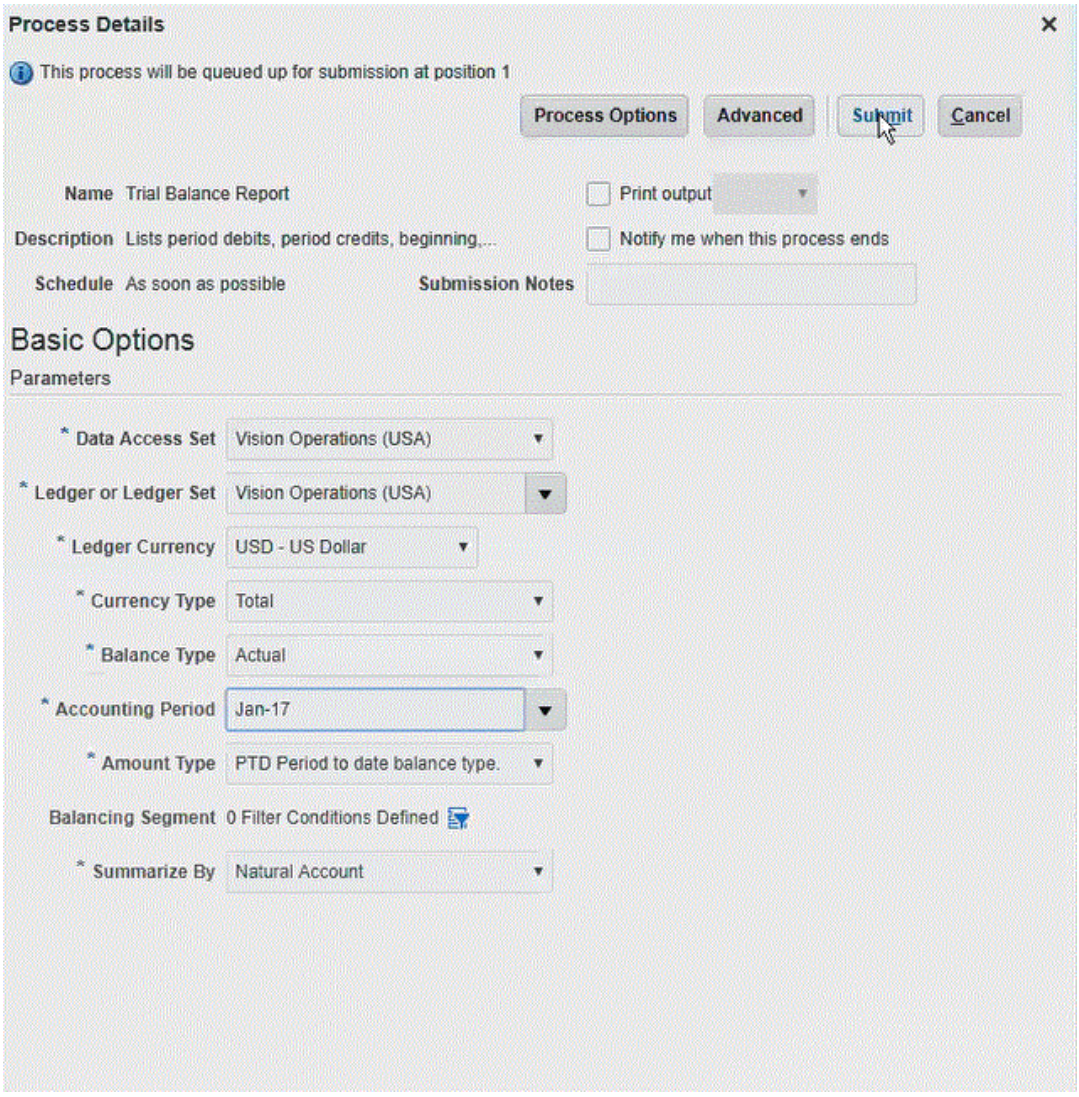
   In the following example, "Ledger" is Vision Operations and "Amount type" is YTD or PTD.

   Be sure to specify the Accounting Period. The Accounting Period is the parameter that will be set up in Data Integration so that the report can be reused.

> **Note**
>
> The Oracle ERP Cloud and Cloud EPM integration will fail unless the selected extract on the Oracle ERP Cloud side has one or more bind parameters passed from the Cloud EPM. The bind parameter is a placeholder for actual values in the SQL statement. Bind parameters must be enclosed in tilde (~~) characters. For example, to use "Period" as a bind parameter specify: ~PERIOD~. The name must exactly match the name specified in the SQL query.
>
> To do this, create a bind parameter directly in the report, which is not referenced in the Data Model query. In Data Integration, specify a random string such as "ABC" in the Report Parameter List that will be passed to the bind parameter you created in the report definition.

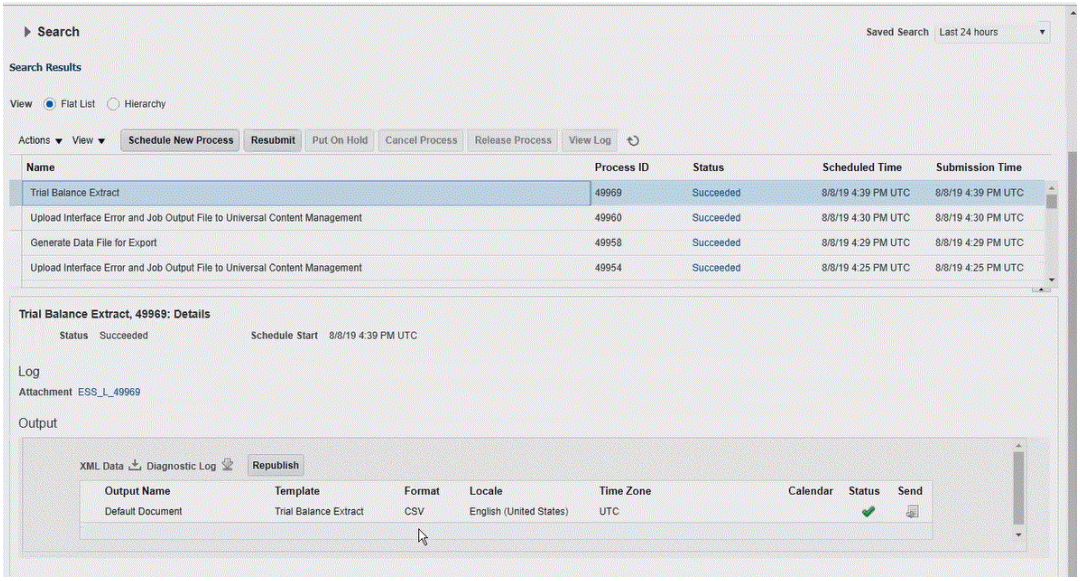
Copyright © 2017, 2026, Oracle and/or its affiliates.



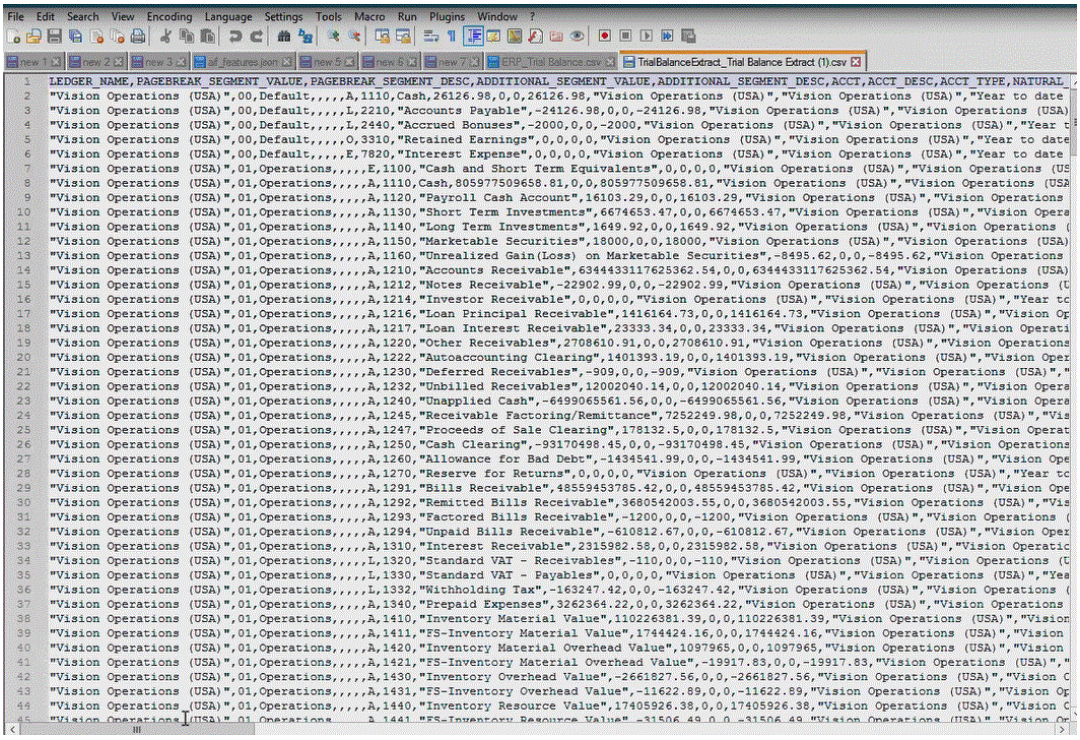


Copyright © 2017, 2026, Oracle and/or its affiliates.

When the report has been generated, the Output section shows the results of the submission.



5. Click **Republish**, and then from the **report output** page, click **csv**.
6. Select the CSV output file, right click it, and then select **Open**.



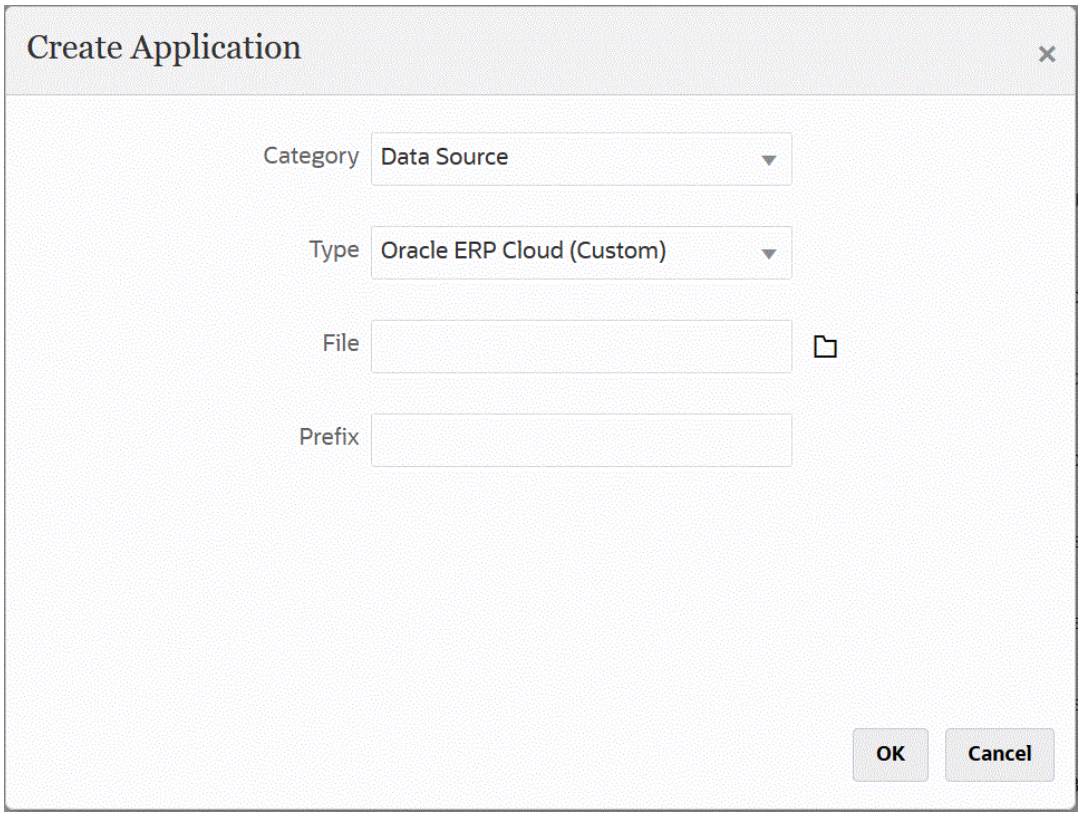
7. Save the report locally to your file system.

   Rename the downloaded output file as *Appname*`.csv` where *Appnam* is the intended application name for the "Data Source" application in Data Integration, which represents the BI Publisher report extract.

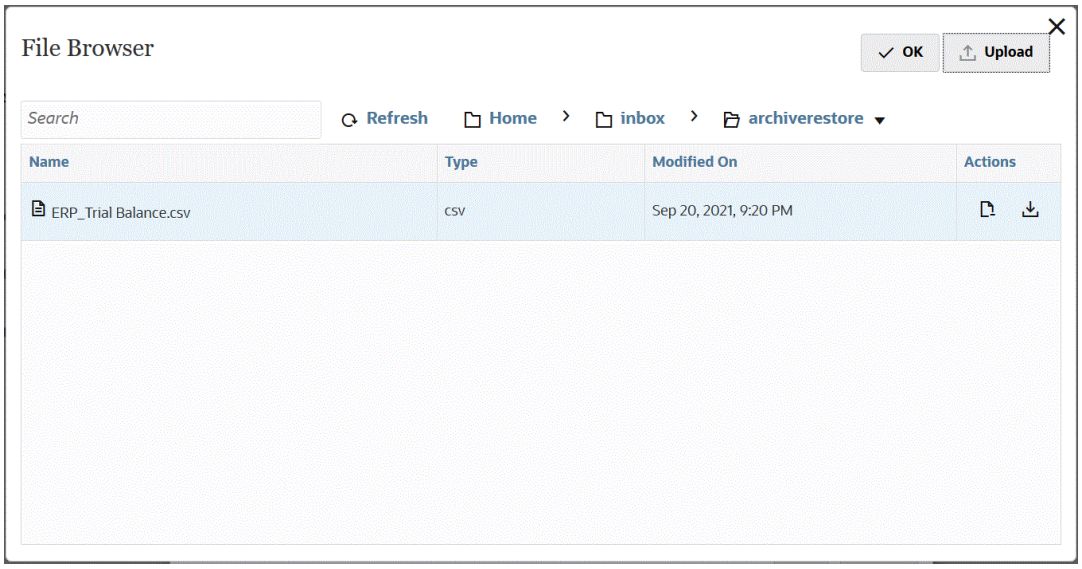
8. From the **Data Integration** home page, and then **Actions**, select **Applications**.
9. On the **Applications** page, click + (Add icon).

Copyright © 2017, 2026, Oracle and/or its affiliates.

10. From **Category**, select **Data Source**.
11. From **Type**, select **Oracle ERP Cloud (Custom)**.



12. From **File**, click and navigate to the folder where you saved the CSV file, select it and then click **OK**.



The report is saved as the target application and the Application Name is populated automatically.

13. In **Prefix**, specify a prefix to make the application name unique.

Copyright © 2017, 2026, Oracle and/or its affiliates.

The prefix is concatenated with the file name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.

14. Click **OK**.
15. Click **Save**.

    Data Integration registers the application and returns all columns in Dimension Details.

16. On the **Application** page, click ... next to data source application and then select **Application Details**.
17. In **Connection Name**, specify the name of the source system.

    For example, if the name of your source system is "ERP Cloud," specify **ERP Cloud.**

    You can also use an Oracle ERP Cloud or the GL source system name or define a new one.

18. **Execution Method**—Specify the method for executing the job.

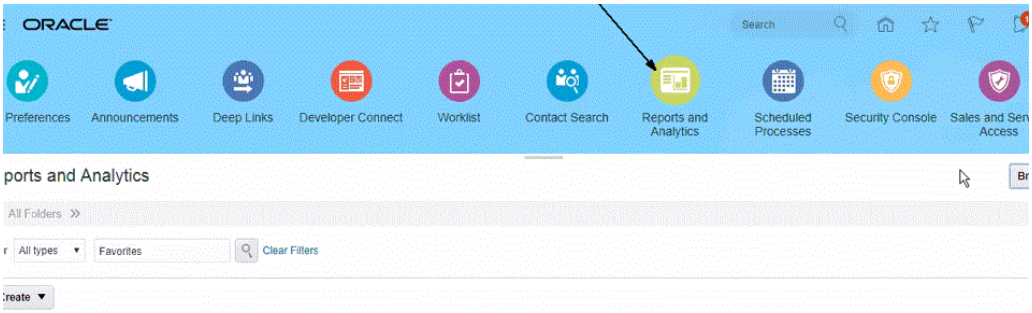
    Valid options:

    - **BIP Report**—Executes the BI Publisher report in synchronous mode, which has a timeout limit of about 5 minutes within the Oracle ERP Cloud instance. This method is suitable for smaller data sets or for queries that execute quickly.
    - **ESS Job**—Executes the BI Publisher report in asynchronous mode, which has no execution time restrictions.

19. **Report Name**—Enter the name of the report and the complete report path when the execution method is **BIP Report**. For example, enter */Custom/MyReport.xdo*. Leave this field blank when the execution method is **ESS Job**.

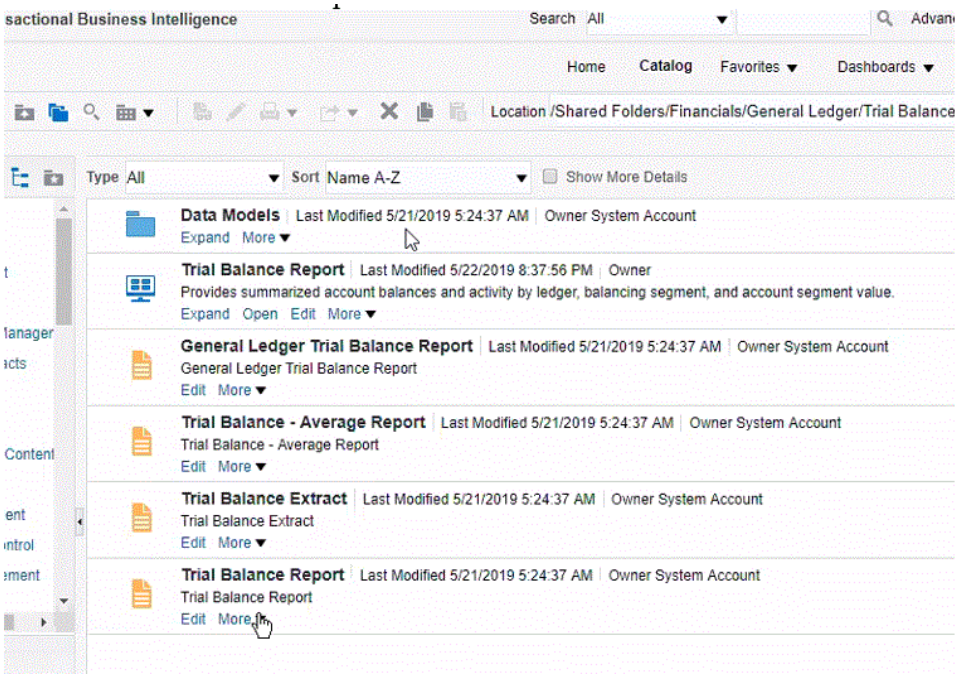
    If you need to find the name of the report in the Oracle ERP Cloud, complete the following steps:

    a. Navigate to **Oracle ERP Cloud**, find the report, and select **Reports and Analytics** to retrieve the parameter information.

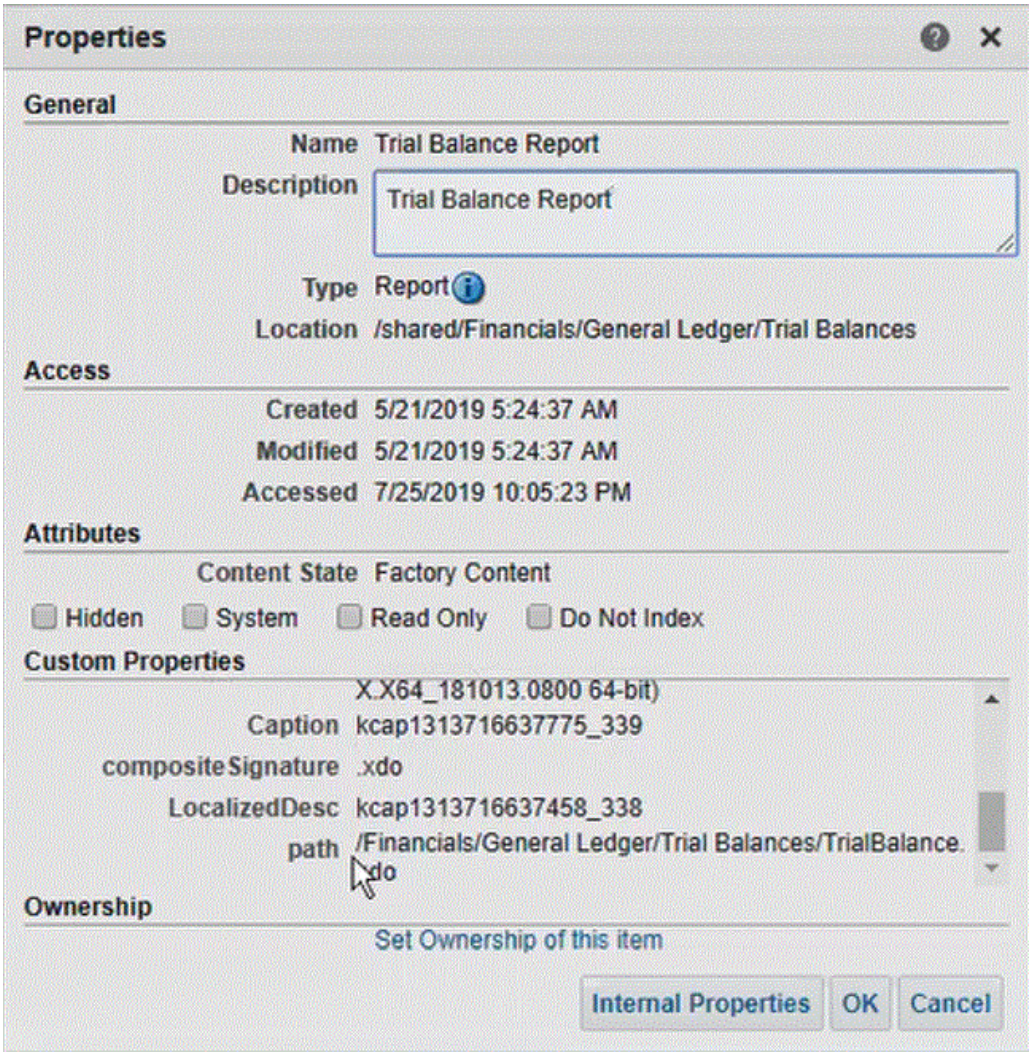
    

    b. Click **Browse Catalog**.
    c. Locate and select the extract or report.

Copyright © 2017, 2026, Oracle and/or its affiliates.



d. Click **More** and then **Properties**.



Copyright © 2017, 2026, Oracle and/or its affiliates.

e. In the **Custom Properties** section, scroll down to the **path** field.

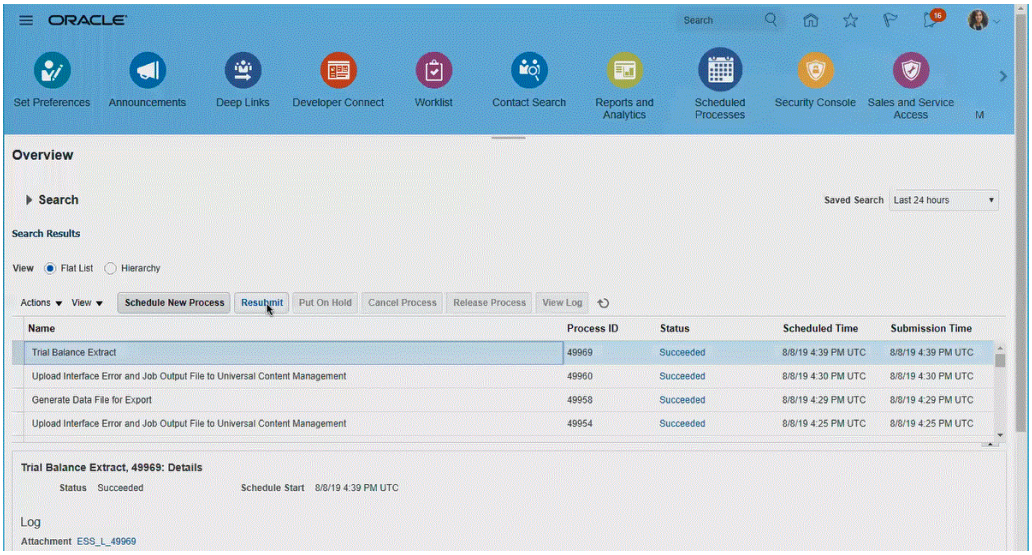
f. Copy the **path** (and name) and paste it to the **Report Name** field when registering the target application in Data Integration.

20. Return to Data Integration and in the **Report Parameter list**, specify the report parameters of the custom query.

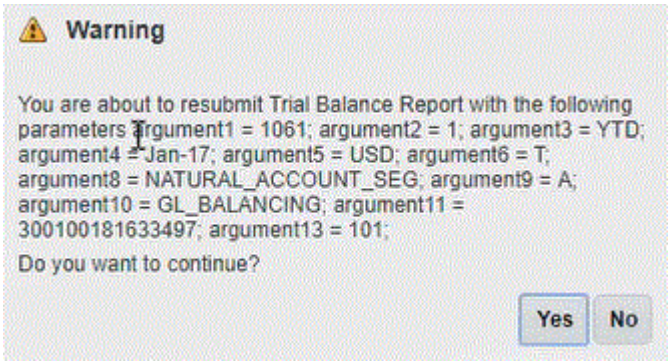
If you need to identify report parameters, complete the following:

a. Navigate to **Oracle ERP Cloud**, and from the **Overview** page, select the report and click **Resubmit**.

This step enables you to view and capture the report parameters defined in the BI Publisher extract or report.



A list of report parameters is generated.



b. Copy the report parameters shown in the **Warnings** window.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> If a #NULL (null) parameter is passed as a BI Publisher parameter from the Oracle ERP Cloud to the Cloud EPM, the #NULL parameter causes the ERP adapter-based framework to fail.
> To fix this issue, remove any parameters that have a value of #NULL in the list of parameters, and just leave the value empty, and remove spaces.
>
> Here's an example:
>
> If your current parameters show:
>
> `argument1=30029384;argument2=#NULL;argument3=01-JAN-2022`
>
> Then change `argument2=#NULL` to show:
>
> `argument1=30029384;argument2=;argument3=01-JAN-2022`

c. Navigate to Data Integration and paste the report parameter list from the **Warnings** window into the **Report Parameter List** field of your custom query.

   Make sure you specify a random string such as "ABC" in the "Report Parameter List" that will be passed to the bind parameter you created in the report definition. If you create a report with a query that doesn't have bind parameters passed from the Cloud EPM, the process will fail on the Cloud EPM side.

21. In Data Integration, set up the integration mapping between the Oracle ERP Cloud data source and the target application by building an import format.

    See Mapping Dimensions.

22. Define the location used to associate the import format.

23. Map members from the source to target.
    See Mapping Members.

24. Select any source and target options.

    See Setting Data Integration Options.

25. Run the integration.

    See Running an Integration.

Registering a BI Publisher Report as an Oracle Enterprise Scheduler (ESS) Job

When you build an Oracle Business Intelligence Publisher report extract as the basis for an Oracle Fusion Cloud Enterprise Performance Management integration definition with the Oracle ERP Cloud, you can execute the integration without encountering execution time restrictions. To do this, register the BI Publisher report as an Oracle Enterprise Scheduler (ESS) job as part of the integration definition.

Process Description for Registering ESS Jobs

The following is a high level description on how to register Oracle Business Intelligence Publisher report extracts as ESS jobs.

1. Define the Data Extract SQL.

Copyright © 2017, 2026, Oracle and/or its affiliates.

The source for the BI Publisher report is a data model which defines the source query and the parameter definition. Before an extract may be defined, the source extract SQL must be defined.

For more information, see Defining the Data Extract Process.

2. Create the data model for the BI Publisher report extract.

   For more information, see Creating the Data Model for the BI Publisher Report Extract.

3. Create the BI Publisher report extract.

   For more information, see Creating the BI Publisher Report Extract.

4. Create the Oracle Enterprise Scheduler (ESS) job to run in asynchronous mode.

   This step is only required when running the extract in asynchronous mode. You create a custom ESS job for the extract report.

   For more information, see Creating an Oracle Enterprise Scheduler (ESS) Job.

5. Create an integration in the Oracle Fusion Cloud Enterprise Performance Management using the BI Publisher report extract as the source.

   For more information, see Creating the Integration Definition Between Cloud EPM and the BI Publisher Report Extract.

Defining the Data Extract Process

The source for the Oracle Business Intelligence Publisher report is a data model, which defines the source query and the parameter used in the query. Before an extract may be defined, the source extract SQL must be defined.

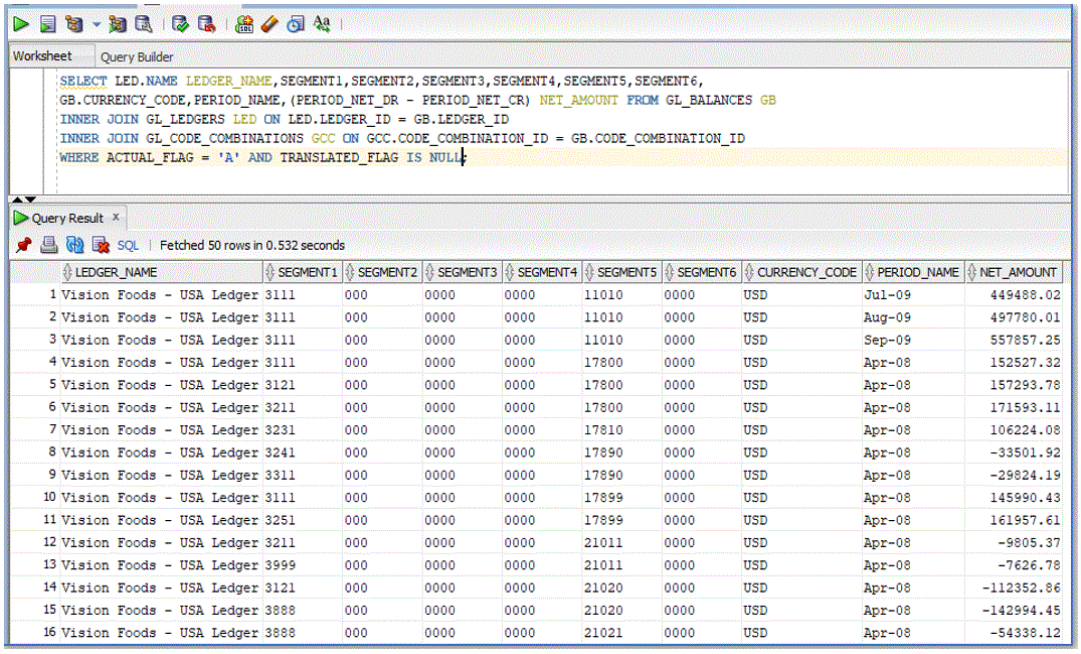
The following example shows a sample query that extracts the net activity for a given period and ledger from the GL_BALANCES table. Any table in the source may be specified, but for clarity and ease of use, an example for GL balances is provided here.

To define the data extract process:

1. Use a SQL tool to build the query.

   In the example, a SQL tool is used to build the query to make it easy to debug the results before using it in BI Publisher. If a tool to perform this step is unavailable, you can just enter the query in BI Publisher. This is the base query, and additional bind variables are included so that period and ledger name may be passed from the Oracle Fusion Cloud Enterprise Performance Management as part of the extract process.

Copyright © 2017, 2026, Oracle and/or its affiliates.

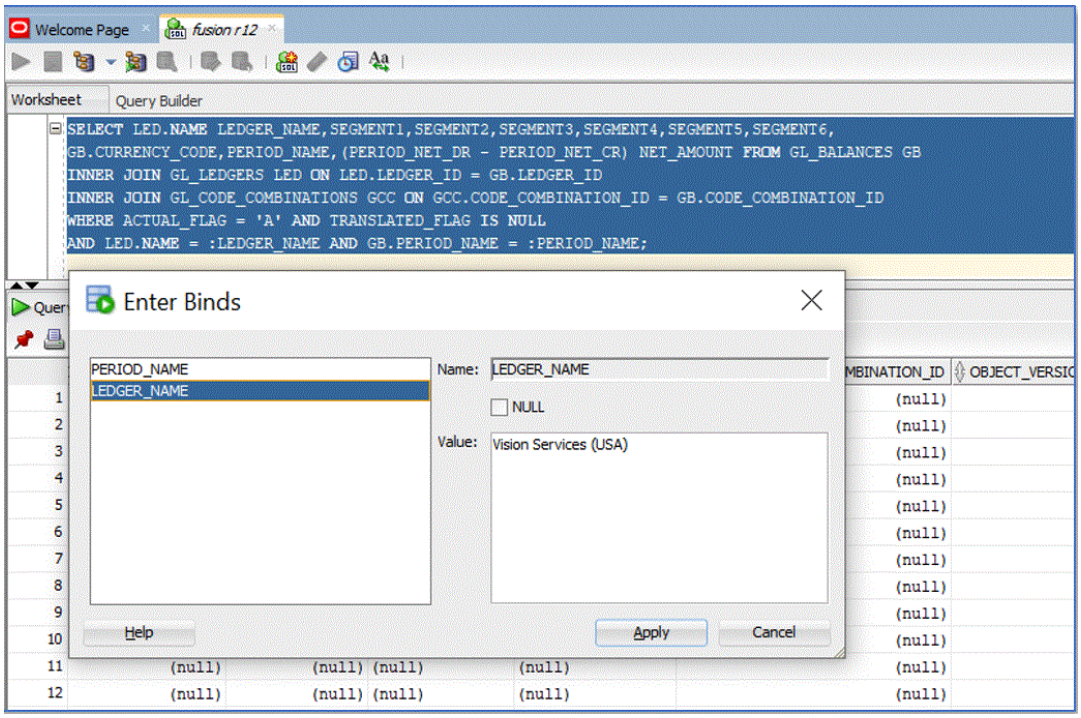


	LEDGER_NAME	SEGMENT1	SEGMENT2	SEGMENT3	SEGMENT4	SEGMENT5	SEGMENT6	CURRENCY_CODE	PERIOD_NAME	NET_AMOUNT
1	Vision Foods - USA Ledger	3111	000	0000	0000	11010	0000	USD	Jul-09	449488.02
2	Vision Foods - USA Ledger	3111	000	0000	0000	11010	0000	USD	Aug-09	497780.01
3	Vision Foods - USA Ledger	3111	000	0000	0000	11010	0000	USD	Sep-09	557857.25
4	Vision Foods - USA Ledger	3111	000	0000	0000	17800	0000	USD	Apr-08	152527.32
5	Vision Foods - USA Ledger	3121	000	0000	0000	17800	0000	USD	Apr-08	157293.78
6	Vision Foods - USA Ledger	3211	000	0000	0000	17800	0000	USD	Apr-08	171593.11
7	Vision Foods - USA Ledger	3231	000	0000	0000	17810	0000	USD	Apr-08	106224.08
8	Vision Foods - USA Ledger	3241	000	0000	0000	17890	0000	USD	Apr-08	-33501.92
9	Vision Foods - USA Ledger	3311	000	0000	0000	17890	0000	USD	Apr-08	-29824.19
10	Vision Foods - USA Ledger	3111	000	0000	0000	17899	0000	USD	Apr-08	145990.43
11	Vision Foods - USA Ledger	3251	000	0000	0000	17899	0000	USD	Apr-08	161957.61
12	Vision Foods - USA Ledger	3211	000	0000	0000	21011	0000	USD	Apr-08	-9805.37
13	Vision Foods - USA Ledger	3999	000	0000	0000	21011	0000	USD	Apr-08	-7626.78
14	Vision Foods - USA Ledger	3121	000	0000	0000	21020	0000	USD	Apr-08	-112352.86
15	Vision Foods - USA Ledger	3888	000	0000	0000	21020	0000	USD	Apr-08	-142994.45
16	Vision Foods - USA Ledger	3888	000	0000	0000	21021	0000	USD	Apr-08	-54338.12

2. Add the bind parameter values to pass along with the query when it is executed,

   The integration to Oracle ERP Cloud will fail unless the selected extract on the Oracle ERP Cloud side has one or more bind parameters passed from the Cloud EPM.

   Here two bind parameters have been added to the query:



   Here is a sample of the output from the query for the ledger named "Vision Services (USA)" and period "Jul-20":

Copyright © 2017, 2026, Oracle and/or its affiliates.

	LEDGER_NAME	SEGMENT1	SEGMENT2	SEGMENT3	SEGMENT4	SEGMENT5	SEGMENT6	CURRENCY_CODE	PERIOD_NAME	NET_AMOUNT
1	Vision Services (USA)	01	740	7844	000	(null)	(null)	USD	Jul-20	0
2	Vision Services (USA)	01	840	1660	000	(null)	(null)	USD	Jul-20	0
3	Vision Services (USA)	01	420	7040	000	(null)	(null)	USD	Jul-20	0
4	Vision Services (USA)	01	420	7420	000	(null)	(null)	USD	Jul-20	0
5	Vision Services (USA)	01	402	4130	000	(null)	(null)	USD	Jul-20	0
6	Vision Services (USA)	01	560	7450	000	(null)	(null)	USD	Jul-20	0
7	Vision Services (USA)	01	570	7450	000	(null)	(null)	USD	Jul-20	0
8	Vision Services (USA)	01	830	7580	000	(null)	(null)	USD	Jul-20	0
9	Vision Services (USA)	01	000	1640	000	(null)	(null)	USD	Jul-20	0
10	Vision Services (USA)	01	000	1660	000	(null)	(null)	USD	Jul-20	0
11	Vision Services (USA)	01	450	7530	000	(null)	(null)	USD	Jul-20	0
12	Vision Services (USA)	01	470	7530	000	(null)	(null)	USD	Jul-20	0
13	Vision Services (USA)	01	480	7530	000	(null)	(null)	USD	Jul-20	0
14	Vision Services (USA)	01	000	1560	000	(null)	(null)	USD	Jul-20	0
15	Vision Services (USA)	01	420	5050	000	(null)	(null)	USD	Jul-20	0
16	Vision Services (USA)	01	430	7350	000	(null)	(null)	USD	Jul-20	0
17	Vision Services (USA)	01	420	7690	000	(null)	(null)	USD	Jul-20	0
18	Vision Services (USA)	01	000	2550	000	(null)	(null)	USD	Jul-20	0
19	Vision Services (USA)	01	420	7560	000	(null)	(null)	USD	Jul-20	0
20	Vision Services (USA)	01	422	7360	000	(null)	(null)	USD	Jul-20	0

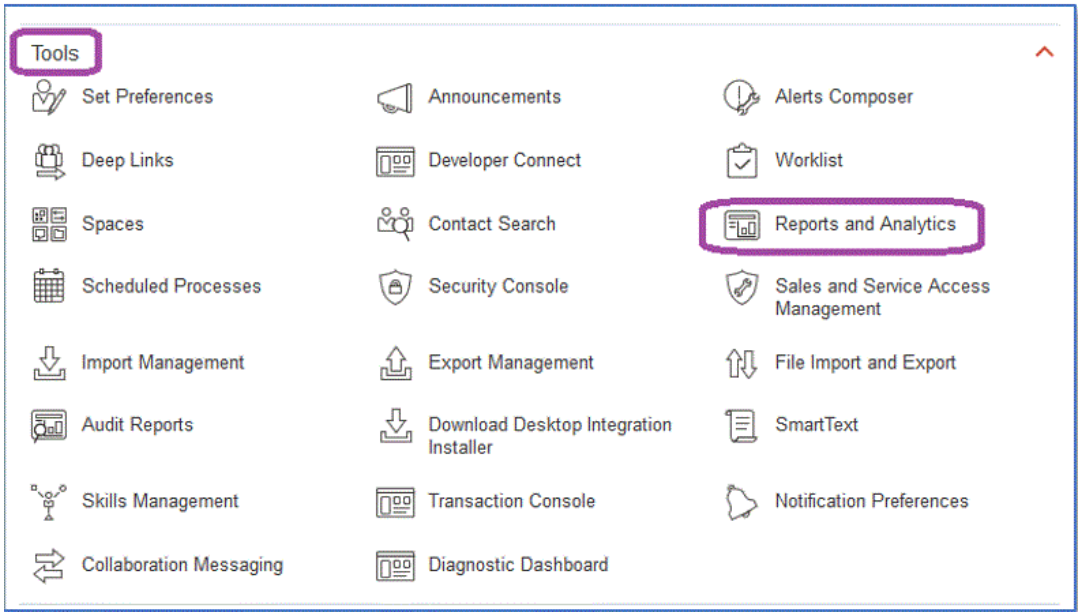
Creating the Data Model for the BI Publisher Report Extract

In this step you create the data model for the Oracle Business Intelligence Publisher report extract.

A data model is an object that contains a set of instructions for BI Publisher to retrieve and structure data for a report. Data models reside as separate objects in the catalog. At the very simplest, a data model can be one data set retrieved from a single data source (for example, the data returned from the columns in the employees table). A data model can also be complex, including parameters, triggers, and bursting definitions as well as multiple data sets.

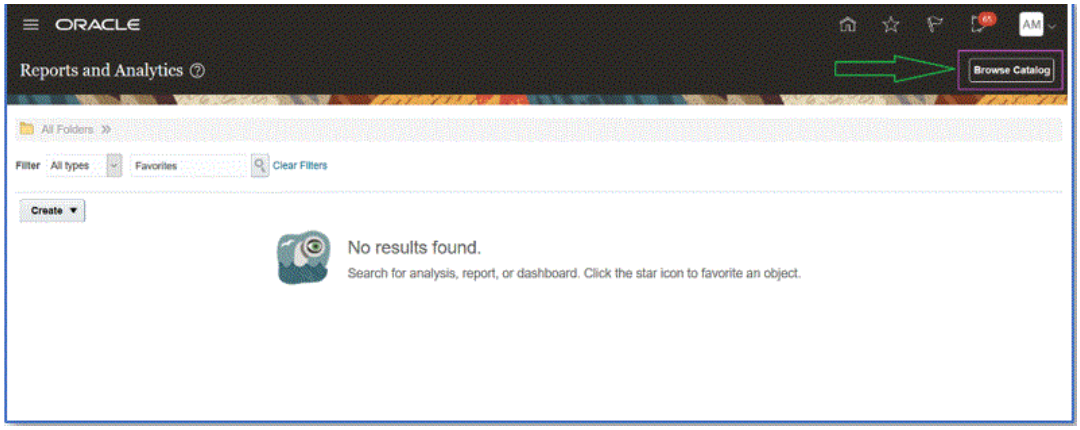
To build a data model:

1. In Oracle ERP Cloud under **Tool**, navigate to **Report and Analytics**.



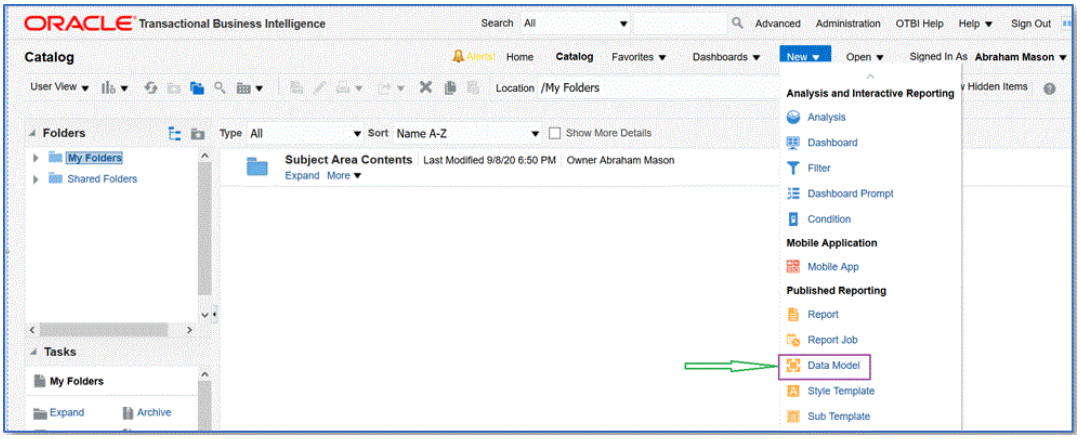
2. When BI Publisher opens, click **Browse Catalog**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



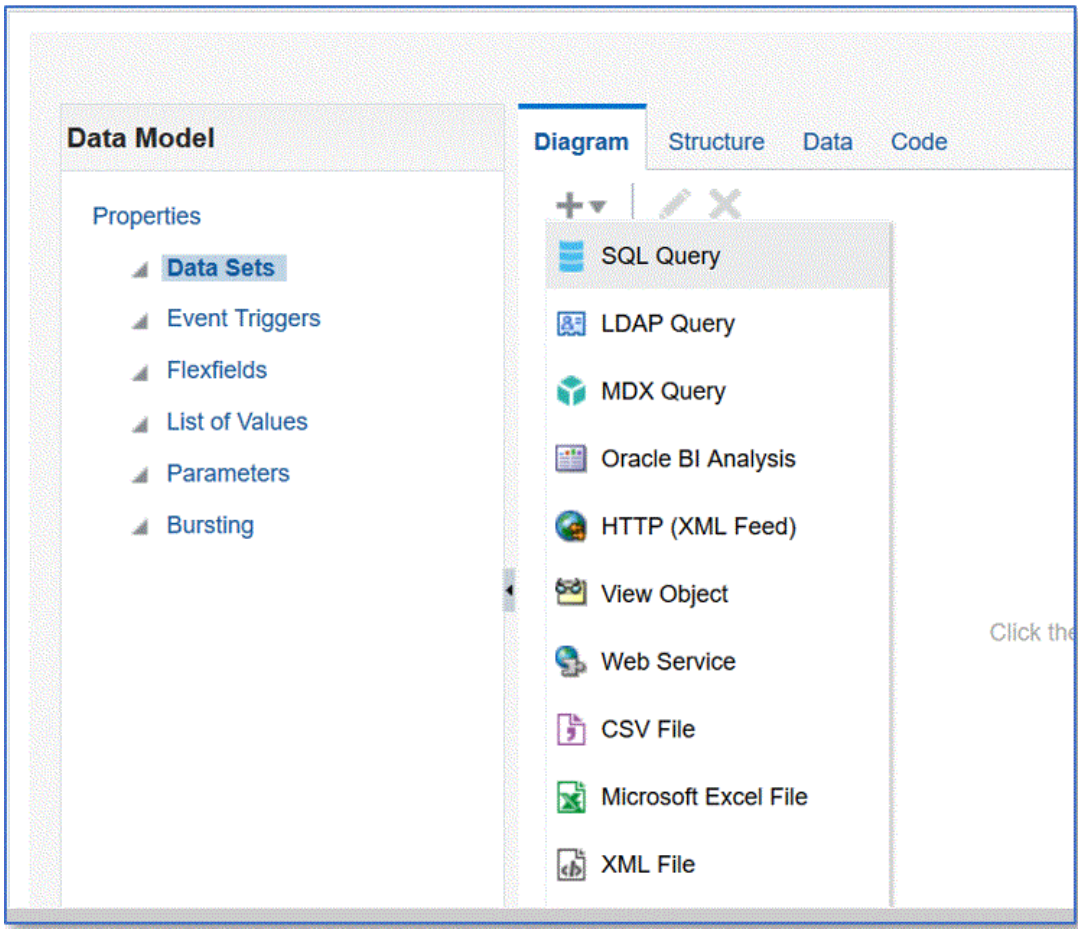
A new tab opens where you can create a new data model. This is required for all BI Publisher extracts, and is the source for the BI Publisher report.

3. Click **Data Model**.



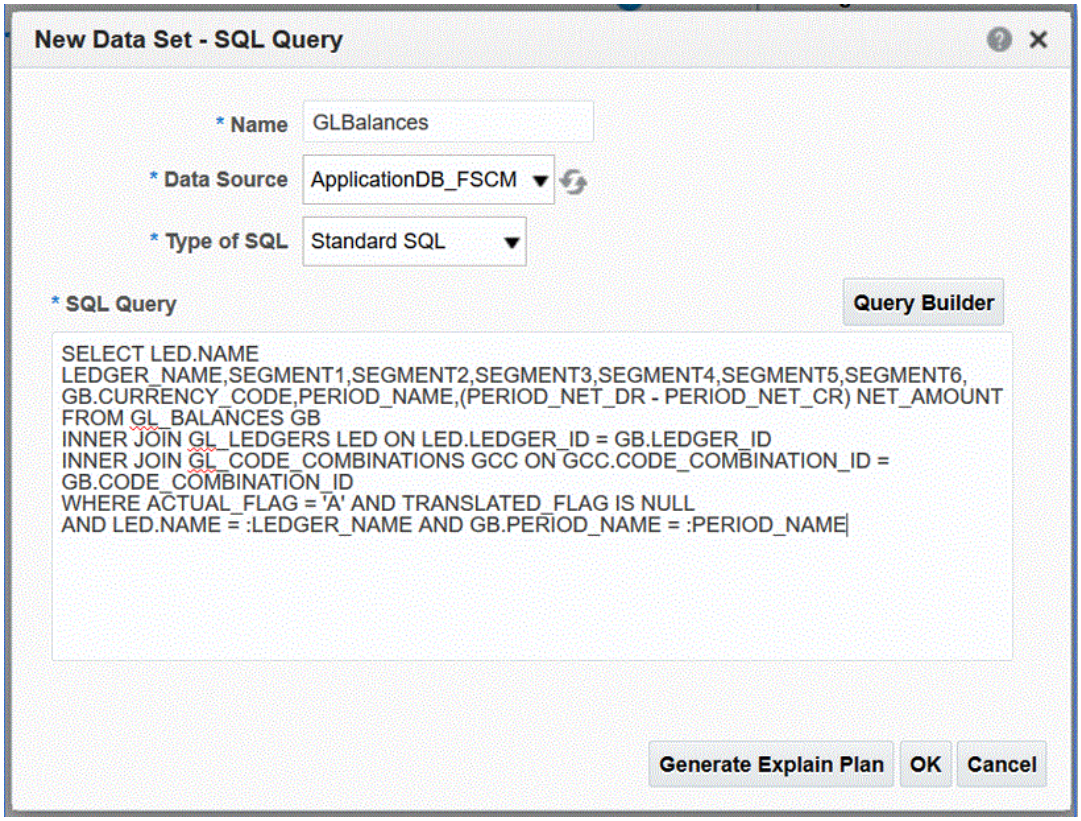
4. From the **Data Model** page, select **SQL Query**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



5. Specify the name of the query and then in **Type of SQL** drop-down, select **Standard SQL**.
6. In **SQL Query**, paste the SQL defined in the prior section and then click **OK**.

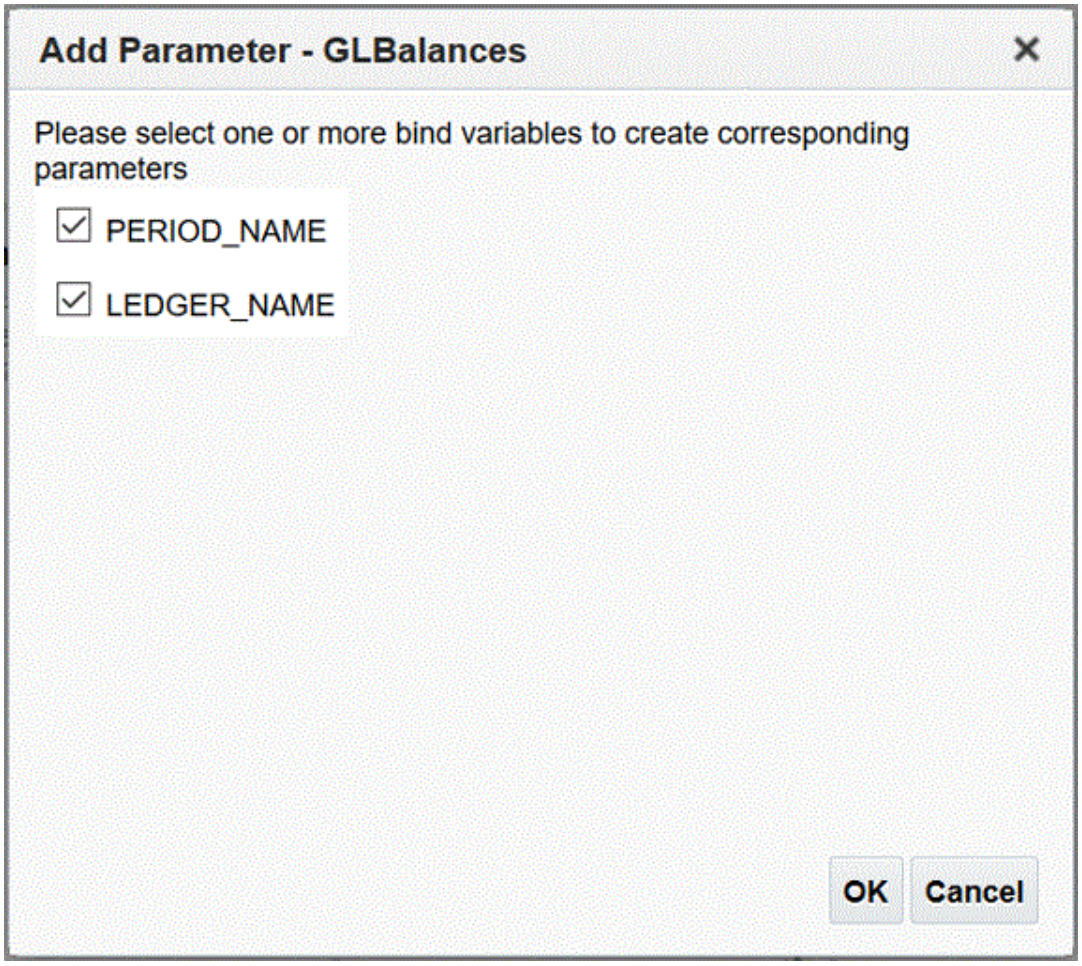
Copyright © 2017, 2026, Oracle and/or its affiliates.



7. From the **Add Parameter** page, check the bind parameters to include and then click **OK**.

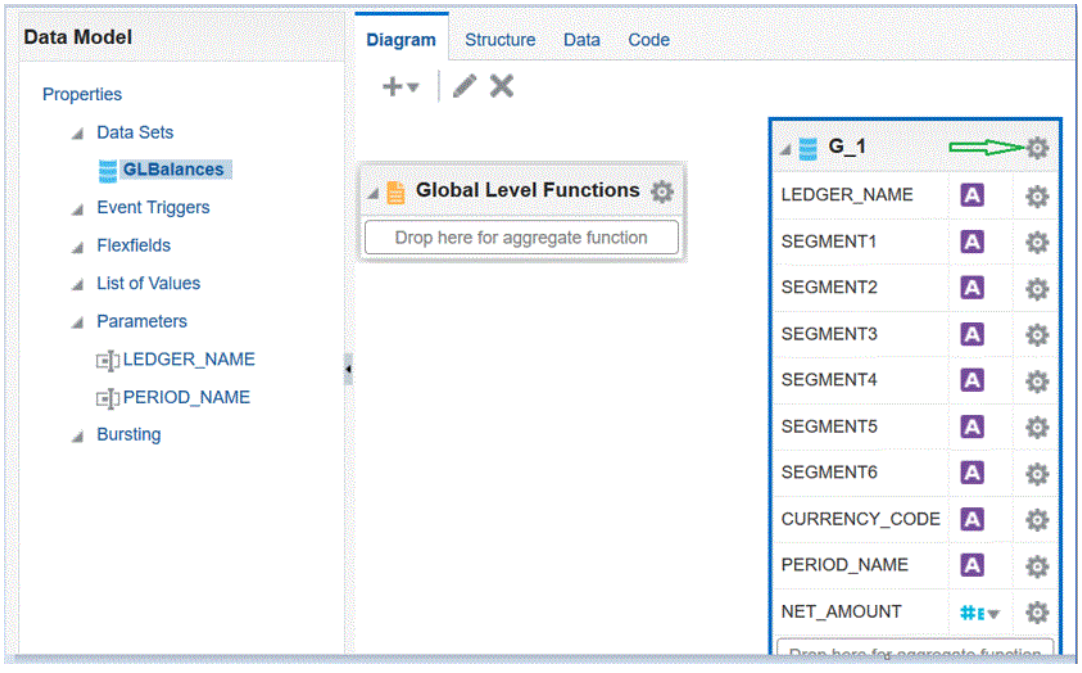
   Bind parameters are detected automatically.

Copyright © 2017, 2026, Oracle and/or its affiliates.



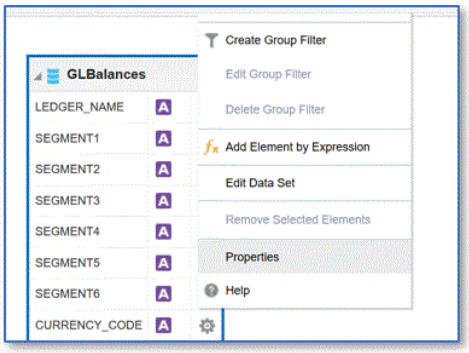
8. From the **Data Model** page, then **Properties**, then **Data Sets**, select the query.

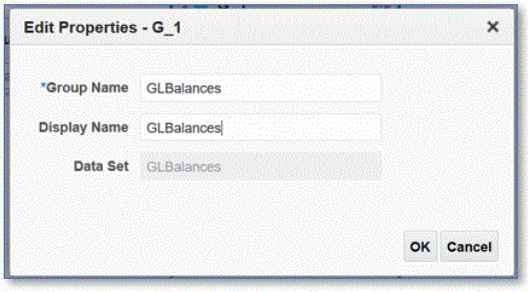
   The query columns are displayed.



9. Select **Properties** to edit the group name as needed and then click **OK**.

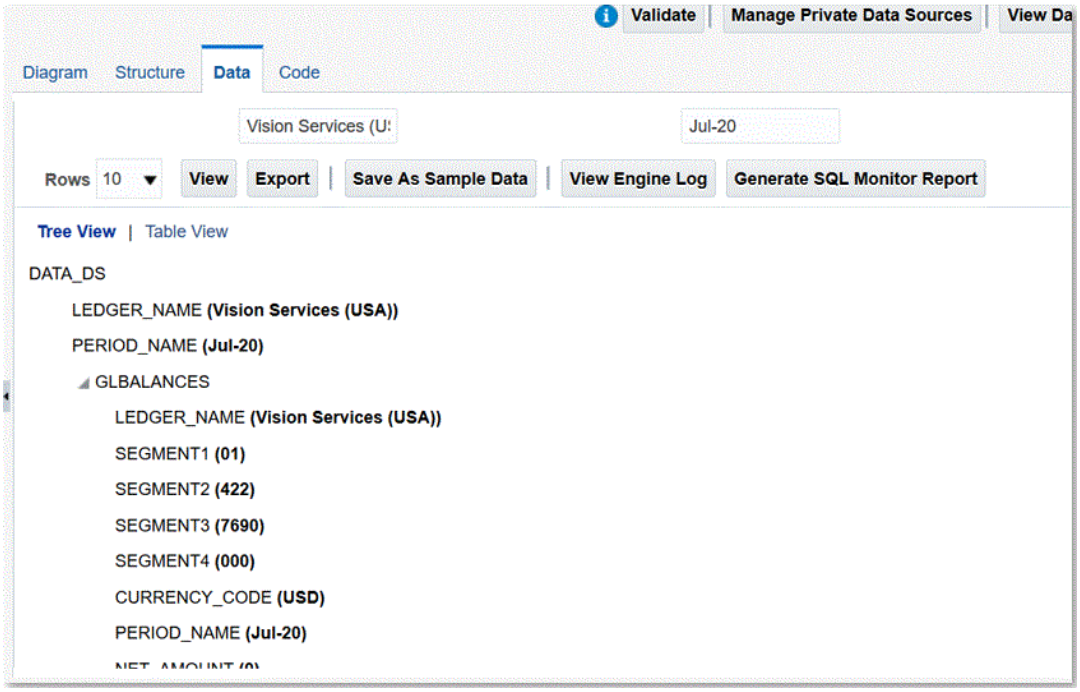
Copyright © 2017, 2026, Oracle and/or its affiliates.

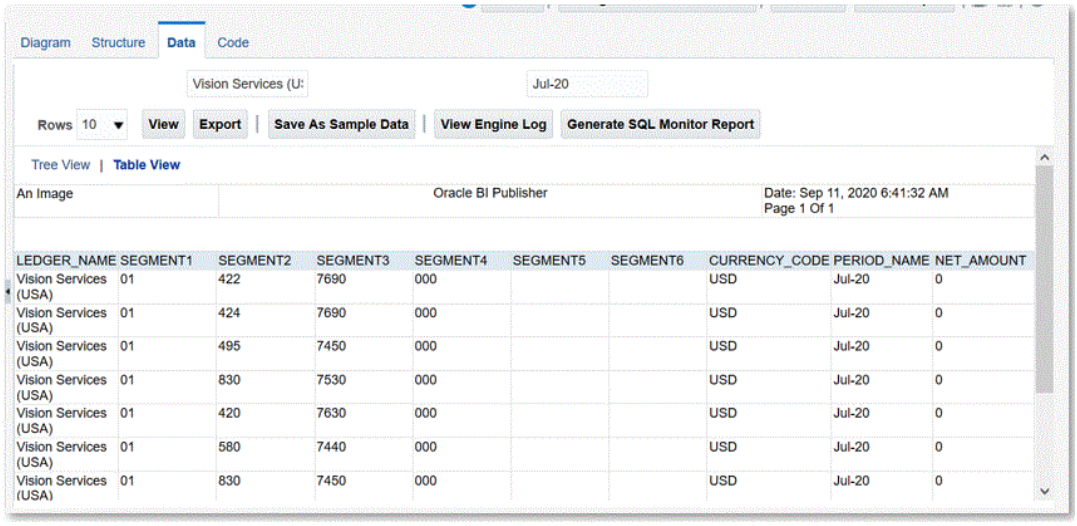




10. From the **Data Model** page, click the **Data** tab to view sample data for the query by entering sample values for the bind parameters.

    Data may be viewed in both tree and table formats.





LEDGER_NAME	SEGMENT1	SEGMENT2	SEGMENT3	SEGMENT4	SEGMENT5	SEGMENT6	CURRENCY_CODE	PERIOD_NAME	NET_AMOUNT
Vision Services (USA)	01	422	7690	000			USD	Jul-20	0
Vision Services (USA)	01	424	7690	000			USD	Jul-20	0
Vision Services (USA)	01	495	7450	000			USD	Jul-20	0
Vision Services (USA)	01	830	7530	000			USD	Jul-20	0
Vision Services (USA)	01	420	7630	000			USD	Jul-20	0
Vision Services (USA)	01	580	7440	000			USD	Jul-20	0
Vision Services (USA)	01	830	7450	000			USD	Jul-20	0

11. Click **Save as Sample Data**.

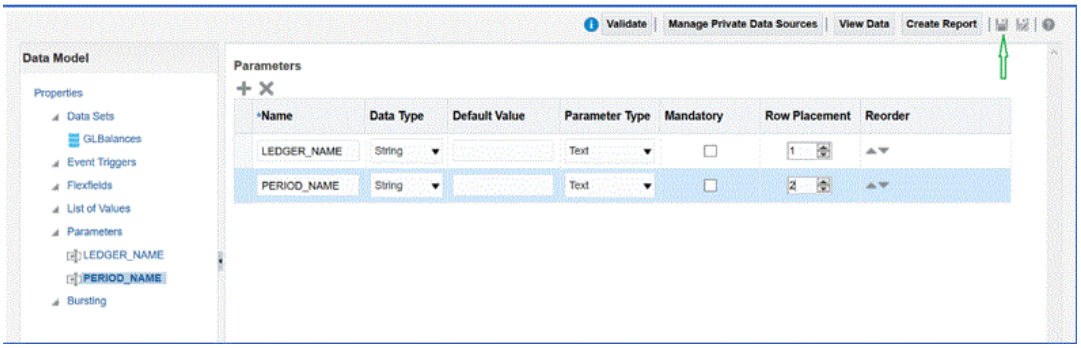
Copyright © 2017, 2026, Oracle and/or its affiliates.

Sample data is required to build the report layout and to preview the report definition.

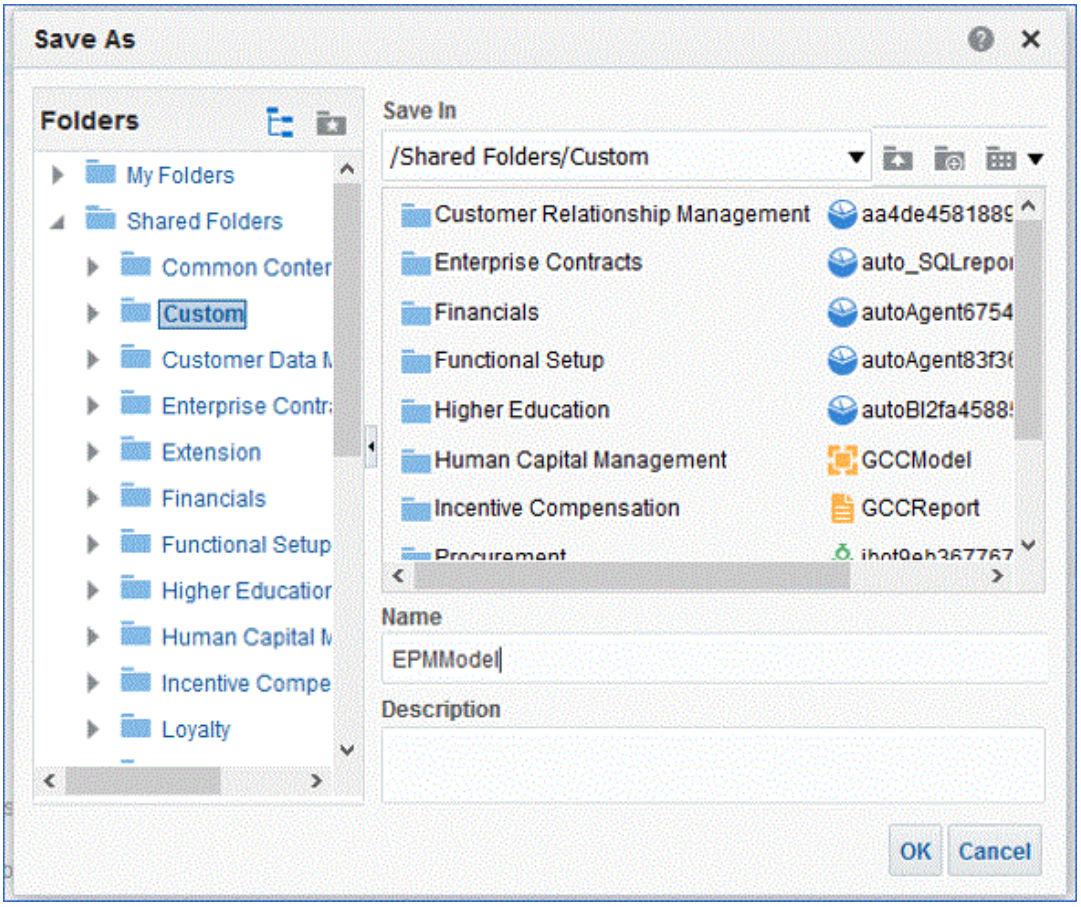
12. From the **Data Model** menu, navigate to **Parameters**, and then select **Row Placement Order**.

    The order is important, and the same order should be defined in the report definition.

13. Confirm the row and click **Save**.

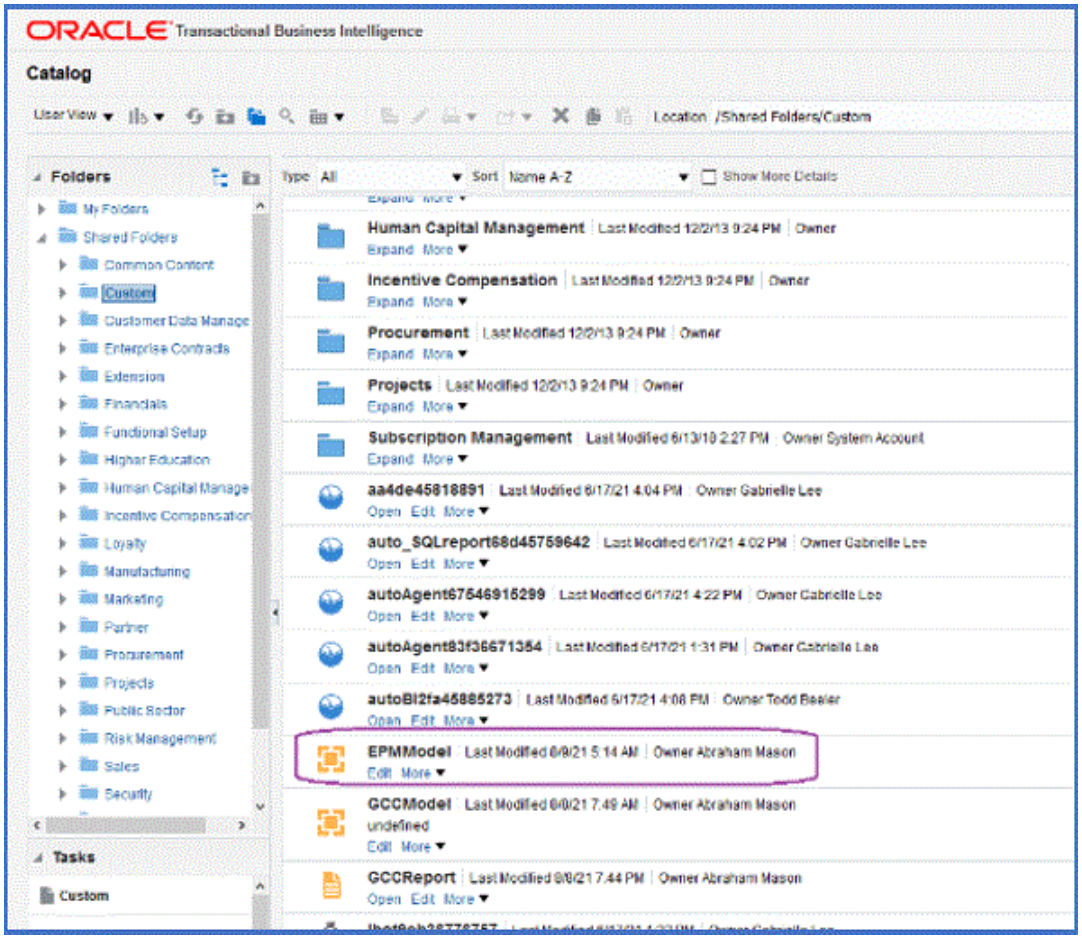


14. You are prompted to select the location to which to save the data model.



15. View the data model and make any edits in the BI Catalog once it has been saved.

Copyright © 2017, 2026, Oracle and/or its affiliates.



After the data model has been saved, it can be used as the basis for the BI Publisher extract.

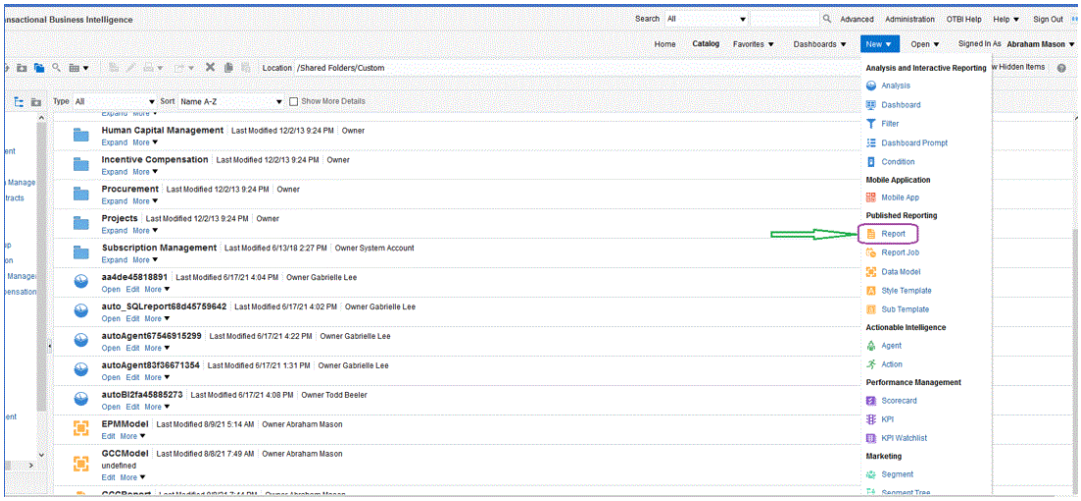
Creating the BI Publisher Report Extract

In this step you create the Oracle Business Intelligence Publisher report extract used as the data source for the integration with the Oracle Fusion Cloud Enterprise Performance Management.

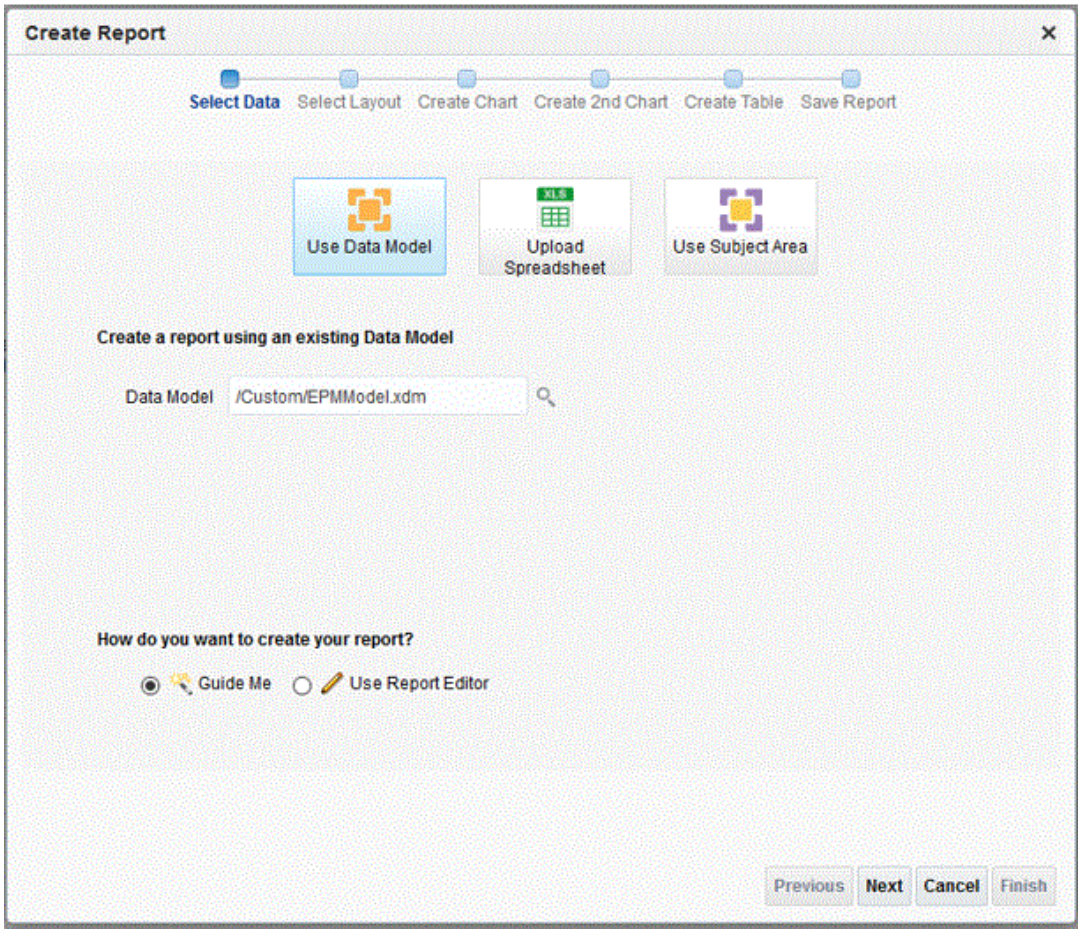
To create the BI Publisher report extract:

1. In BI Publisher, select **New**, then under **Published Reporting**, select **Report** menu option to create a new report.

Copyright © 2017, 2026, Oracle and/or its affiliates.

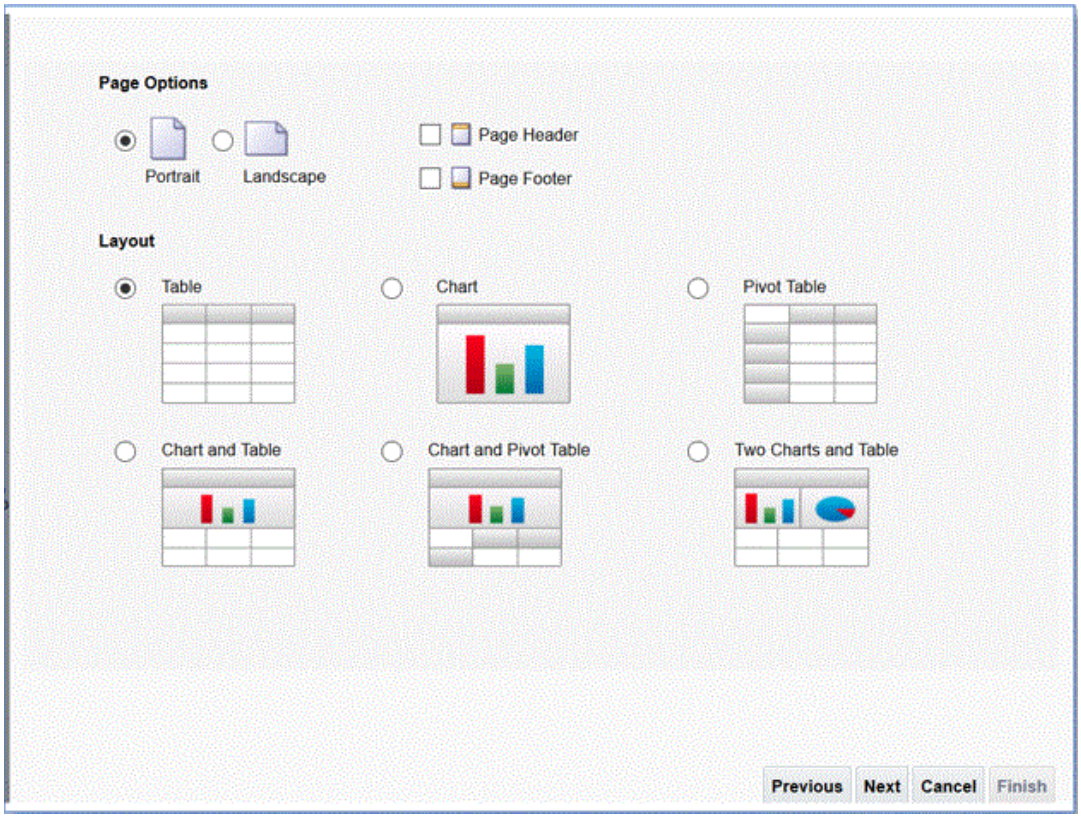


2. On the **Create Report** page, click **Use Data Model** to use an existing data model and then from **Create a report using an existing Data Model**, select the data model and then click **Next**.



3. On the following page, leave the default values.

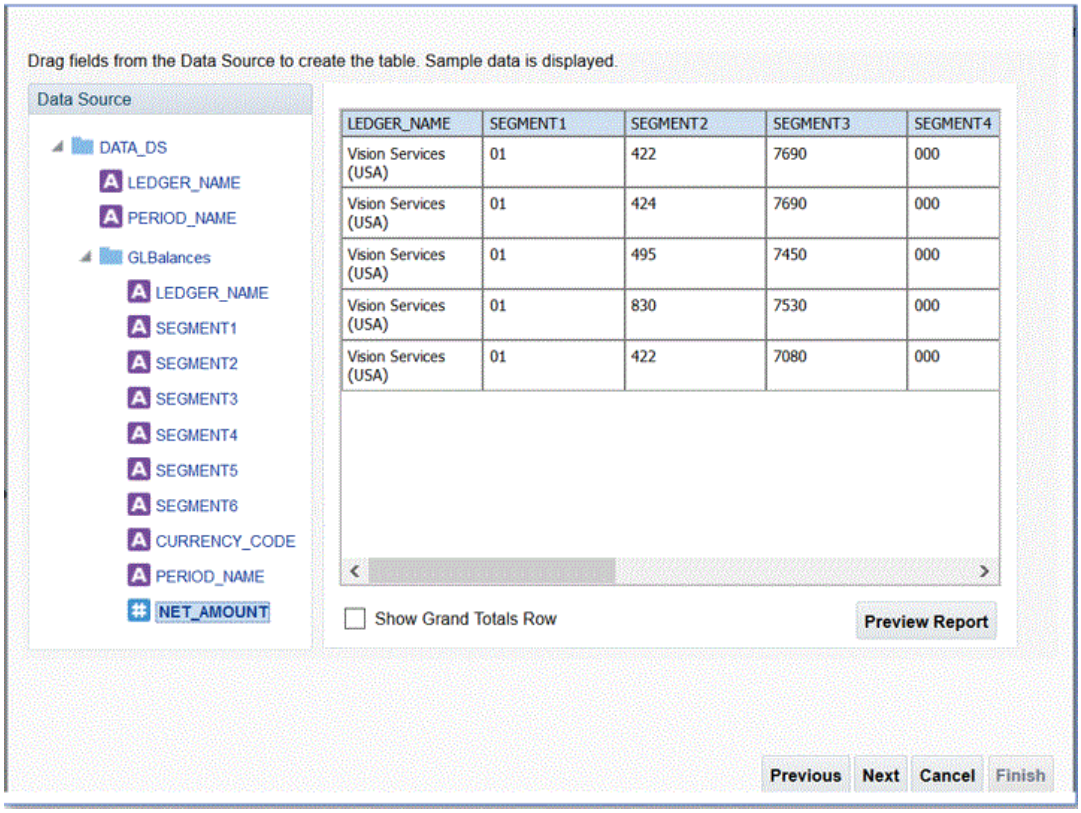
Copyright © 2017, 2026, Oracle and/or its affiliates.



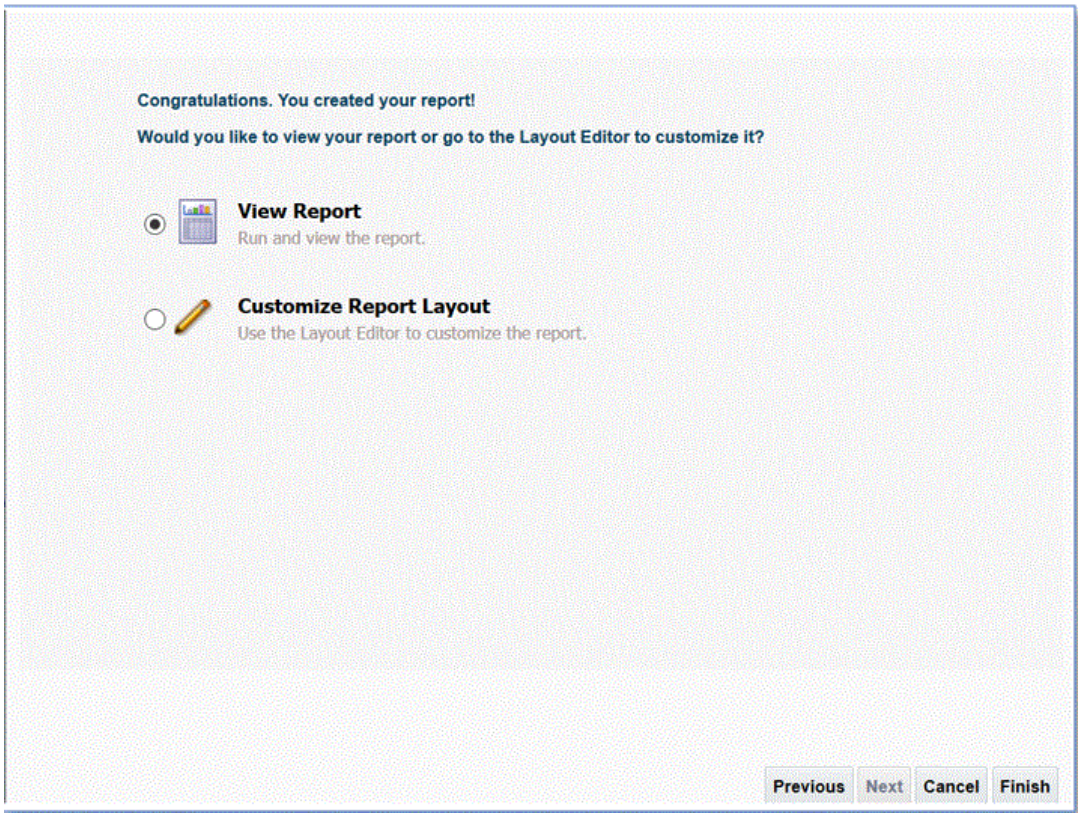
4. On the next page, uncheck the **Show Grand Totals Row** field at the bottom of the page.
5. On the same page, drag and drop the columns from the **Data Source** in the left pane to the layout area in the right page including all columns for the extract.

   The displayed sample data is based on the sample data generated when you created the data model.

Copyright © 2017, 2026, Oracle and/or its affiliates.



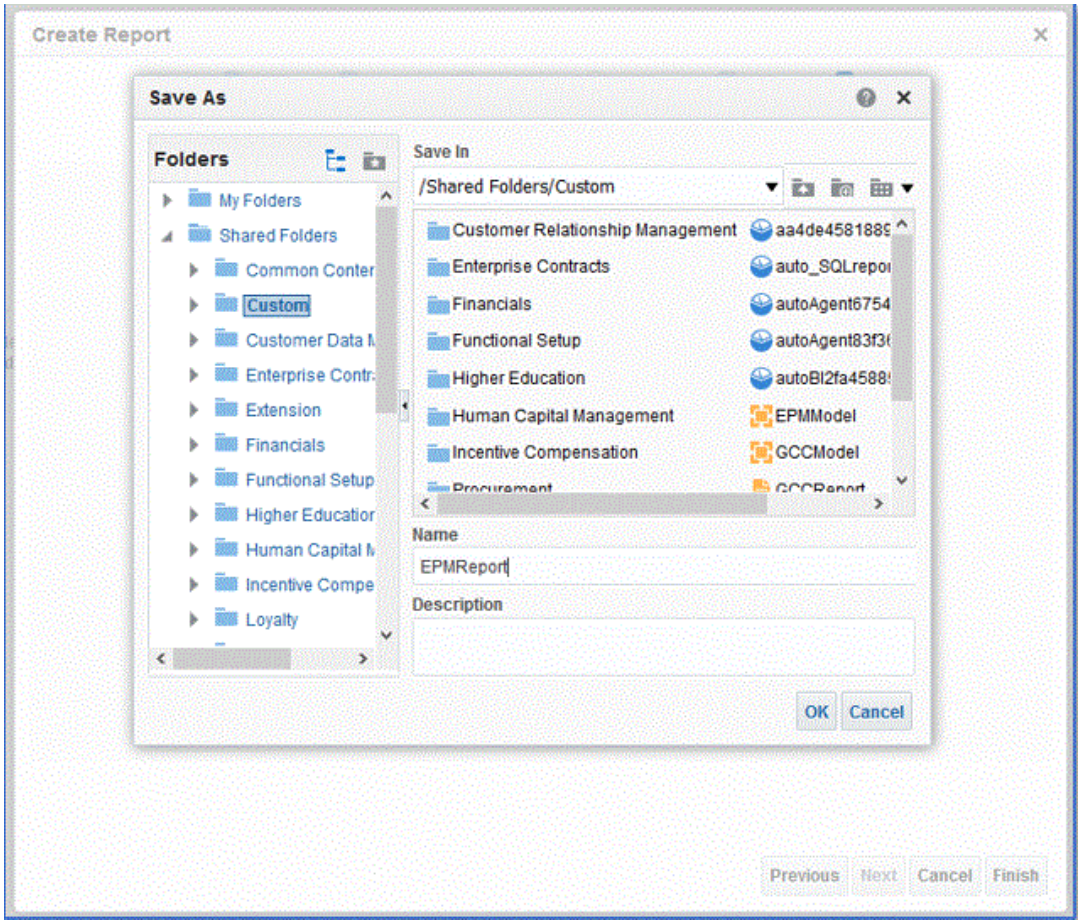
6. Click **Finish**.



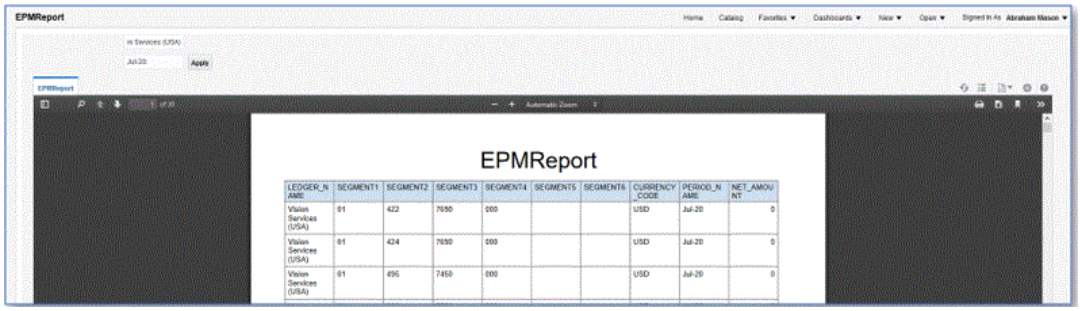
7. Save the report to the desired location.

   In this example, the report definition is saved to */Custom/EPMReport.xdo*.

Copyright © 2017, 2026, Oracle and/or its affiliates.

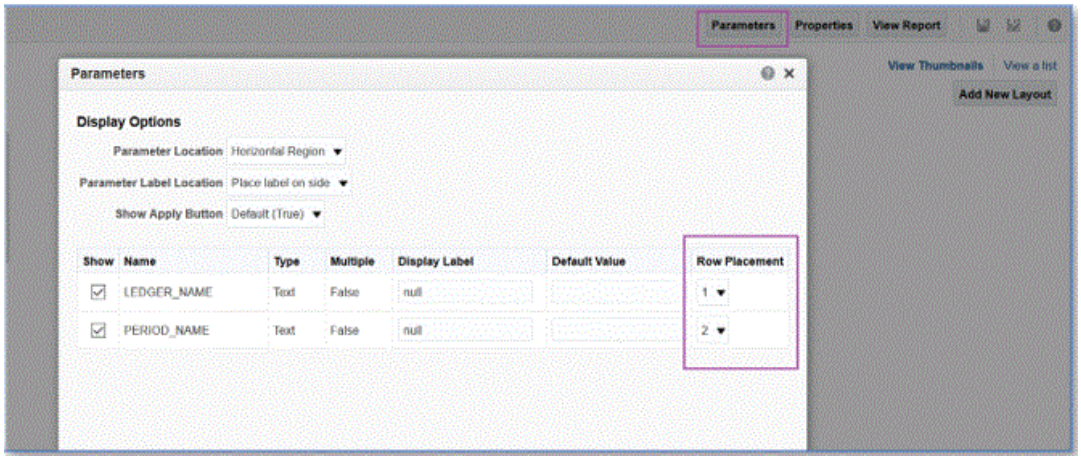


8. After the report is saved, run the report to confirm that the data is displayed correctly. Then provide sample values for the bind parameters if needed.



9. Confirm the row placement order, edit the report, and confirm that the order is the same as the data model.

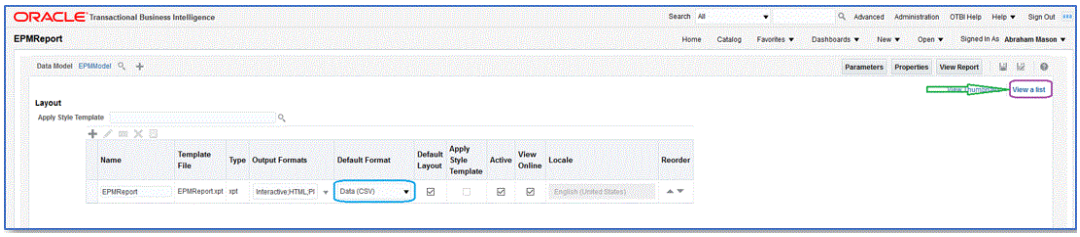
Copyright © 2017, 2026, Oracle and/or its affiliates.



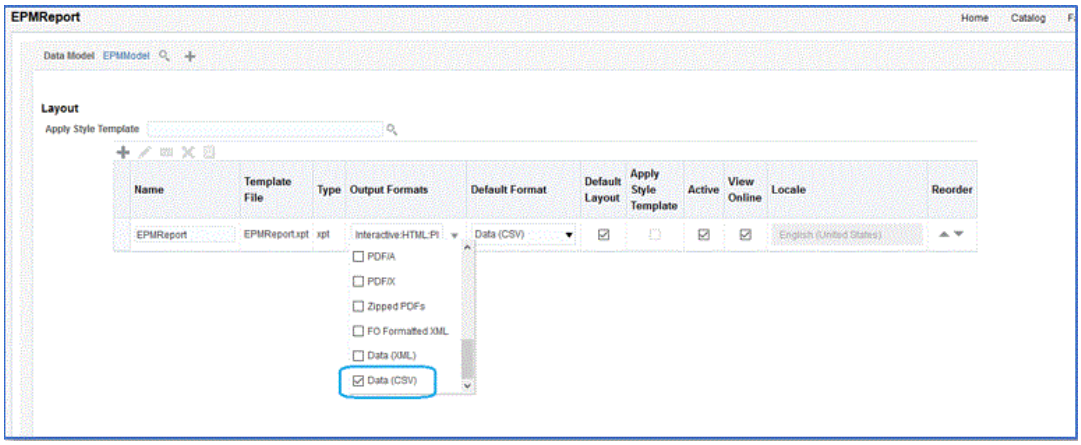
10. While editing the report, ensure that CSV is included as the default output format for the report.

> **Note**
>
> When running the extract in synchronous mode, you do not need to set the default output format to CSV because the execution is called with CSV as a parameter. However, when running in asynchronous mode, the custom ESS job needs to have the default format set to CSV.

11. To select CSV as the default output format, click **View as List**.



12. From **Output Formats**, check **Data (CSV)**.

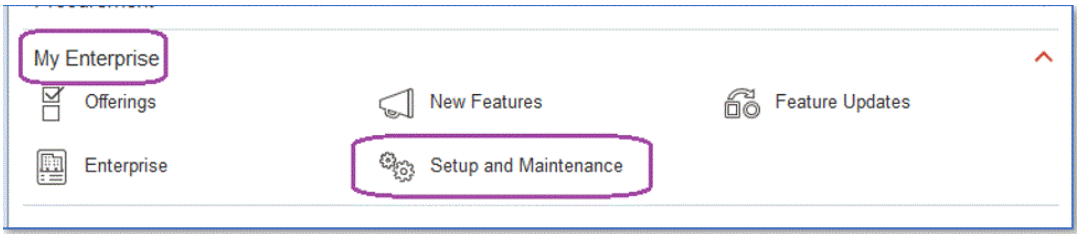


Copyright © 2017, 2026, Oracle and/or its affiliates.

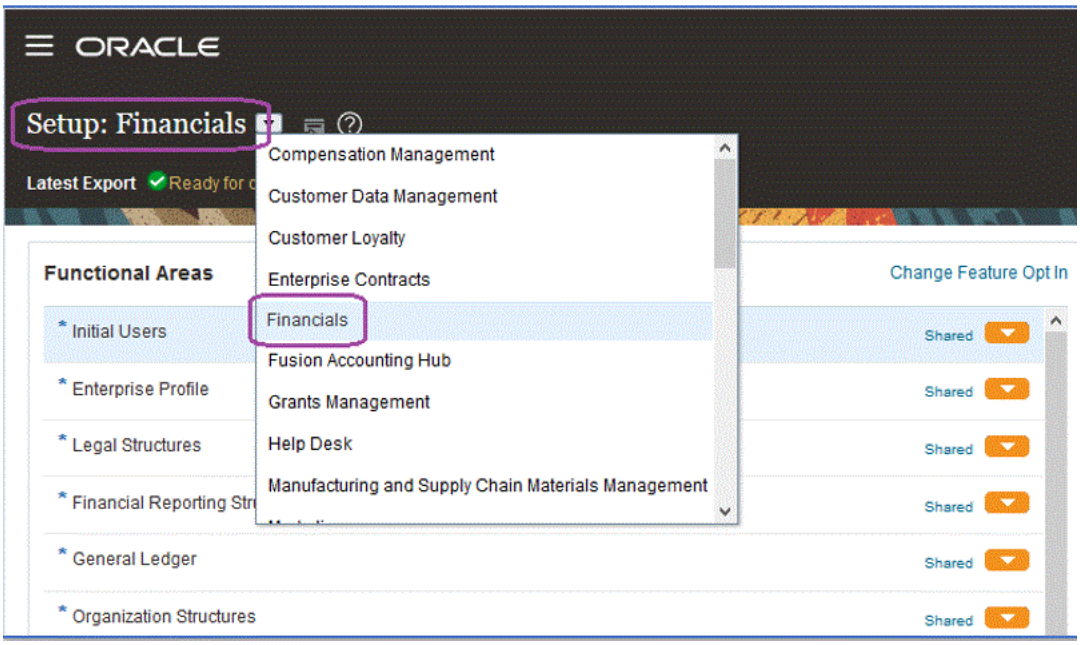
Creating an Oracle Enterprise Scheduler (ESS) Job

You register the Oracle Business Intelligence Publisher report as an Oracle Enterprise Scheduler (ESS) Job to run an integration in asynchronous mode. This mode enables you to schedule a job to be executed at specific intervals not subject to timeout restrictions.
To create an ESS job and run an extract in asynchronous mode:

1. In BI Publisher, navigate to **My Enterprise**, and then select **Setup and Maintenance**.

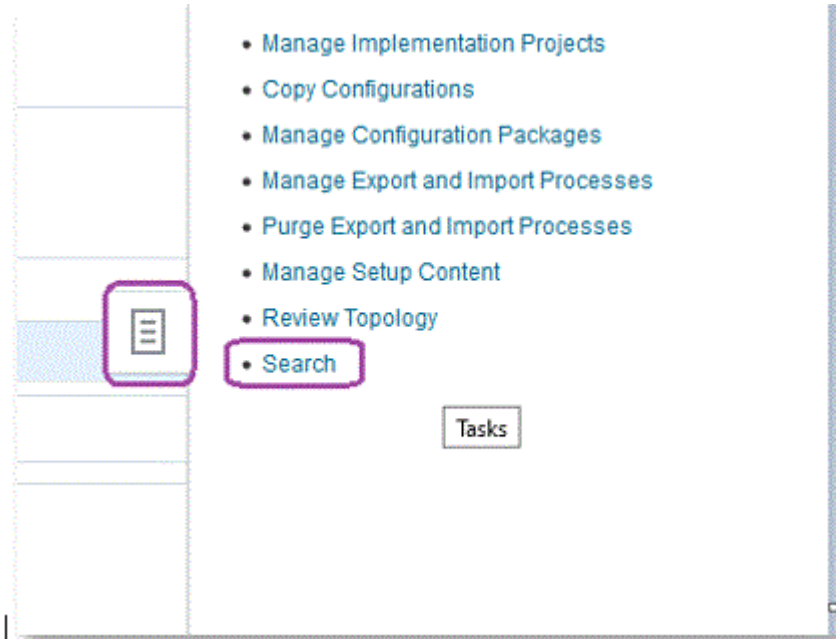


2. From the **Subject Area** drop-down, select **Financials**.

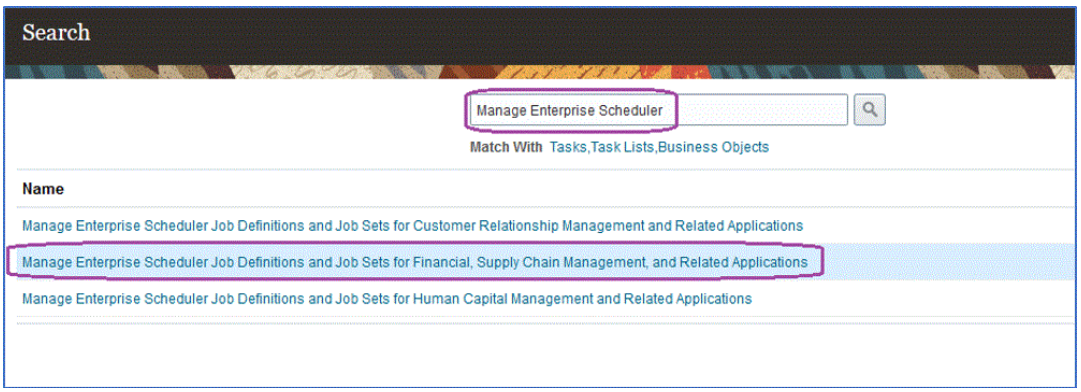


3. Click **Search**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



4. Search for **Manage Enterprise Scheduler** and then select **Manage Enterprise Scheduler Job Definitions and Job Sets for Financial, Supply Chain Management, and Related Applications**.



5. On the **Manage Job Definitions** page, click to add or create a new ESS job.

6. On the **Manage Enterprise Scheduler Job Definitions and Job Sets for Financial, Supply Chain Management, and Related Applications** page, complete the following fields:

   a. **Display Name**—Enter a user-defined name for the extract. This name is displayed under Scheduled Processes.

   b. **Name**—Specify an alphanumeric name without spaces and special characters.

      This name is used for the ESS Job Name in Data Integration application options.

   c. **Path**—Specify the path where to save the ESS job.

      This path is created as a subfolder of `/oracle/apps/ess/custom`. For example, if you specify **epm**, then the ESS Job path is `/oracle/apps/ess/custom/epm`.

      The path entered here populates the **ESS Job Path** field in Data Integration application options.

   d. **Job Application Name**—Select **FscmEss**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

e. **Job Type**—Select **BIPJobType**.

   This is a mandatory parameter because Data Integration can only trigger ESS jobs of the type – BIPJobType.

f. **Default Output Format**—Select **XML**.

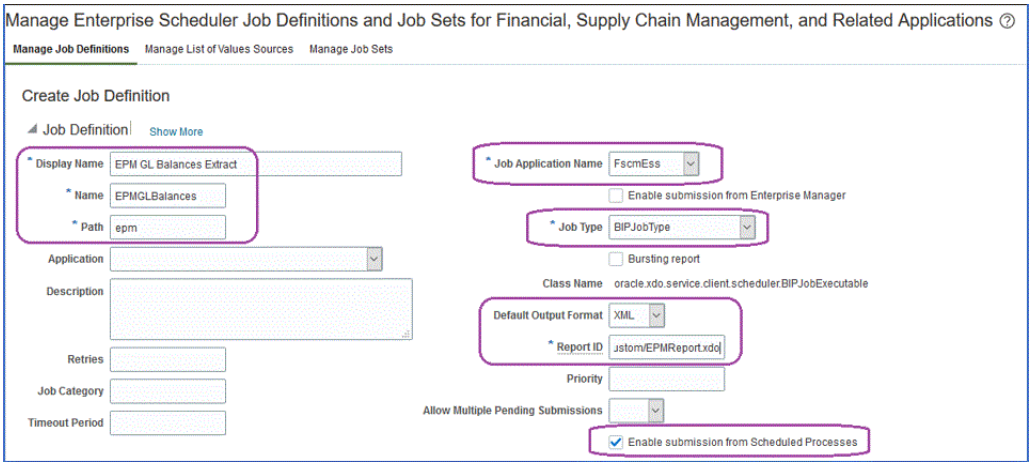
   The XML output format parameter is output in addition to the CSV format but does not change the CSV format and is not streamed to the Oracle Fusion Cloud Enterprise Performance Management.

g. **Report ID**—Specify the report path of the custom report defined in the Creating the BI Publisher Report Extract step.

   For example, specify `/Custom/MyReport.xdo`.

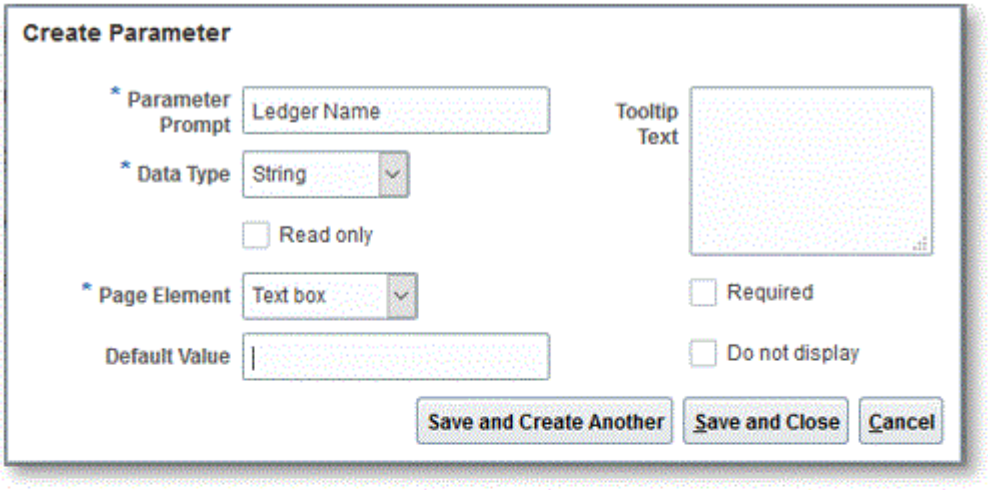
h. **Enable submission from Scheduled Processes**—Check to enable.

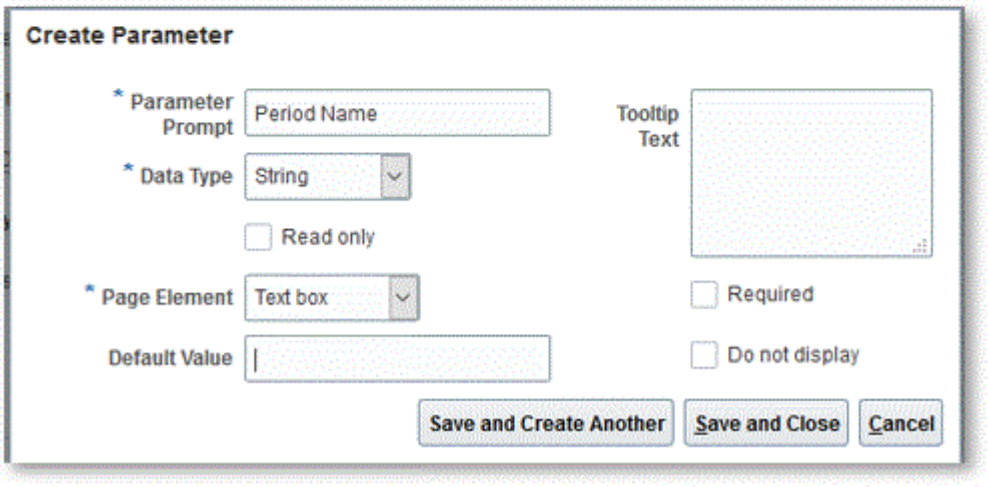
   This option enables you to invoke the job manually and to troubleshoot it in case of a data mismatch.



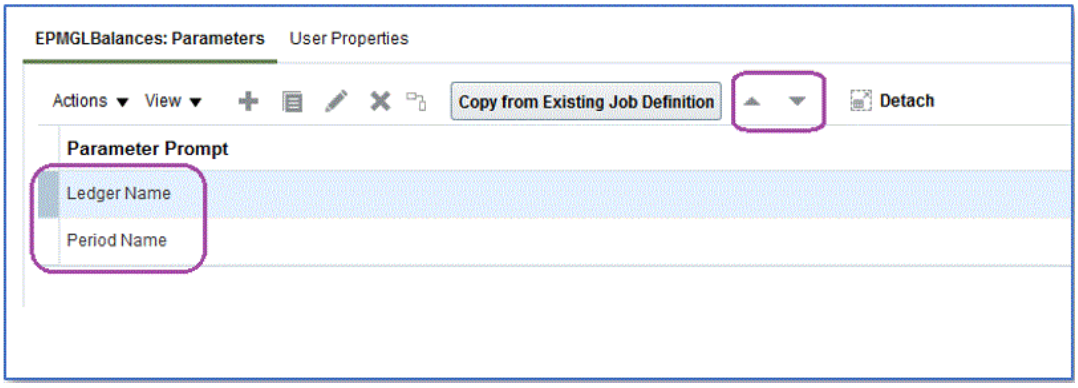
7. On the **Data Model** components pane, then **Parameters** , click **Create Parameter** page.
8. Create parameters for each of the custom report parameters defined in BI Publisher in the same order that they were created for the report.

Copyright © 2017, 2026, Oracle and/or its affiliates.



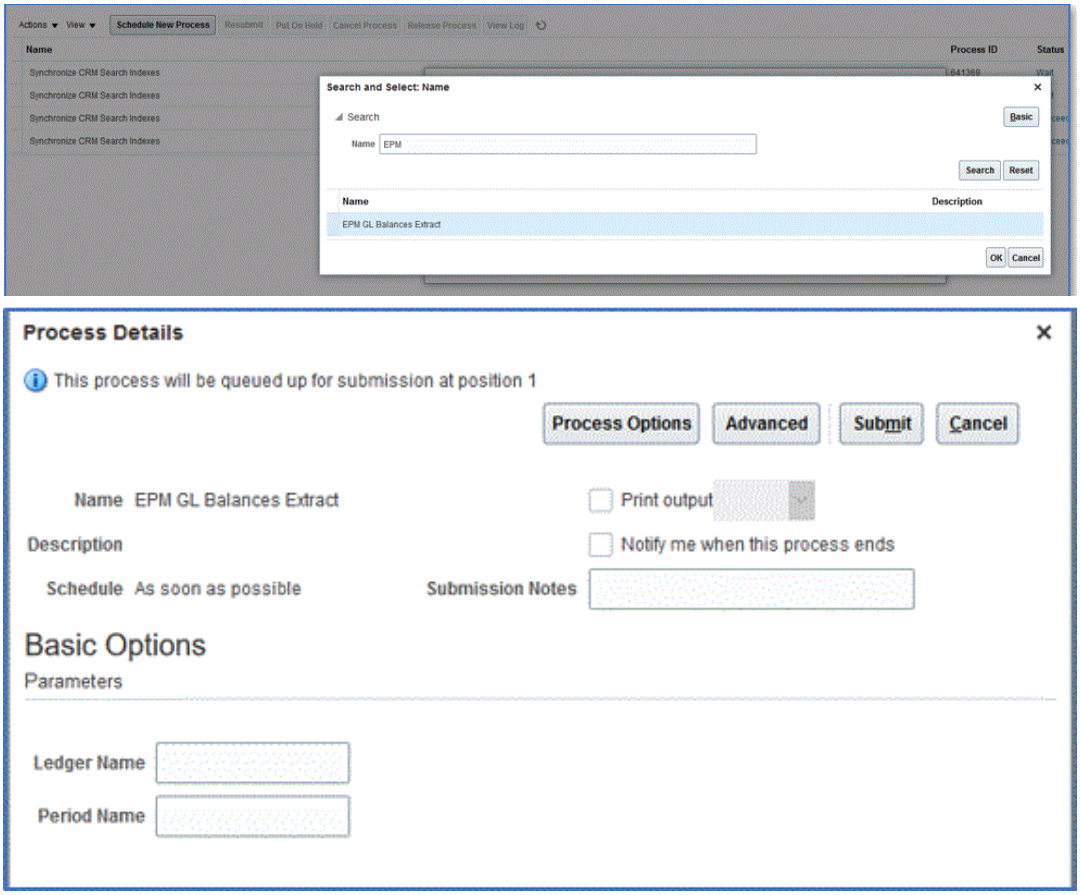


9. On the **Parameters** page, reorder parameters using the up/down arrow if necessary.



10. In the Oracle ERP Cloud, view or submit the BI Publisher job by selecting **Schedule New Process**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

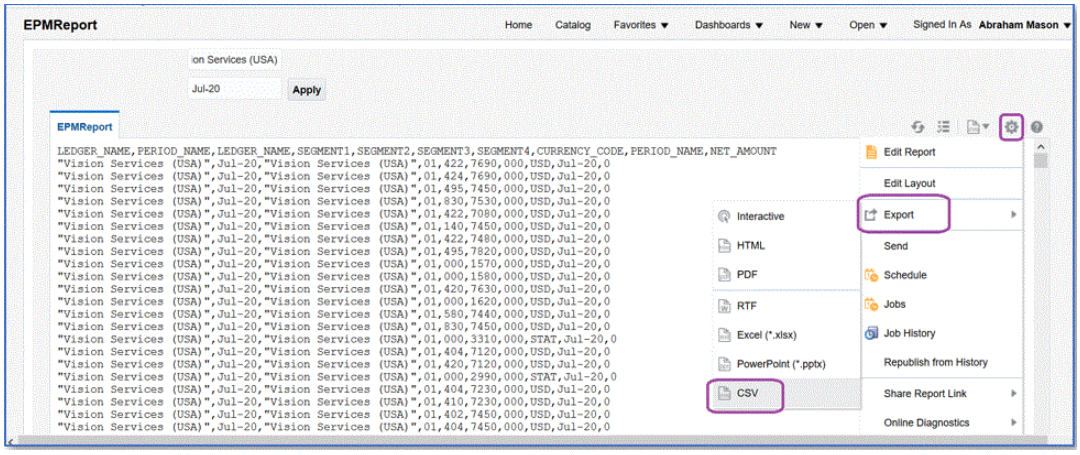


Creating the Integration Definition Between Cloud EPM and the BI Publisher Report Extract

After you built the Oracle Business Intelligence Publisher report and registered it as an ESS job, you define the integration definition between Oracle Fusion Cloud Enterprise Performance Management and the BI Publisher report extract.

To create the integration definition:

1. In BI Publisher, run the BI Publisher report extract, and export it in CSV file format locally to your file system if registering for the first time.



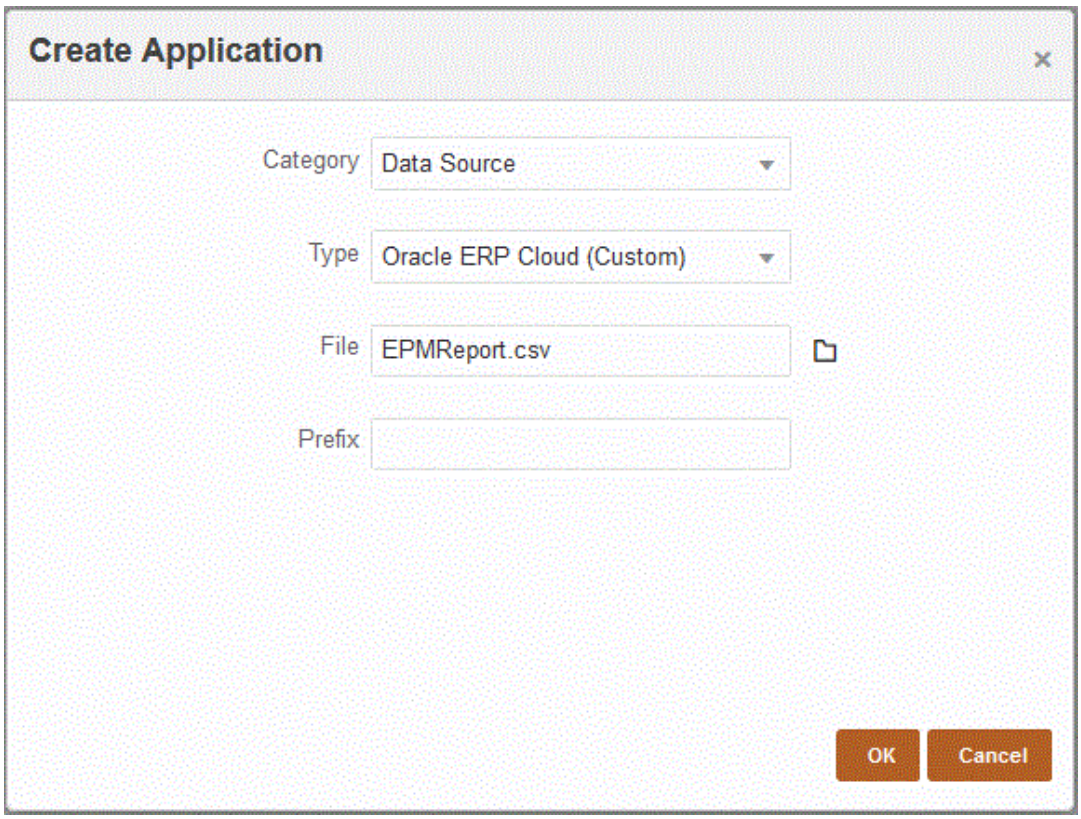
Copyright © 2017, 2026, Oracle and/or its affiliates.

You may want to rename the downloaded CSV output file as *<Appname>.csv* where *<Appname>* is the name of the intended application for the data source application in Data Integration, which represents the BI Publisher report.

The the CSV file needs to be copied (uploaded) to the Data Integration inbox folder. For information about uploading files, see Using the File Browser.

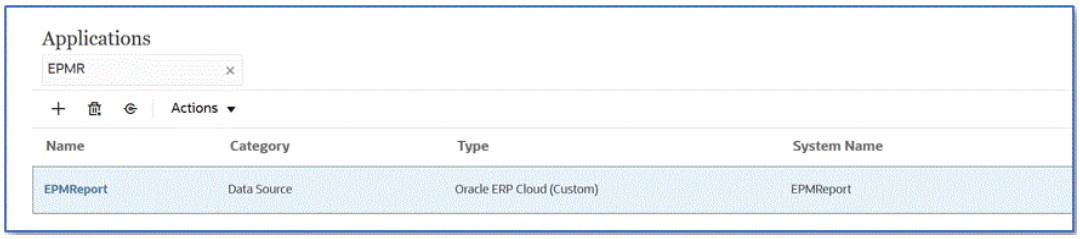
2. From the **Data Integration** home page, and then **Actions**, select **Applications**.
3. On the **Applications** page, click (Add icon).
4. From **Category**, select **Data Source**.
5. From **Type**, select **Oracle ERP Cloud (Custom)**.
6. From **File**, specify the name of BI Publisher report extract CSV file or click and navigate to the folder where you saved the BI Publisher report extract CSV file, select it and then click **OK**.
7. In **Prefix**, specify a prefix to make the application name unique.

   The prefix is concatenated with the application name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.
8. Click **OK**.
9. Click **Save**.



When the application is registered, the system returns the message: "Application registered successfully" and the application is available on the Application page as shown below:

Copyright © 2017, 2026, Oracle and/or its affiliates.



Data Integration registers the application and returns all the columns in Dimension Details.

10. On the **Application** page, click ... next to data source application and then select **Application Details**.
11. On the **Application Details** page, click the **Options** tab.
12. Complete the following fields and click **Save**.
    a. **Connection Name**—Specify the source system name.
    b. **Execution Method**—Specify the method for executing the job.

       Valid options:

       - **BIP Report**—Executes the BI Publisher report in synchronous mode, which has a timeout limit of about 5 minutes within the Oracle ERP Cloud instance. This method is suitable for smaller data sets or for queries that execute quickly.
       - **ESS Job**—Executes the BI Publisher report in asynchronous mode, without encountering execution time restrictions.

       > **Note**
       >
       > If you encounter a timeout in Data Integration when using the ESS feature for a BI Publisher report, increase the batch timeout in Data Management. To do this, specify the maximum time a job can run in the **Timeout** field of the batch definition. For more information, see Working with Batch Definitions in *Administering Data Management for Oracle Enterprise Performance Management Cloud*.

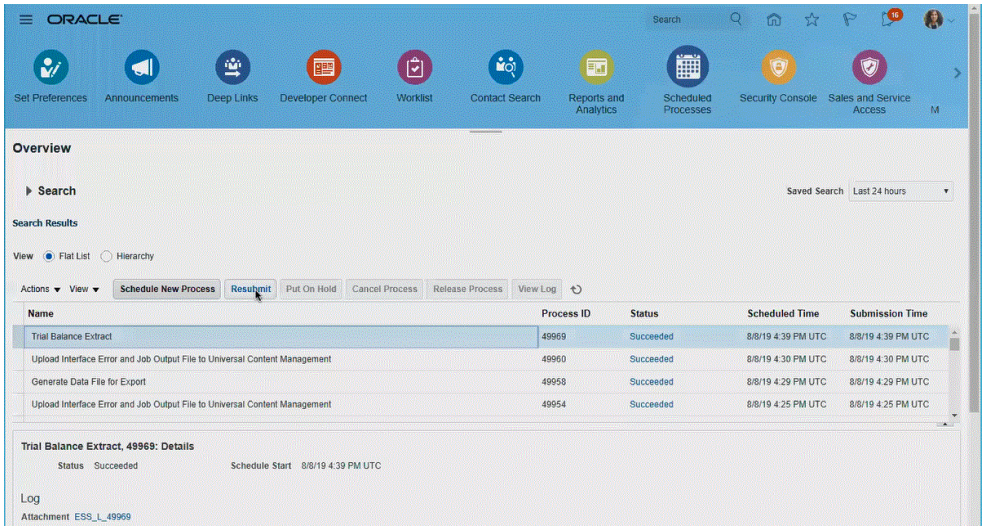
    c. **Report Name**—Enter the name of the report and the complete report path when the execution method is **BIP Report**. For example, enter */Custom/MyReport.xdo*. Leave this field when the execution method is **ESS Job**.
    d. **ESS Job Path**—Enter the path to the folder that contains the ESS job definition. The path begins with */oracle/apps/ess/custom/* for custom ESS jobs.
    e. **ESS Job Name**—Enter the ESS job name.
    f. **Report Parameter List**—Specify the report parameters of the custom query.

       Make sure you specify a random string such as "ABC" in **Report Parameter List** that will be passed to the bind parameter you created in the report definition. If you create a report with a query that doesn't have bind parameters passed from the Cloud EPM, the process fails on the Oracle ERP Cloud side.

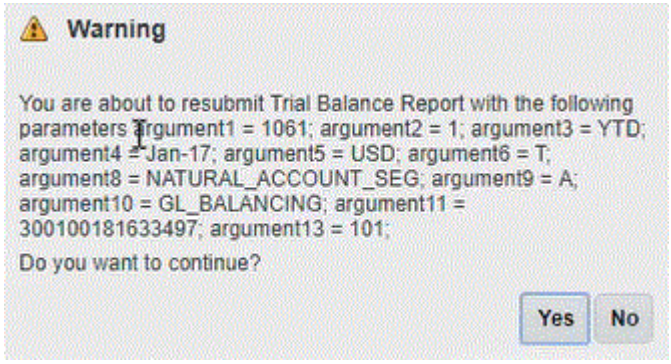
       If you need to identify report parameters, complete the following:

       i. Navigate to **Oracle ERP Cloud**, and from the **Overview** page, select the report and click **Resubmit**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



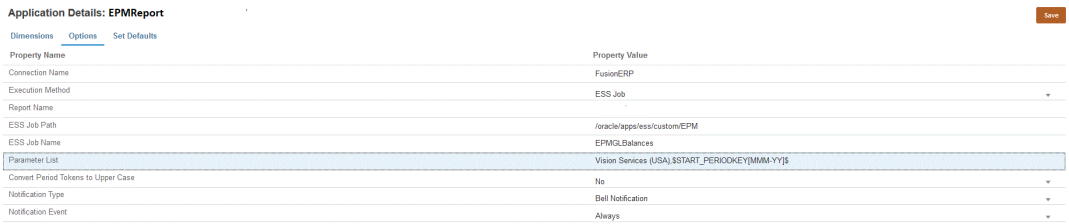
A list of report parameters is generated.



ii. Copy the report parameters shown in the **Warnings** window.

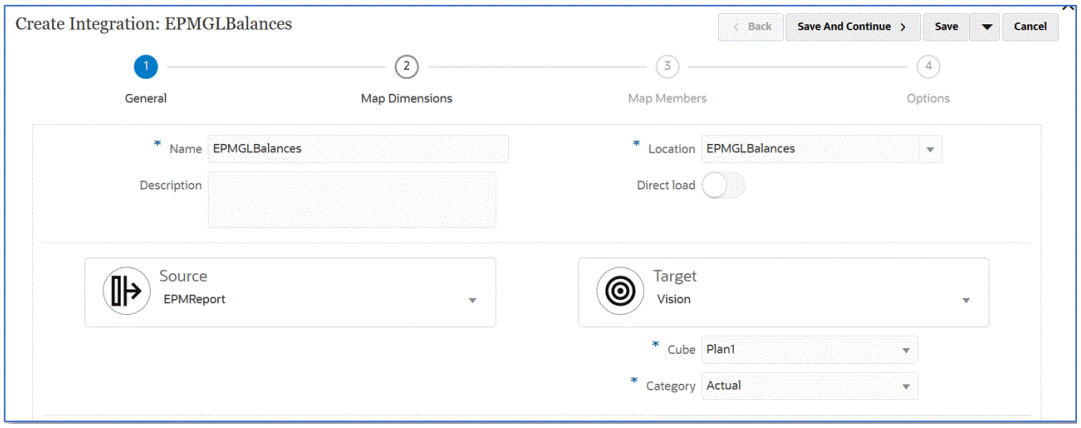
iii. Navigate to Data Integration and paste the report parameter list from the **Warnings** window into the **Report Parameter List** field of your custom query.

Make sure you specify a random string such as "ABC" in the "Report Parameter List" that will be passed to the bind parameter you created in the report definition. If you create a report with a query that doesn't have bind parameters passed from the Cloud EPM, the process will fail on the Cloud EPM side.



13. Create a new integration and select the data source application as the source and the target application. Then create the remaining steps to create the integration.

Copyright © 2017, 2026, Oracle and/or its affiliates.



a. Set up the integration mapping between the Oracle ERP Cloud data source and the target application by building an import format, location, and dimension mappings.

   See Mapping Dimensions.

b. Map members from the source to target.
   See Mapping Members.

c. Select any source and target options.

   See Setting Data Integration Options.

d. Run the integration.

   See Running an Integration.

Loading Oracle ERP Cloud Exchange (FX) Rates to the Cloud EPM

In Data Integration, you can extract exchanges rates (FX rates) from the Oracle ERP Cloud using an ERP Cloud FX Rate adapter. The adapter is used as a data source in Data Integration. The adapter enables customers to select daily conversion rates for specific combinations of foreign currency, dates, and conversion rate types. This feature is only available for the standard load method.

The ERP FX Rate adapter supports period mapping requirements for the FX rate. The source data from the Oracle ERP Cloud is provided by the conversion date. Date-based FX Rate data is converted to EPM periods so it can be easily mapped and loaded to Oracle Fusion Cloud EPM applications.

> **Note**
>
> Oracle Fusion Cloud Enterprise Performance Management customers who wish to use the ERP FX Rate adapter are required to be on Oracle Fusion Financials Update24B or later in order to extract exchange rates from the Oracle ERP Cloud.

Process Description for Loading Foreign Exchanges (FX) Rates Description

These are the steps for loading foreign exchange rates (FX rates) data from the Oracle ERP Cloud to your Oracle Fusion Cloud Enterprise Performance Management business process.

To load exchange rates:

1. Confirm that you are using Oracle Fusion Financials Update24B or later.

Copyright © 2017, 2026, Oracle and/or its affiliates.

For more information, see Update 24B.

2. An Oracle ERP Cloud integration requires that you have the privileges or user role and data access to the Oracle ERP Cloud.

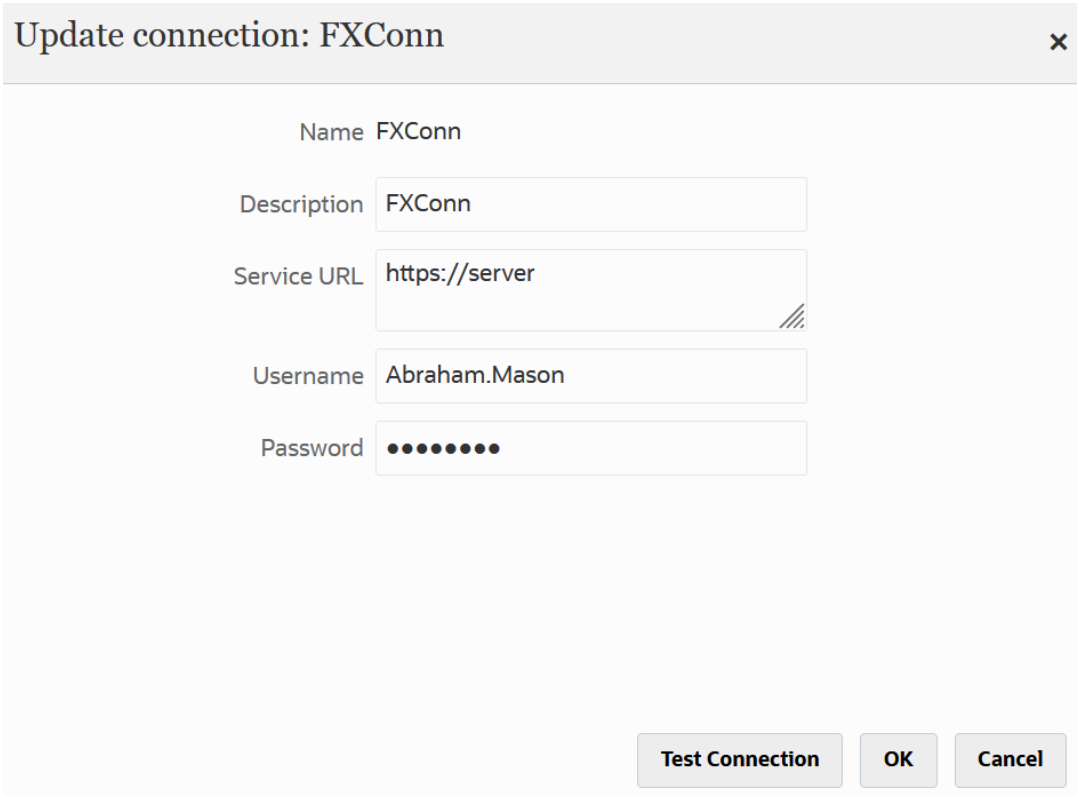
   For more information, see Security Role Requirements for Oracle ERP Cloud Integrations.

3. Register the source system for the **Oracle ERP Cloud** and specify your user credentials.

   This step includes specifying the connection details and testing the connection.

   Cloud EPM customers who wish to use the ERP FX Rate adapter are required to be on Fusion Financials Update24B or later in order to extract exchange rates from the Oracle ERP Cloud.

   For more information, see Configuring an Oracle ERP Cloud Connection.



4. Register the exchange rate application as an Oracle ERP Cloud data source application type and save it.

   a. On the **Applications** page, click (Add icon).

   b. From **Category**, select **Data Source**.

   c. From **Type**, select **Oracle ERP Cloud (Exchange Rates)**.

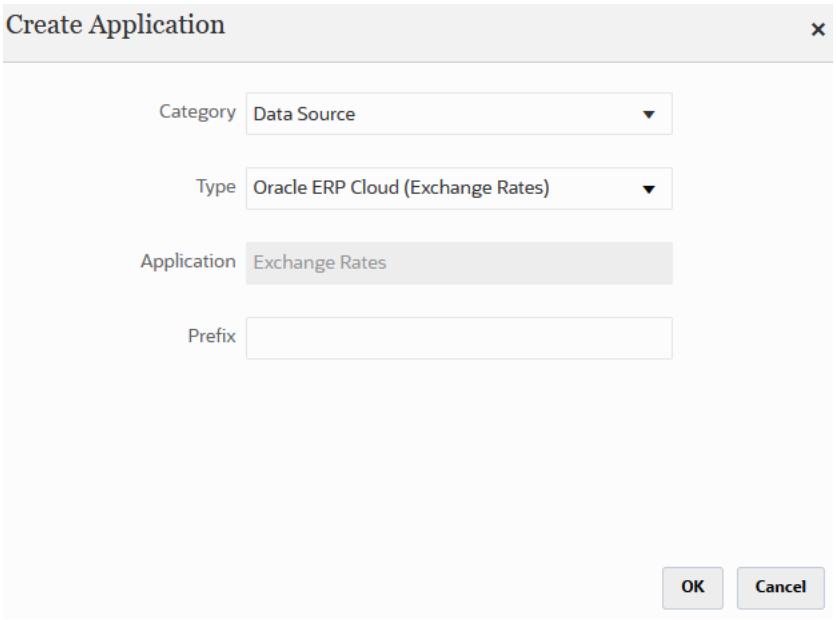
      The exchange rate application is preselected in **Application**.

   d. In **Prefix**, specify a prefix to make the application name unique.

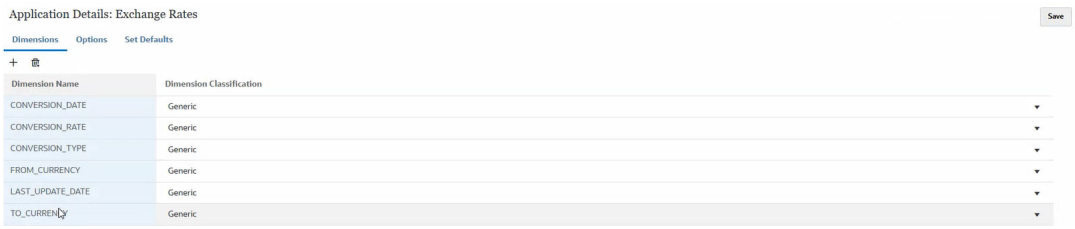
      The prefix is concatenated with the file name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.

   e. Click **OK**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

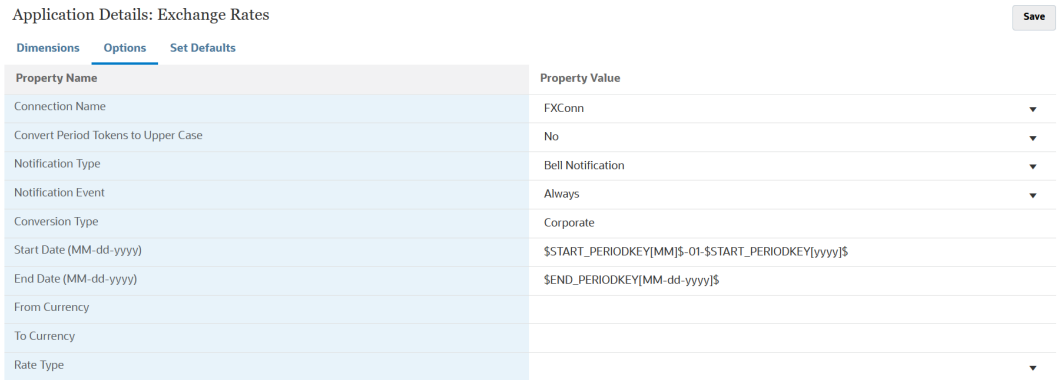


5. From the **Application** page, click ... to the right of the Exchange Rates application, and then select **Application Details**.
6. Select the **Dimensions** tab to confirm the dimensions passed from the Exchange Rates BI Publisher report.



7. From the **Options** tab, then **Connection Name**, specify the connection used to the Oracle ERP Cloud.

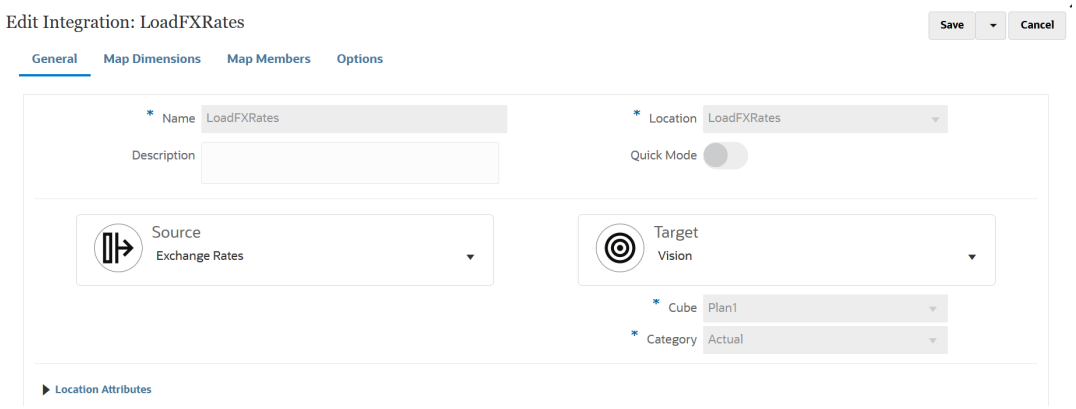
> **Note**
>
> At the application level, you specify only the connection name.



Copyright © 2017, 2026, Oracle and/or its affiliates.

8. Create the integration between the exchange rate source application and the Cloud EPM business process.

   For information on creating an integration, see Creating Direct Integrations.



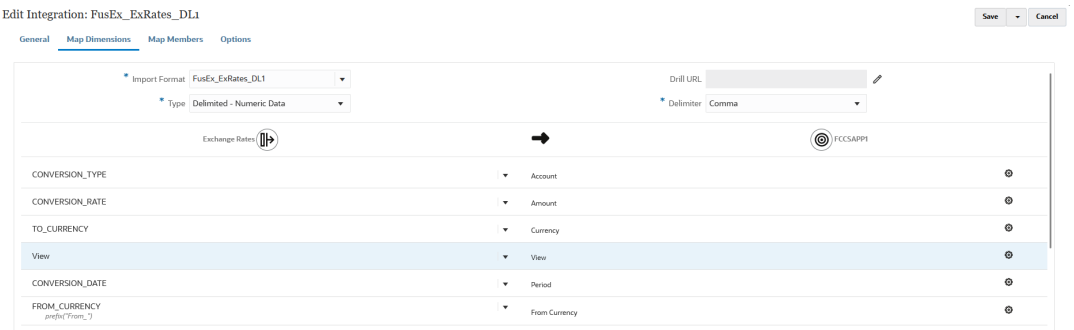
> **Note**
>
> Account Reconciliation customers can extract rates into a file by using a Data Export application as the target. When the extract rates are loaded, customers can then use a custom import script to load it to Account Reconciliation. For more information, see Registering a Data Export File Application.

9. Map the following dimensions between exchange rate source application and the Cloud EPM application.

   Dimensions to map:

   - CONVERSION_TYPE maps to the Account dimension.
   - CONVERSION_RATE maps to the Amount dimension.
   - FROM_CURRENCY maps to the From Currency dimension.
   - VIEW maps to the View dimension.
   - TO_CURRENCY maps to the Currency dimension.
   - CONVERSION_DATE maps to the Period dimension.

   Here is an example of how source exchanges rates are mapped to Financial Consolidation and Close dimensions:



   For more information about mapping dimensions, see Mapping Dimensions.

10. In **Map Members**, map the Amount column to any or all members in the target application.

    For more information, see Mapping Members.

Copyright © 2017, 2026, Oracle and/or its affiliates.

11. From **Options**, specify parameters specific to the exchange rate source and target system at the integration level including: **Conversion Type**, **Start Date (dd-MM-yyyy)**, and **End Date (dd-MM-yyyy)**.

> **Note**
>
> You can specify multiple values for the "Conversion Type", "Start Date (dd-MM-yyyy)", "End Date (dd-MM-yyyy)", "From Currency", and "To Currency" parameters below by separating multiple values using a semi-colon (;) between values. For example to specify "Corporate" and "Spot" for the conversion types, type: Corporate;Spot.

Conversion Type—Specify the conversion rate type to automatically assign a rate when you convert foreign currency journal amounts to functional currency equivalents.

The Oracles General Ledger provides the following predefined daily conversion rate types:

- **Spot:** An exchange rate type to indicate a conversion based on the rate on a specific date. It applies to the immediate delivery of a currency.
- **Corporate:** An exchange rate identifies standardized rates for your company. This rate is generally a standard market rate determined by senior financial management for use throughout the organization.
- **User:** An exchange rate identifies when a foreign currency journal entry has been entered.
- **EMU Fixed:** An exchange rate General Ledger provides automatically when journals are entered (after the EMU effective starting date) using a foreign currency that has a fixed relationship with the Euro.

For more information about conversion rate types, see **Guidelines for Creating Conversion Rate Types**.

12. In **Start Date (dd-MM-yyyy)**, specify the start date to load exchange rates using the format: dd-MM-yyyy.
13. In **End Date (dd-MM-yyyy)**, specify the end date to load exchange rates using the format: dd-MM-yyyy.
14. **(Optional)**: In **From Currency**, specify the source currency value from which to base the exchange rate.

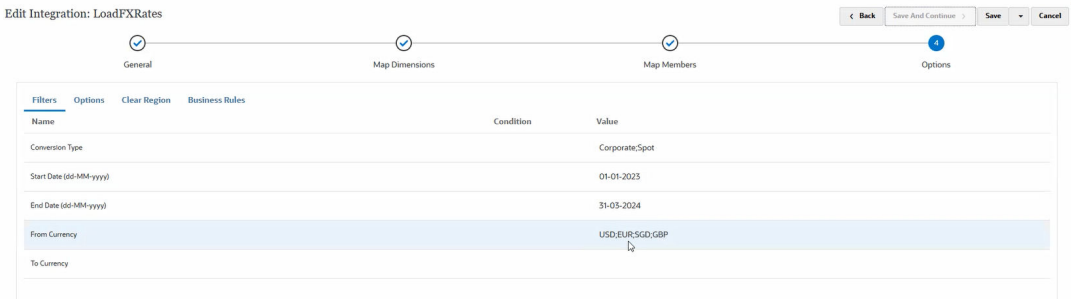
    For example, you could "USD" for the US dollar.

    To include all "From" currencies, leave this field blank.
15. **(Optional)**: In **To Currency**, specify the target currency value.

    For example, you could specify "EUR" for the EURO.

    To include all "To" currencies, leave this field blank.
16. Click **Save**.

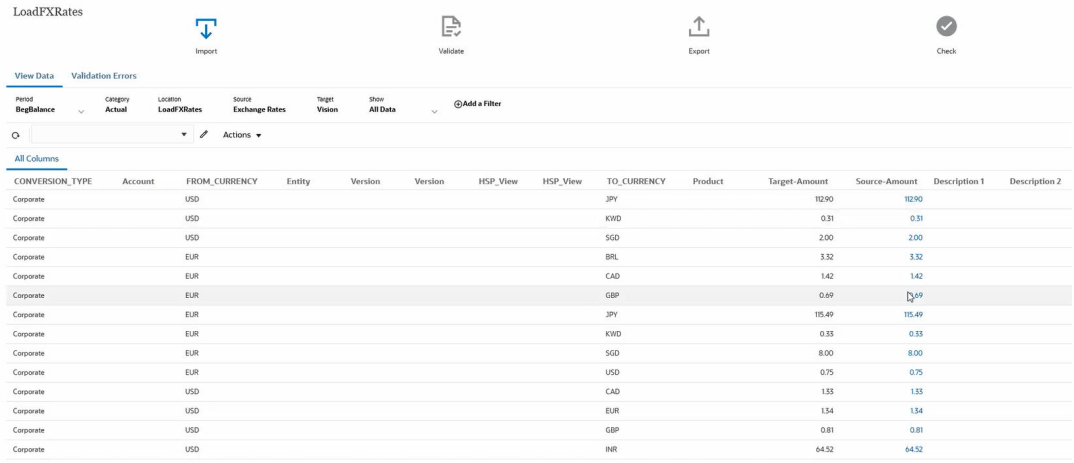
Copyright © 2017, 2026, Oracle and/or its affiliates.



17. Run the integration between exchange rate application and the Cloud EPM application.

    For information about running an integration, see Running an Integration.

    The Workbench below shows the results of integration exchange rate data:



Security Role Requirements for Oracle ERP Cloud Integrations

Oracle General Ledger Security role requirements for Oracle ERP Cloud integrations with the Oracle Fusion Cloud Enterprise Performance Management include:

- Integration User Privileges
- Integration User Predefined Roles
- Integration User Custom Roles
- Allowlist

Integration User Privileges

The required "Integration User" privileges in the Oracle ERP Cloud used for the Oracle General Ledger and Oracle Fusion Cloud Enterprise Performance Management integration are:

Privilege	Description
GL_RUN_TRIAL_BALANCE_REPORT_PRIV	Import data from the Oracle General Ledger to the Cloud EPM.
GL_ENTER_BUDGET_AMOUNTS_FOR_FINANCIAL_REPORTING_PRIV	Write-back data from Cloud EPM to the Oracle General Ledger.
FUN_FSCM_REST_SERVICE_ACCESS_INTEGRATION_PRIV	Execute REST API used to perform the integration

Copyright © 2017, 2026, Oracle and/or its affiliates.

Integration User Predefined Roles

When importing data, you can assign one of the following pre-defined roles to the integration user:

- General Accountant
- Journal Management
- Period Close Management

When importing and writing back data, you can assign the "General Accountant" pre-defined roles to the integration user.

Integration User Custom Roles

You can use a Custom Role to the integration use and then assign the following privileges: When importing data, you can assign one of the following custom roles to the integration user:

Privilege	Description
GL_RUN_TRIAL_BALANCE_REPORT_PRIV	Import data from the Oracle General Ledger to the Oracle Fusion Cloud Enterprise Performance Management.
FUN_FSCM_REST_SERVICE_ACCESS_INTEGRATION_PRIV	Execute REST API used to perform the integration.

When importing data, you can assign one of the following custom roles to the integration user

Privilege	Description
GL_RUN_TRIAL_BALANCE_REPORT_PRIV	Import data from the Oracle General Ledger to the Cloud EPM.
GL_ENTER_BUDGET_AMOUNTS_FOR_FINANCIAL_REPORTING_PRIV	Write-back data from Cloud EPM to the Oracle General Ledger.
FUN_FSCM_REST_SERVICE_ACCESS_INTEGRATION_PRIV	Execute REST API used to perform the integration.

Allowlist

If you have enabled IP Allowlist in the Oracle ERP Cloud, then add the Oracle Fusion Cloud Enterprise Performance Management IP addresses to the list.

Refer to **IP Allowlist for Web Service Calls Initiated by Oracle Cloud Applications (Doc ID 1903739.1)** for details.

Integrating EPM Planning Projects and Oracle Fusion Cloud Project Management (Project Management)

This section explains how to integrate the EPM Planning Projects module (Projects) and Oracle Fusion Cloud Project Management (Project Management) to perform organizational planning and budgeting and to execute projects.

Copyright © 2017, 2026, Oracle and/or its affiliates.

About Integrating EPM Planning Projects and Project Management

Push EPM Planning Projects module (Projects) and Oracle Fusion Cloud Project Management (Project Management) to perform organizational planning and budgeting and to execute projects.

You can integrate the EPM Planning Projects module (Projects) and Oracle Fusion Cloud Project Management (Project Management) to perform organizational planning and budgeting and to execute projects. Develop your strategic corporate plan and budget using EPM Planning Projects and execute and collect costs for approved projects using Project Management. Actual costs are then included in budget analysis, forecasting, and re-planning using EPM Planning Projects.

With this two-way integration, use EPM Planning Projects to develop new projects, create project budgets, and do overall project planning. Then, use Project Management to capture actuals. Bring in the actuals to EPM Planning Projects for budget variance analysis.

With this integration, the same Indirect and Capital projects are visible in both EPM Planning Projects and Project Management depending on the cadence of the synchronization. The capabilities include:

- Transfer projects and budgets created in EPM Planning Projects to Project Management. The strategic budget is created in Project Management as a baseline budget at the resource class level.
- Use the budget approval validation to validate the detailed budgets created by project managers versus. the strategic budgets created in EPM Planning Projects (Optional).
- Transfer actual cost amounts from Project Management to EPM Planning Projects at the resource class level.
- Transfer re-planned budgets from EPM Planning Projects to Project Management at the resource class level.

You use Data Management and Data Integration to drive the integration of data between EPM Planning Projects and Project Management. Data Management and Data Integration provide an out of box solution that enables EPM Planning Projects customers to apply predefined mappings from the Project Management data model to target dimensions. You can also customize and extend these integrations, for example, by applying other mappings as needed to meet your business requirements.

For additional information about the EPM Planning Projects and Project Management integration, see: Unified Planning Budgeting Execution and Analysis of Projects White Paper (Doc ID 2739200.1).

Process Description for Integrating EPM Planning Projects and Project Management

Data Integration provides an out of the box solution that enables customers to apply predefined mappings for the two way integration between EPM Planning Projects and Project Management.

Customers can customize and extend these integrations, for example, when they need to use a different calendar to the business unit other than the default project accounting calendar from the primary ledger calendar.

You must be assigned a *Service Administrator* role to perform most of the tasks below. For more information about EPM roles, see Understanding Predefined Roles.

Copyright © 2017, 2026, Oracle and/or its affiliates.

At a high level, here are the steps for integrating EPM Planning Projects and Project Management:

1. Enable EPM Planning and complete any setup and configuration requirements in EPM Planning Projects.

   For detailed steps on setting up and performing the integration, see Integrating EPM Planning Projects and Project Management.

   See also: Setting Up Data Integration for EPM Planning and Project Management.

2. Assign security privileges and job roles required to use the feature.

   This step requires that you have been assigned BI Publisher privileges and Projects and Budgets File-Based Data Import Processes security roles.

   For more information, see Security Role Requirements for Cloud EPM to Project Management Integrations.

3. In Data Management, select the **Source System** and complete the following:

   a. Add **Project Management** as a source system and identify the application type as **Oracle ERP Cloud**.

   b. **Configure** and then **test** the source connection.

      This step requires that you specify the connection information to the Oracle ERP Cloud including the user name, password, and Services URL.

      The test connection part of this step fails when the user has been assigned only Project Management roles but has not been provisioned for GL integration roles. However, integrations run with only Project Management roles.

      For information on GL integration roles, see Security Role Requirements for Oracle ERP Cloud Integrations.

4. Register the primary EPM Planning Projects application with an **OEP_PFP** input cube or **All Input Cubes**.

   For more information, see Registering the EPM Planning Projects Application.

5. In Application Details for the primary EPM Planning Projects application, verify that the **Project**, **Project Element**, and **Resource Class** dimensions have been categorized to the appropriate dimension classifications.

   For more information, see Classifying Project Dimensions in the EPM Planning Projects Application.

6. Register the **Reporting plan type** application with the **PFP_REP** cube as an application.

   For more information, see Register the Reporting Type Application.

7. Register the **Dimension** applications.

   Only the Entity and Custom dimensions are mandatory. Other dimension types, such as the Account, Scenario, and Version can be optionally deleted after the registration.

   For more information, see Registering the Dimensions Application.

8. On the **Application** page, verify that the list of applications includes the following:

   a. Primary EPM Planning Projects

   b. Reporting Type

   c. Dimension

   If any of the above applications have not been registered, the initialization of the integration fails.

Copyright © 2017, 2026, Oracle and/or its affiliates.

9. On the **Application** page, click to the right of the primary EPM Planning Projects application, and then select **Initialize Integration**.
10. On the **Initialize Integration** page, select the **Project Portfolio Management (PPM)** initialization type, connection to the Oracle ERP Cloud, and any prefix used to designate an integration iteration.

    For more information, see Initializing the Integration.

    This process creates all out of the box integration definitions required to integrate the EPM Planning Projects and Project Management including:

    - Export Project Budgets
    - Export Projects
    - Exported Project Status Sync
    - Import Project Actuals
    - Import Project Budgets
    - Import Project Organizations
    - Import Projects
    - Imported Project Status Sync

    For a description of the integration definitions created after the initialization, see Integration Definition Reference.
11. If the application has multiple integration definitions, define the required mapping for the following integration definitions:

    - Exported Project Status Sync
    - Export Project Budgets
    - Import Actuals
    - Import Budgets
    - Import Project Properties

    For more information, see: Post Initialization Mappings.
12. If the application has additional custom dimensions, define the required mapping for these dimensions on the Map Dimensions page in Data Integration.

    For more information, see Mapping Custom Dimensions.
13. Add period mappings to define the period and years relationships between EPM Planning Projects and Project Management during the integration.

    By default, period mappings between EPM Planning Projects and Project Management are set up automatically during the application creation. For the integration, both EPM Planning Projects and Project Management use the default calendars set up in Period Mappings defined in Data Management.

    If you need to add or modify a period mapping, you can customize period mappings using the Application mapping and Source mapping options defined in Data Management. Period mappings can be defined at two levels:

    - Application period mappings are used to define any special period mapping used in the export of budgets to Project Management. For more information, see Defining Application Mappings.

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Source period mappings are required to create calendar period mappings for actuals and budgets in EPM Planning Projects and Project Management. For more information, see Applying Source Mappings.

Security Role Requirements for Cloud EPM to Project Management Integrations

Project Management security role requirements to integrate with the Oracle Fusion Cloud Enterprise Performance Management include:

- BI Publisher Security
- Projects and Budgets File-Based Data Import Process Security

BI Publisher Security

To run Oracle Business Intelligence Publisher Reports used in this integration, you need these roles:

Role Name	Role Code
Import Project Data into Third-Party Software	PJF_IMPORT_PROJECT_INTO_THIRD_PARTY_SOFTWARE_PRIV_OBI
Get Project Setups	PJF_GET_PROJECT_SETUPS_PRIV_OBI

Projects and Budgets File-Based Data Import Process Security

To run the Projects and Budgets file-based data import (FBDI) processes used in this feature, you need these privileges:

Projects and Budgets FBDI processes roles include:

Privilege Name	Privilege Code
Run Import Projects Process	PJF_RUN_IMPORT_PROJECT_PROCESS_PRIV
Run Import Project Budgets Process	PJO_RUN_IMPORT_PROJECT_BUDGET_PROCESS_PRIV

Role names and codes include:

Role Name	Role Code
Run Import Projects Process	PJF_RUN_IMPORT_PROJECT_PROCESS_PRIV_OBI
Run Import Project Budgets Process	PJO_RUN_IMPORT_PROJECT_BUDGET_PROCESS_PRIV_OBI

The Duty name and code includes:

Duty Name	Duty Code
FSCM Load Interface Administration	ORA_FUN_FSCM_LOAD_INTERFACE_ADMIN_DUTY

Copyright © 2017, 2026, Oracle and/or its affiliates.

Registering the Project Management Source

When integrating sources from Project Management, you first create and register the source system and then specify the application type: **Oracle ERP Cloud**.

To add Project Management as a source:

1. Launch Data Management.
2. On the **Setup** tab, under **Register**, select **Source System**.
3. In **Source System**, click **Add**.
4. Enter the source system details:
   a. In **Source System Name**, enter the source system name.

      Enter the name you want to use for the Project Management source such as **Project Management**.

   b. In **Source System Description**, enter a description of the source system.

   c. In **Source System Type**, select **Oracle ERP Cloud**.

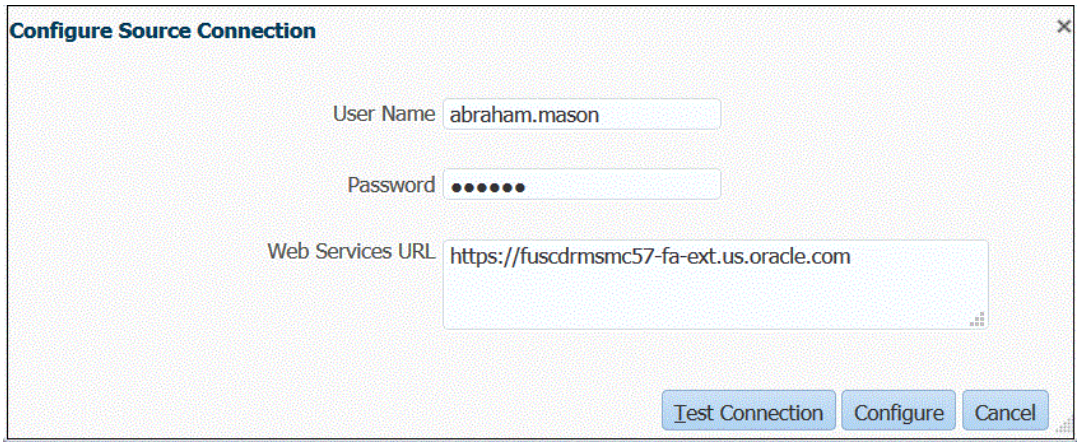
   d. Leave **Fusion Budgetary Control** unchecked.

   e. Leave **Drill Though URL** blank.

   f. Leave **Application Filter** blank.

5. Click **Configure Source Connection**.

   The source connection configuration stores the Oracle ERP Cloud user name and password, and Service URL.



6. In **User Name**, enter the Oracle ERP Cloud user name.

   Enter the name of the Oracle ERP Cloud user who launches the process requests to send information between EPM Planning Projects and Project Management.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> Web services require that you use your native user name and password and not your single sign-on user name and password.

7. In **Password**, enter the Oracle ERP Cloud password.

   You must update this password any time you change your Oracle ERP Cloud password.

8. In **Services URL**, enter the server information for the Fusion web service. For example, enter: `https://server`.

9. Click **Test Connection**.

   The Test Connection step fails if the user has been assigned only Project Management roles. In addition, the user should be provisioned for the GL Integration role. However; integrations run with only Project Management roles.

   For information on GL integration roles, see **Security Role Requirements for Oracle ERP Cloud Integrations**.

10. Click **Configure**.

    The confirmation "Source system [*source system name*] has been configured successfully" is displayed.

11. Click **Save**.

Defining the Integration

Related Links:

- Registering the EPM Planning Projects Application
- Classifying Project Dimensions in the EPM Planning Projects Application
- Register the Reporting Type Application
- Registering the Dimensions Application

Registering the EPM Planning Projects Application

The first step in defining the integration is to register the primary EPM Planning Projects application and select the input cube from the EPM Planning Projects module.

To register an EPM local application:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.

2. On the **Applications** page, click (Add icon).

3. On the **Create Application** page, and then **Category**, select **EPM Local**.

4. From **Application**, select the primary Planning Projects application.

5. From **Cubes**, select **Input Cubes - OEP_PFP**.

   The OEP_PFP input cube has the only dimension applicable for Project Financial Planning (PFP). To show all dimensions across the input cube, select **All Input Cubes**.

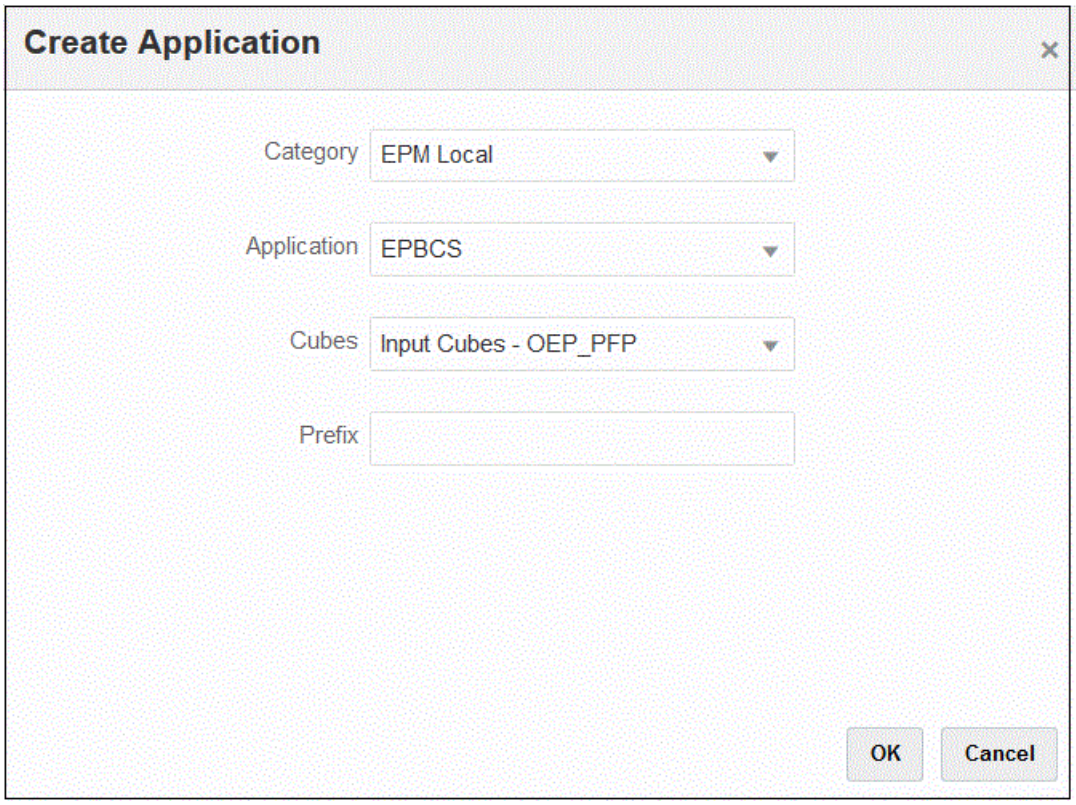
6. **Optional**: In **Prefix**, specify a prefix to make the application name unique.

Copyright © 2017, 2026, Oracle and/or its affiliates.

If you are registering the application for the out of box integration, you do not need to specify a prefix.

The prefix is concatenated with the application name to form a unique application name.

Typically, you prefix an application name when you want to associate it with a selected integration. Multiple integration definitions can be created that address different business requirements. For example, you could have one integration that references the default calendar periods from Project Management and another that has custom mappings for an alternate calendar. If you are initializing the out of box integration, you do not need to specify a prefix.



7. Click **OK**.

Classifying Project Dimensions in the EPM Planning Projects Application

The second step in defining the integration definition is to reclassify the **Project**, **Project Element**, and **Resource Class** dimensions in the EPM Planning Projects application. This is necessary when dimensions have been renamed from the default names in Planning. For Data Integration to identify the correct dimension, classify the dimensions as "Project," "Project Element," and "Resource Class" so that they correspond to the same dimensions in Project Management.

To categorize the Project, Project Element, and Resource Class dimensions:

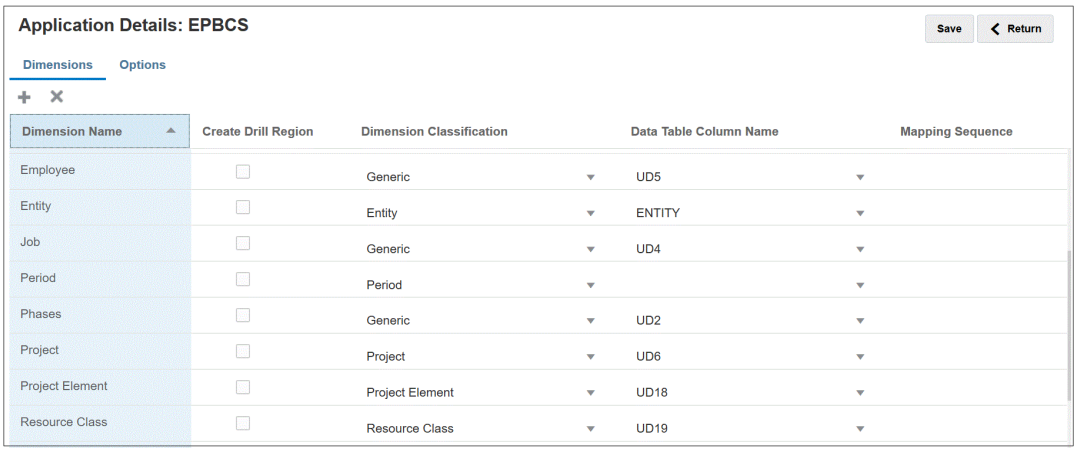
1. From the **Application** page, click ... to the right of the EPM Planning Projects application, and then select **Application Details**.
2. Select the **Project** from the dimension name row and then from the **Dimension Classification** drop-down, select **Project**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

3. Select the **Project Element** from the dimension name row and then from the **Dimension Classification** drop-down, select **Project Element**.
4. Select the **Resource Class** from the dimension name row and then from the **Dimension Classification** drop-down, select **Resource Class**.
5. Click **Save**.

   The remaining applications dimensions for the integration definition do not need to be re-categorized.

   The dimension classifications required for the integration definition are shown below:



Register the Reporting Type Application

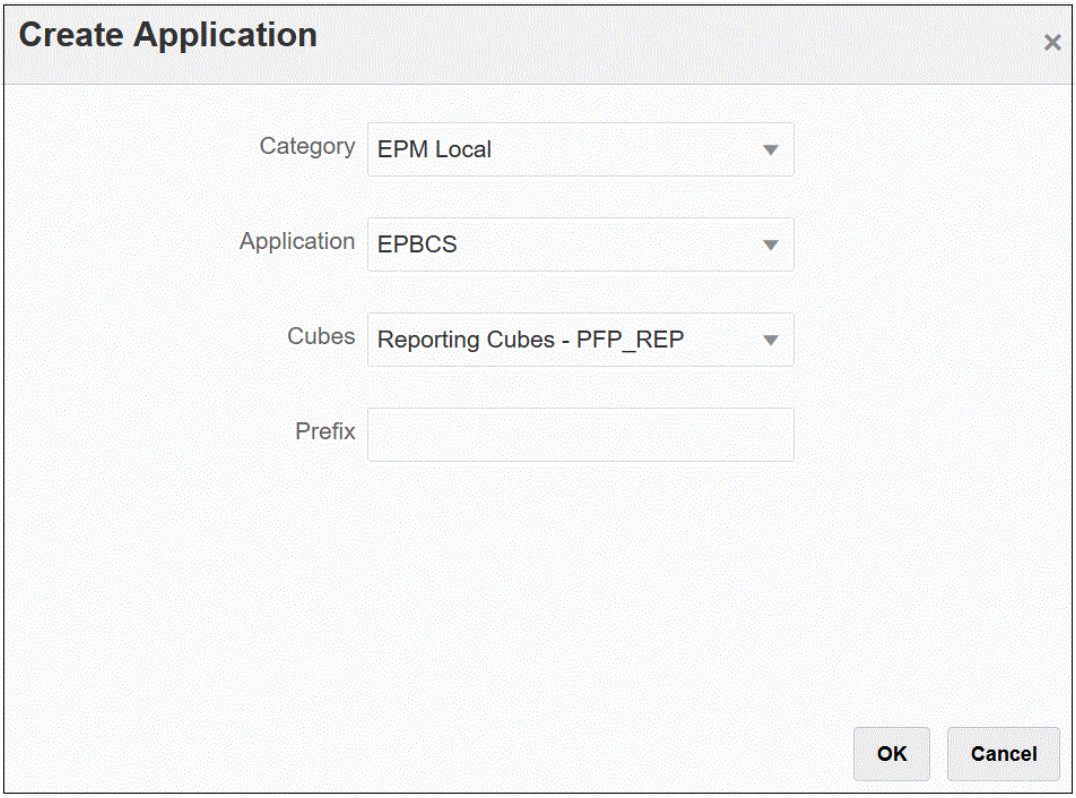
The third step in defining the integration definition is registering the Reporting Type (the project reporting cube (PFP_REP)) application.

To register the Reporting Type application:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, click (Add icon).
3. On the **Create Application** page, and then **Category**, select **EPM Local**.
4. From **Application**, select the primary EPM Planning project application.
5. From **Cubes**, select the **Reporting Cubes - PFP_REP** input cube.
6. **Optional**: In **Prefix**, specify the same prefix as you selected for the OEP_PFP input cube application.

   If you are registering the application for the out of box integration, you do not need to specify a prefix.

Copyright © 2017, 2026, Oracle and/or its affiliates.



7. Click **OK**.

Registering the Dimensions Application

The fourth step in defining the integration definition is registering the Dimensions application, which includes:

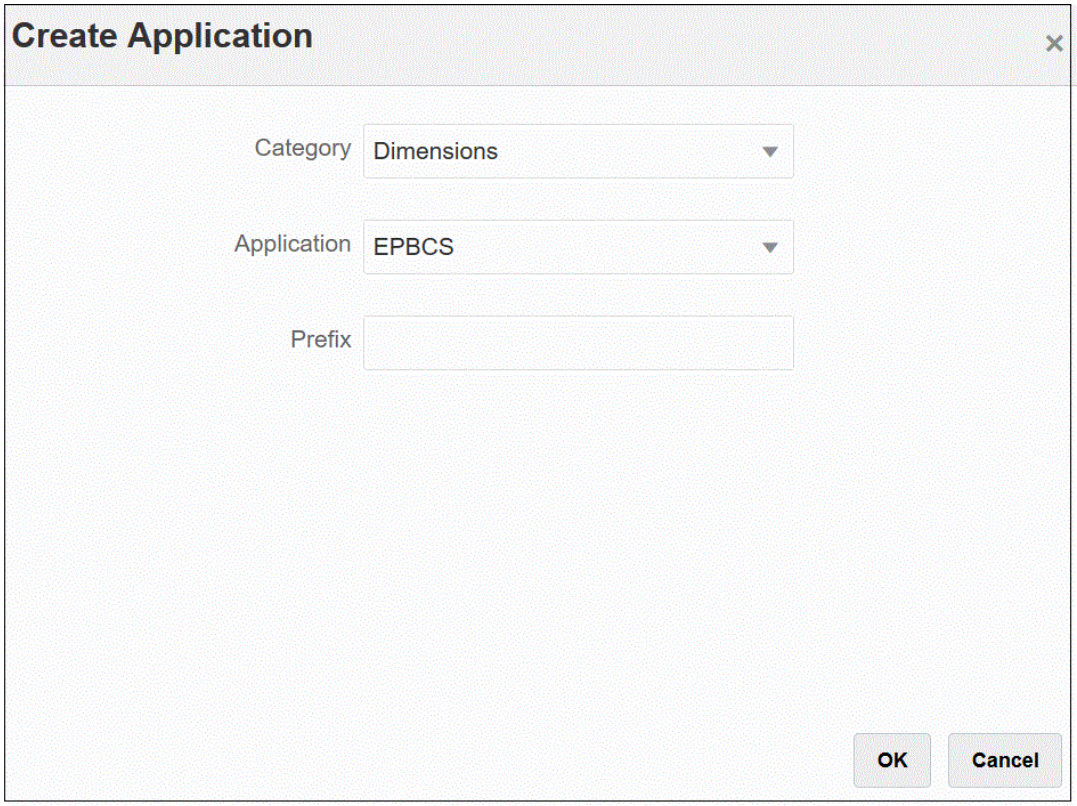
- custom dimensions
- existing custom members of entities, projects, jobs, employee, material etc. as applicable
- project dimensions metadata information used to create members in any of the custom applications

To register the Dimensions application:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, click (Add icon).
3. On the **Create Application** page, and then **Category**, select **Dimensions**.
4. From **Application**, select the primary EPM Planning Projects application.
5. **Optional**: In **Prefix**, specify the same prefix as you selected for the OEP_PFP input cube application.

   If you are registering the application for the out of box integration, you do not need to specify a prefix.

Copyright © 2017, 2026, Oracle and/or its affiliates.



6. Click **OK**.

Initializing the Integration

Initializing the integration creates all out of the box integrations definitions required to integrate EPM Planning Projects and Project Management.

Before initializing the integration, be sure the following applications are registered in Data Integration:

1. Primary EPM Planning Projects
2. Reporting Type
3. Dimension

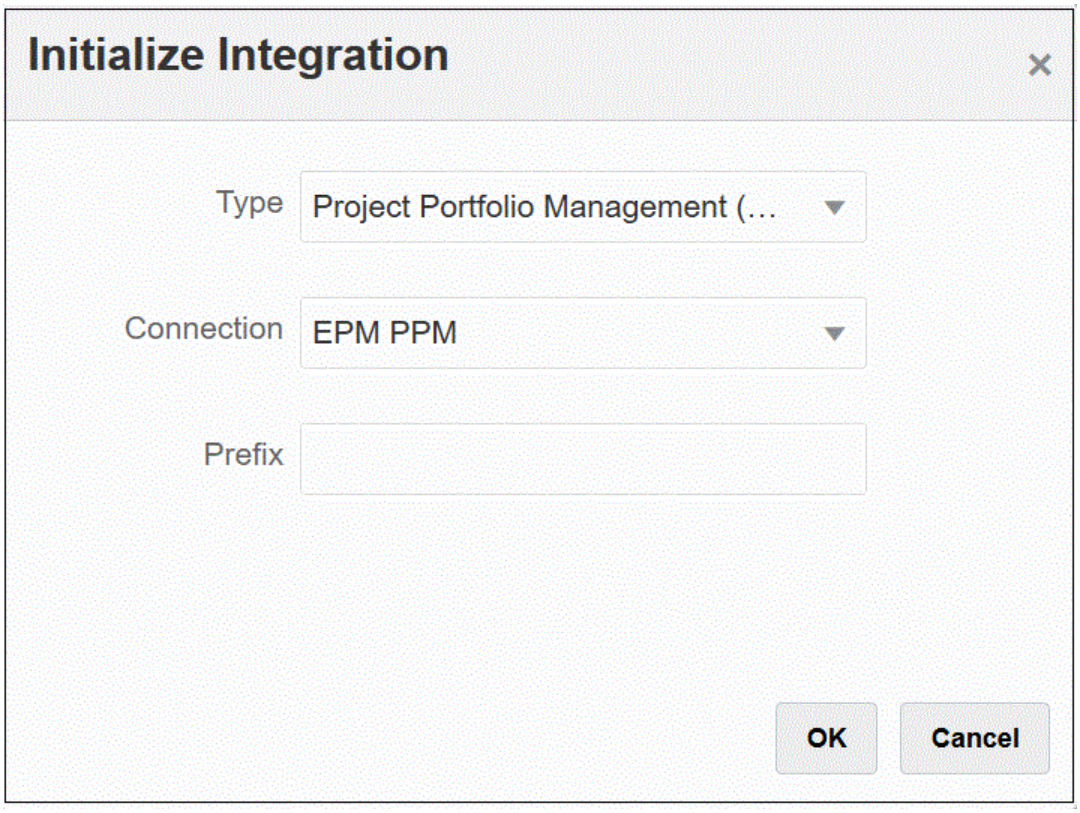
In the following example, "EPBCS" is the primary Planning Projects application, "EPBCS-PFP_REP" is the Reporting Type application, and "EPBCS - Custom" is the Dimension application.

To initialize the integration:

1. On the **Application** page, click ... to the right of the primary EPM Planning Projects application, and then select **Initialize Integration**.
2. On the **Initialize Integration** page, from the **Type** drop-down, select **Project Portfolio Management (PPM)**.
3. From **Connection**, select the source system connection name setup for the EPM Planning Projects and Project Management integration.

Copyright © 2017, 2026, Oracle and/or its affiliates.

This connection has been set up in the source system registration. For more information, see Registering the Project Management Source.

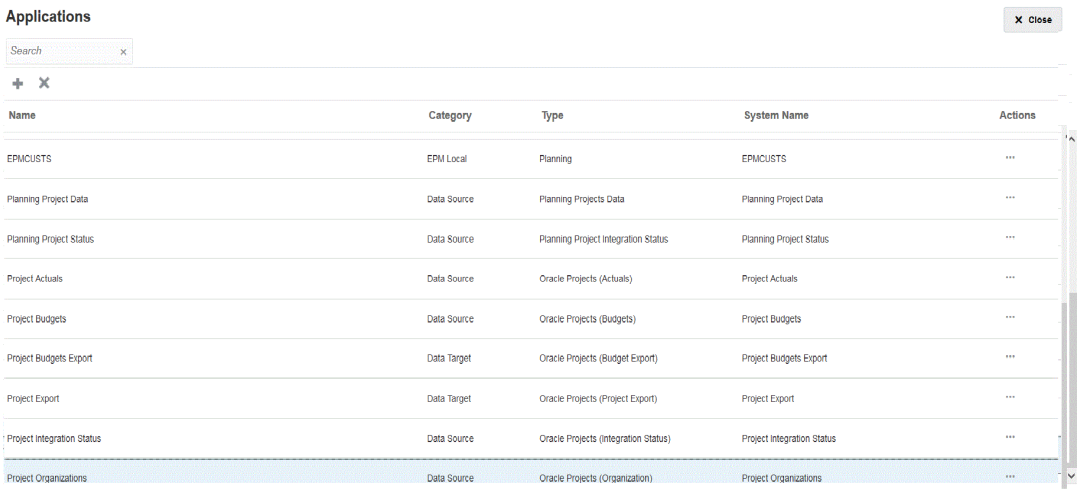


4. **Optional**: In **Prefix**, specify a prefix to the integration definition when using multiple integration definitions.

   You add a prefix to an integration definition to distinguish it from other integration definitions. Multiple integration definitions can be created that address different business requirements. For example, you could have one integration that references the default calendar periods from Project Management and another that has custom mappings for an alternate calendar. If you are initializing the out of box integration, you do not need to specify a prefix.

5. Click **OK**.

   The out of box data source applications are loaded to the Application page.



Copyright © 2017, 2026, Oracle and/or its affiliates.

For a description of the integration definitions created after the initialization, see Integration Definition Reference.

Integration Definition Reference

After the initialization of the EPM Planning Projects and Project Management integration has successfully completed, Data Integration loads the following out of box integration definitions to the Application page.

Integration Definition	Description
Export Project Budget	Export Budgets of approved projects from EPM Planning Projects to Project Management.
Export Projects	Exports approved projects from EPM Planning Projects to Project Management.
Exported Project Status Sync	Updates the integration status in EPM Planning Projects of projects whose budgets have been exported to Project Management. Launches automatically after Export Project Budget completes.
Import Project Actuals	Imports Actuals of projects that are present in EPM Planning Projects from Project Management.
Import Project Budgets	Imports Budgets of projects that are present in the EPM Planning Projects from Project Management.
Import Project Organizations	Imports the Entity hierarchy from the Project Management to EPM Planning Projects.
Import Project Properties	Imports project properties such as the start date and end date from Project Management when projects are imported from Project Management. Launches automatically after Import Projects completes.
Import Projects	Imports projects from Project Management, which are not yet available in EPM Planning Projects.
Imported Project Status Sync	When projects are successfully imported into EPM Planning Projects from Project Management, the status is marked as integrated in EPM Planning Projects. Launches automatically after Import Project Budget completes.

Post Initialization Mappings

Multiple integration definitions enable you to manage project integrations across multiple currencies, accounting periods, and organizational boundaries. You can create multiple integration definitions by prefacing an integration definition with a different name and then associating it with a unique calendar. In this case, you also need to further define settings in each of the integration definition to identity unique values specific to the integration, such as the business unit, currency, or calendar.

For example, suppose you have set up your export business and your domestic sale business as two separate business units: "export" and "domestic sales." The "export" business unit may use accounting periods that differ from the accounting period selected for domestic sales. In this case, the administrator needs to explicitly define the entity or business unit used for the integration definition.

Copyright © 2017, 2026, Oracle and/or its affiliates.

You can add or modify mappings by clicking next to the integration project definition file on the Integration page and then selecting Options. Then from the Edit Integration page, select the Options or Filters tab.

The following table describes the project integration definition file where you may need to define the specific settings depending on the integration definition.

Project Integration Definition Name	Mapping Requirements
Export Project	On the **Filters** tab, replace the "Entity" value in the Value field with the business unit associated with the specific integration definition.
Export Project Budget	On the **Filters** tab, replace the "Entity" value used in the predefined Org filter condition with the business unit associated with the specific integration definition. For example, if the filter condition shows: @ILvl0Descendants("OEP_Total Entity") and your business unit is "Domestic Sales" for the integration definition, replace ("OEP_Total Entity") with ("OEP_Domestic Sales").
Export Project Actuals	On the **Options** tab, from **Period Mapping Type**, select **Explicit**. Then from the **Calendar** drop-down, select the name of the source period mapping calendar used for exporting project actuals. If the **Calendar** field is empty, all actuals are exported provided mappings exists between EPM Planning Projects and Project Management. For more information, see Applying Source Mappings.
Import Project	On the **Filters** tab, specify the "Business unit" value used for the specific integration definition in the Value field. In **Maximum Age of Past Projects in Days**, specify the number of days to include for both active and closed projects in the integration definition. For example, if you enter a value of **10**, the project includes projects finished in the last 10 days. But if you enter **365**, then even a project one year old is migrated, but not older than that.
Import Project Budgets	On the **Filters** tab, specify the "Business unit" value used for the specific integration definition in the Value field. In **Maximum Age of Past Projects in Days**, if there are multiple currencies, you have the option to include past projects that are in an active status. You can do this by indicating the number of days in the past you want to include. By default, the value is considered as "0." Only projects that have the finish date today or later - and not earlier - are considered when importing. All active future ending projects are always imported.
Import Project Actuals	On the **Filters** tab, specify the accounting calendar name in the **Value** field. On the **Options** tab, from **Period Mapping Type**, select **Explicit**. Then from the **Calendar** drop-down, select the name of the source period mapping calendar used for exporting project actuals. If the **Calendar** field is empty, all actuals are loaded provided mappings exists between EPM Planning Projects and Project Management. For more information, see Applying Source Mappings.

Copyright © 2017, 2026, Oracle and/or its affiliates.

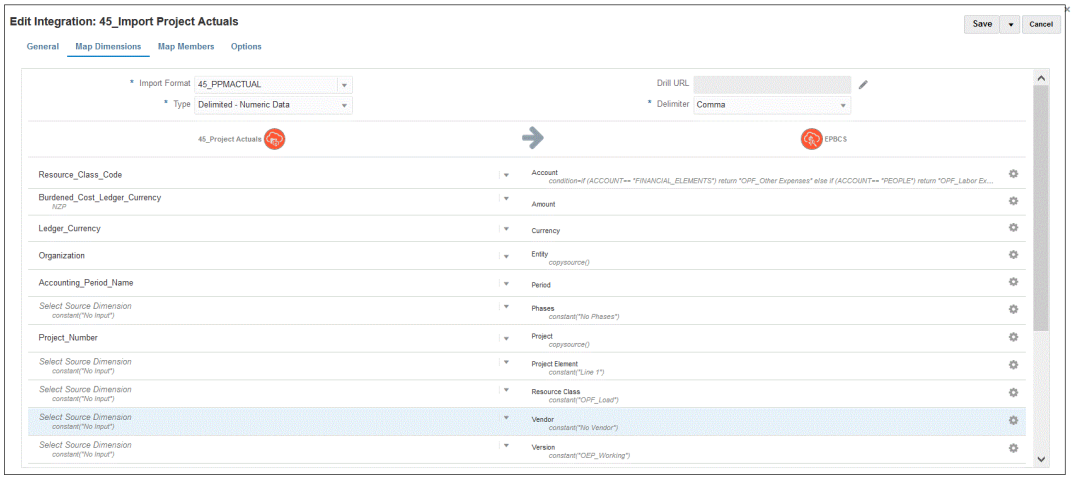
Project Integration Definition Name	Mapping Requirements
Import Project Properties	On the **Filters** tab, specify the "Business unit" value used for the specific integration definition in the **Value** field. In **Maximum Age of Past Projects in Days**, if there are multiple currencies, you have the option to include past projects that are in an active status. You can do this by indicating the number of days in the past you want to include. By default, the value is considered as "0." Only projects that have the finish date today or later - and not earlier - are considered when importing. All active future ending projects are always imported. For example, if you enter a value of **10**, the project includes projects finished in the last 10 days. But if you enter **365**, then even a project one year old is migrated, but not older than that.

Mapping Custom Dimensions

If the EPM Planning Projects or Project Management application has custom dimensions, you must define the required mapping for the source to target dimensions in the integration definition. For example, if the Import Project Actuals integration definition was created to accommodate a project-related business unit that uses a currency other than the ledger currency of the business unit, then map the custom currency dimension on the Map Dimensions page in Data Integration.

To map a custom dimension:

1. From the **Data Integration** home page, click to the right of the integration, and then select **Map Dimensions**.
2. In the **Select Source Dimension** column in the left column, select the name of the source dimension from the drop-down to assign to the target dimension in the right column.



3. Click **Save**.

Defining Period Mappings

Period mappings define the mapping between and Project Management calendars and the EPM Planning Projects application year or periods. You can define period mappings in two ways:

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Default Period Processing
- Explicit Period Processing

Use default period processing when EPM Planning Projects and Project Management use consistent period definitions and period naming. For example, when a monthly calendar is used in both systems and the names of the periods like Jan-20 match both systems, then you use the Period Mapping Type of Default in the integration. No additional mapping is required.

Use explicit period processing when EPM Planning Projects and Project Management period definition or period naming conventions are different. In this case, you are required to define an Application Period and Source Period mapping.

- Use Application Period Mapping to define the Period Names used in Project Management. For more information, see Defining Application Mappings.
- Use Source Period Mapping to define the mapping between the periods in EPM Planning Projects and Project Management. For more information, see Applying Source Mappings.

Defining Application Mappings

Use the application mappings to specify the application period mappings for the Project Management budgets export target application. The mappings that you create here apply only to an individual Project Management export target application. If no application mappings are selected for the export target application, the system uses the global mappings defined for periods in global mappings.

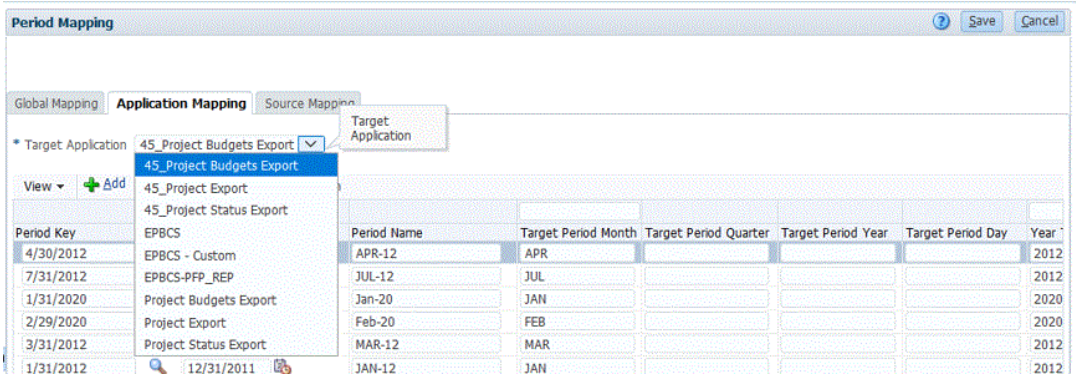
> **Note**
>
> In Project Management, Accounting Calendar of 12 periods which include monthly, 4-4-5, 5-4-4, and 4-5-4 are supported. The 4-4-4 Accounting Calendar is not supported.

To create period mappings for an application:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Period Mapping**.
2. On the **Period Mapping** page, select the **Application Mapping** tab.
3. In **Target Application**, select the target application project budgets export target application for which to add or modify an application period mapping.

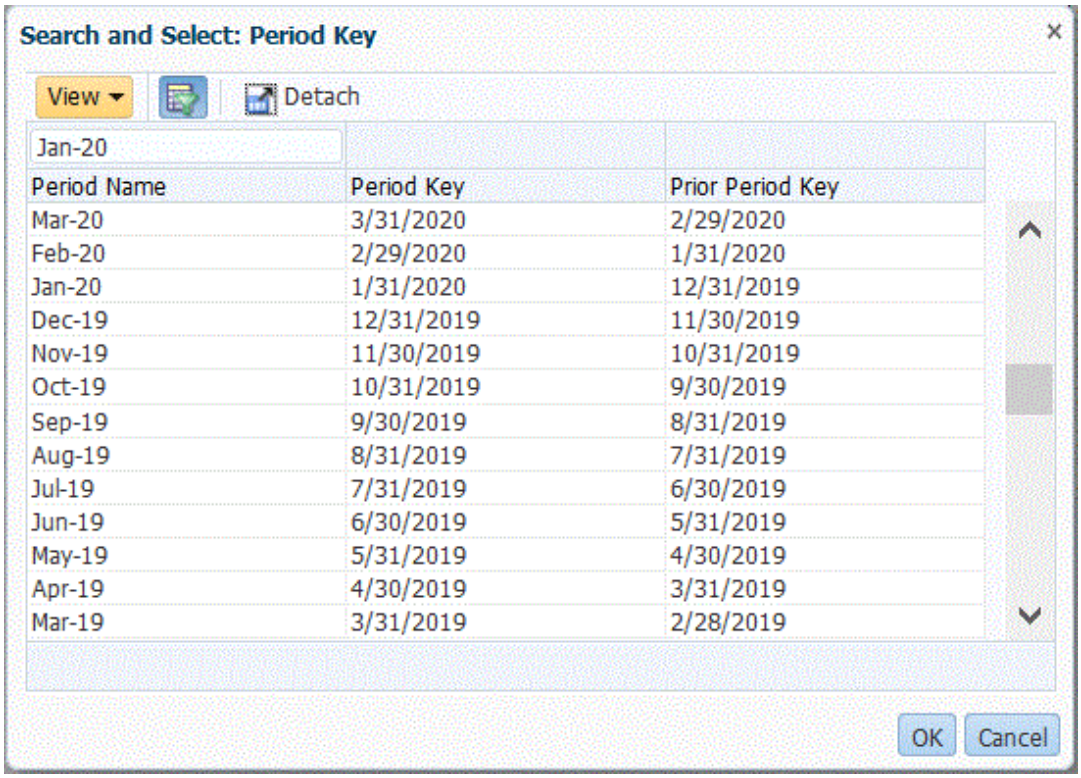
   If you use multiple integration jobs for example, when using multiple calendars, you must select the name of the project budgets export target application created in the initialization and then specify the period mapping. For example, if you have a unique calendar set up for the "45_Project Budgets Export" integration, select this name from the **Target Application** drop-down as shown below.

Copyright © 2017, 2026, Oracle and/or its affiliates.



4. Click **Add**.

   The Search and Select: Period Key page is displayed. This page lists all the global mapping periods not used in the application mappings for the project budgets export target application.



5. Select the Project Management period to add and click **OK**.

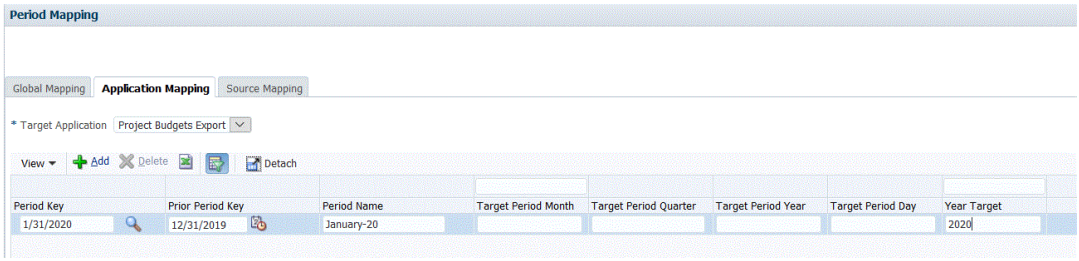
   For example, you might select **Jan-20**.

6. On the **Application** tab, select the **Period Key** for the period that you selected in step 6.

7. In **Period Name**, enter the period name that corresponds to the Project Management period if you are overriding the period name defined for the period in global mappings. Otherwise, the system uses the period name defined for the period in global mappings.

   For example, if the period name is shown as **Jan-20** in Data Management, you can change it to **January-20** if this is how the name has been defined in Project Management.

Copyright © 2017, 2026, Oracle and/or its affiliates.



8. Enter a target period month in **Target Period Month**.

   The target period month is a required entry for all Data Management application mappings, but it does not affect the period name used in the export to Project Management.

9. Click **Save**.

Applying Source Mappings

Source mappings are required to create period mappings between actuals and budgets. If you have initialized separate integrations to accommodate multiple accounting calendars in Project Management, you must define source period mappings for each of the integration definitions.

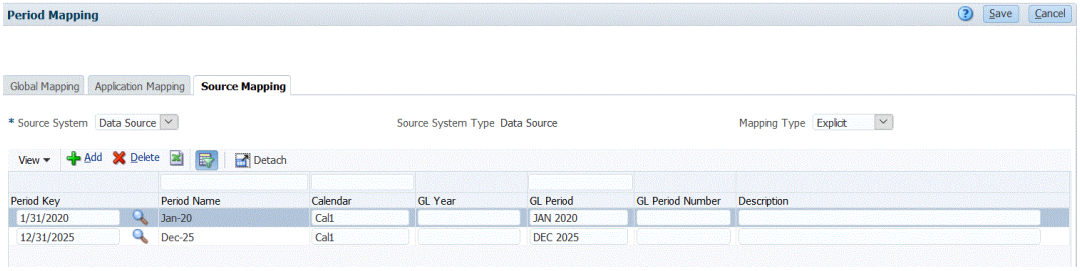
The Source Mapping tab consists of two areas:

- Master—Selects the source system and mapping type.
- Grid—Defines the period mapping. The mapping can be defined only for periods defined on the Global Mapping. New Data Management periods cannot be created on this tab.

> **Note**
>
> Before running an integration, you can choose between Default period mapping and Explicit Period mapping. If you choose Source Period mapping, then source periods are mapped based on the period key and previous period.

To create source mappings:

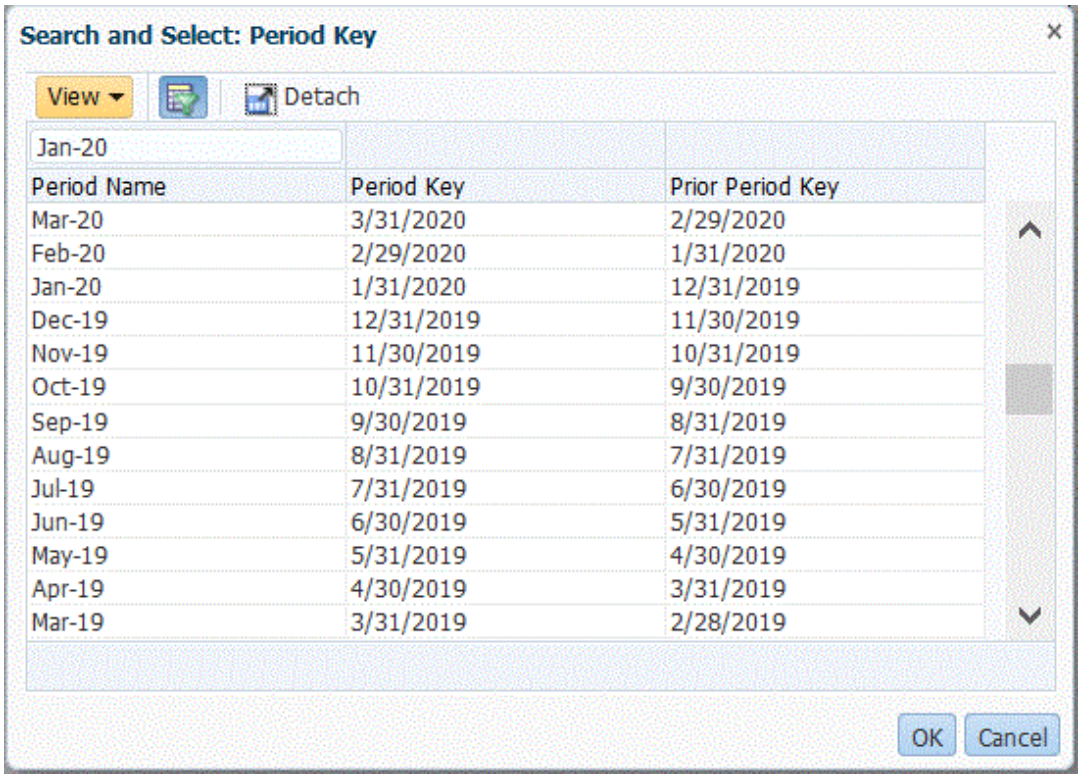
1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Period Mapping**.
2. Select the **Source Mapping** tab.



3. In **Source System**, select **Data Source** for the Project Management source when creating actuals and budgets.
4. In **Mapping Type**, select **Explicit**.
5. Click **Add**.

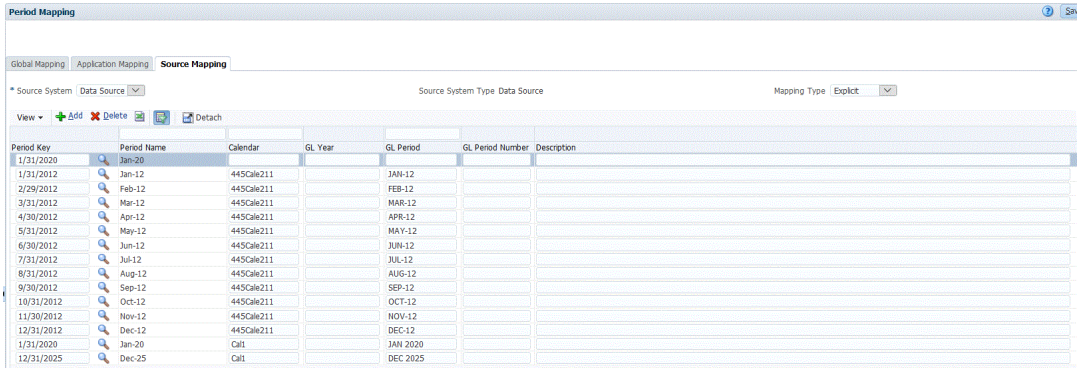
Copyright © 2017, 2026, Oracle and/or its affiliates.

The Search and Select: Period Key page is displayed. This page lists all the global mapping periods that are not used in the source mappings for the source application.



6. Select the period to add and click **OK**.

   For example, you might select **Jan-20**.



7. On the **Source Mapping** tab, enter the source system **Period Name**, and then click **OK**.
8. Enter the source system **Period Key** to identify the EPM Planning Projects period.
9. Enter the source system **Calendar**, name to identify the mapping.
10. In **GL Period**, enter the Project Management name.
11. **Optional**: Enter a description for the mapping.
12. Click **Save**.

> **Tip**
>
> To delete a mapping, select the mapping, and then click **Delete**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Integrating Budgetary Control

Load budgeting, actual or the equivalent amounts from Oracle General Ledger or Budgetary Control to Planning. For budget to actual reporting, budgets can be loaded from Planning and Budgetary Control to Oracle General Ledger.

Budgets can be prepared in Planning. To facilitate budgeting, actual or the equivalent amounts can be loaded from Oracle General Ledger or Budgetary Control to Planning. For budget to actual reporting, budgets can be loaded from Planning and Budgetary Control to Oracle General Ledger. For online validations of consumption against budget, budgets should be loaded fromPlanning and Budgetary Control.

You can import commitments, obligations, other anticipated expenditures, and expenditures from Budgetary Control into your Planning and Planning Budget Revisions modules for use with budget revision, future budget preparation and reporting purposes.

In addition, you can develop comprehensive budgets in Planning and Planning Budget Revisions modules and then using the write-back feature in Data Integration, transfer budgets to Budgetary Control and enable budgetary control validation.

> **Note**
>
> In this release, Budgetary Control only integrates with Planning modules.

For more information about Budget Revisions with Budgetary Control, see Integrating Budget Revisions with Budgetary Control.

Loading Budgetary Control Budget Consumption Balances to Planning

You can load commitments, obligations, expenditures and other anticipated expenditures from Budgetary Control to use in your Planning applications.

Loading Budgetary Control Budget Consumption Balances to Planning Process Description

Describes how to load commitments, obligations, and other anticipated expenditures from Budgetary Control to use in your Planning applications.

You can load commitments, obligations, expenditures and other anticipated expenditures from Budgetary Control to use in your Planning applications.

At a high level, these are the steps to load commitments, obligations, and other anticipated expenditures from Budgetary Control to Planning:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, click (Configure Connections icon).
3. In **Connections**, register, configure, and initialize a source connection for Budgetary Control.

   For more information, see: Configuring a Connection to a Budgetary Control Source.

Copyright © 2017, 2026, Oracle and/or its affiliates.

4. In Applications, register the type **Oracle ERP Cloud** as a data source, specify the connection, select any applicable filter and enable the **Budgetary Control** option.

   The Import Applications button brings over Budgetary Control balance Essbase cubes into Data Integration as Budgetary Control applications. A control budget dimension member within each Budgetary Control target application represents a control budget in Budgetary Control used to load and write back data from and to the Oracle Fusion Cloud EPM.

   For more information, see:

   > **Note**
   >
   > Drill through is not supported in this release for the Budgetary Control integration with the Oracle Fusion Cloud Enterprise Performance Management.

5. Integrate the Planning application from the approved budget in Planning to the Budgetary Control.

   For more information, see Integrating Planning and Budgetary Control Consumption Balances.

6. In Map **Dimensions**, specify an import format to set up the integration mapping between the Budgetary Control (Budget Chart of Accounts) segments and dimensions.
   When you select the source and target, Data Integration populates the source and target columns automatically.

7. For more information, see Mapping Dimensions Between Budgetary Control and Planning.

8. In **Map Members**, map the members between the Budgetary Control application and the Planning application.

   Mapping Members Between Budgetary Control and Planning.

9. In **Options**, specify any filters to limit the results.

   Data Integration automatically creates filters when a integration is created. You can modify the filters as needed but cannot delete them. (If filters are deleted. Data Integration recreates the default value.)

   For information, see Adding Filters.

10. Run the integration to execute the load and transfer commitment, obligation and expenditure amounts to the Planning application from the Budgetary Control application.

    For more information, see Running an Integration.

Configuring a Connection to a Budgetary Control Source

Procedure describes how to configure an Budgetary Control source.

Budgetary Control Balance cubes and their underlying control budgets are eligible for integration with the Oracle Fusion Cloud Enterprise Performance Management when the control budget in Budgetary Control:

- has a "Planning" Source Budget Type.
- does not have Project Portfolio Management keys as a budget segment
- is associated with a Budgetary Control Balances cube

To begin integrating the Budgetary Control with the Cloud EPM, create and register the source system with the source system type Oracle ERP Cloud.

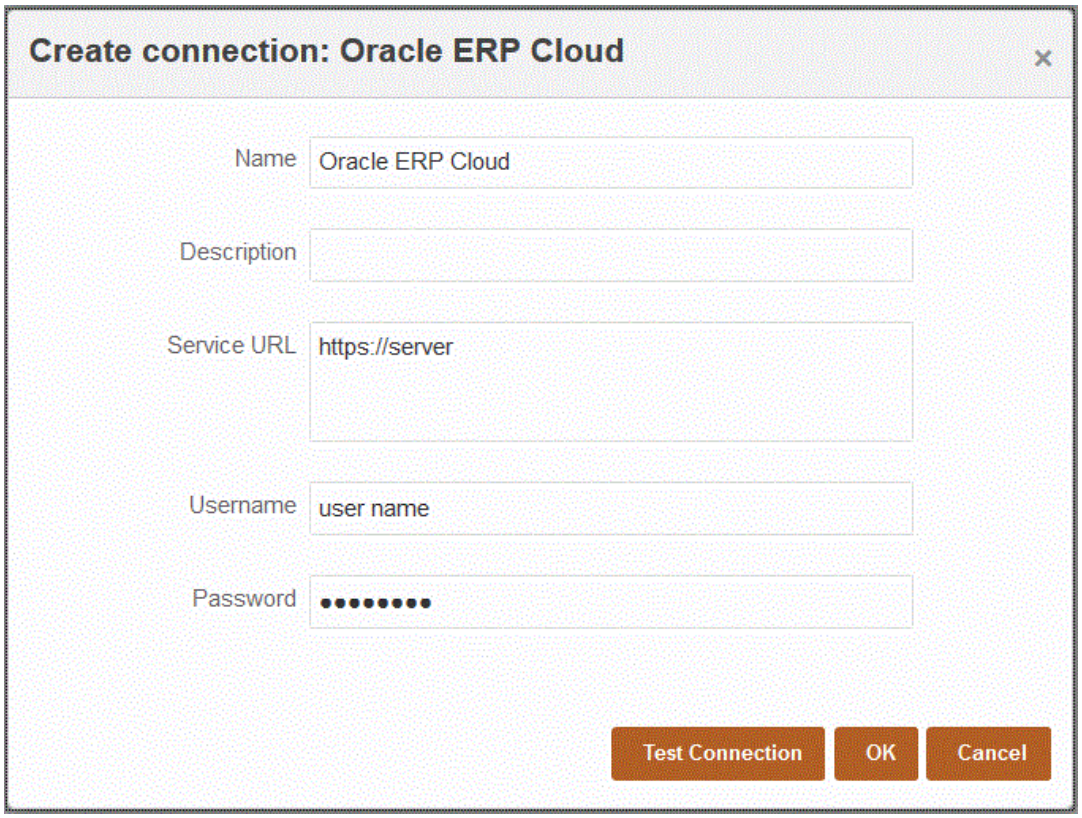
Copyright © 2017, 2026, Oracle and/or its affiliates.

After the source system and connection information are specified, initialize the source system to copy the Budgetary Control information to Cloud EPM as one of many Oracle Essbase target applications.

To add a Budgetary Control source system:

To add a Budgetary Control source system:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, click (Configure Connections icon).
3. On the **Connections** page, select **Oracle ERP Cloud** from the (Add drop-down icon) drop-down.



4. In **Name**, enter the source system name.
5. In **Description**, enter a description of the source system.
6. In **Service URL**, enter the server information for the web services.
7. In **Username**, enter the Oracle ERP Cloud user name.

   Enter the name of the Oracle ERP Cloud user who launches the process requests to send information between Cloud EPM and the Oracle ERP Cloud . This user must have an assigned Oracle General Ledger job role such as "Financial Analyst," "General Accountant," or "General Accounting Manager."
8. In **Password**, enter the Oracle ERP Cloud password.

   You must update this password anytime you change your Oracle ERP Cloud password.
9. Click **Test Connection**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

When the connection has been tested correctly, the information message "Connection to [*source system name*] successful" is displayed.

Registering Budgetary Control as a Data Source and Importing Budgetary Control Applications

When integrating your Planning modules with Budgetary Control, you need to register Budgetary Control as a data source and then import the Budgetary Control applications.

The Import Applications button brings over Budgetary Control balance Essbase cubes into Data Integration as Budgetary Control applications. A control budget dimension member within each Budgetary Control application represents a control budget in Budgetary Control used to load and write back data from and to the Planning

To import and register Budgetary Control applications:

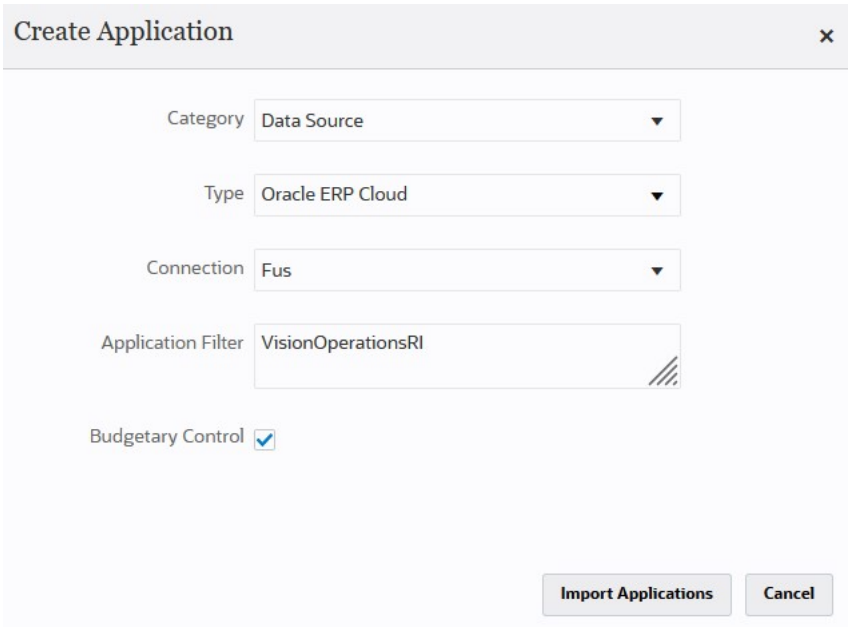
1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, click (Add icon).

> **Note**
>
> Make sure you click on the Application page and not the home page.

3. From **Category**, select **Data Source**.
4. From **Type**, select **Oracle ERP Cloud**.
5. From **Connection**, select the connection name used to connect to the Oracle ERP Cloud.
6. Enable **Budgetary Control**.
7. Click **Import Applications**.

   The Import Applications button brings over Budgetary Control balance Essbase cubes into Data Integration as Budgetary Control target applications. A control budget dimension member within each Budgetary Control target application represents a control budget in Budgetary Control used to load and write back data from and to the Oracle Fusion Cloud Enterprise Performance Management.

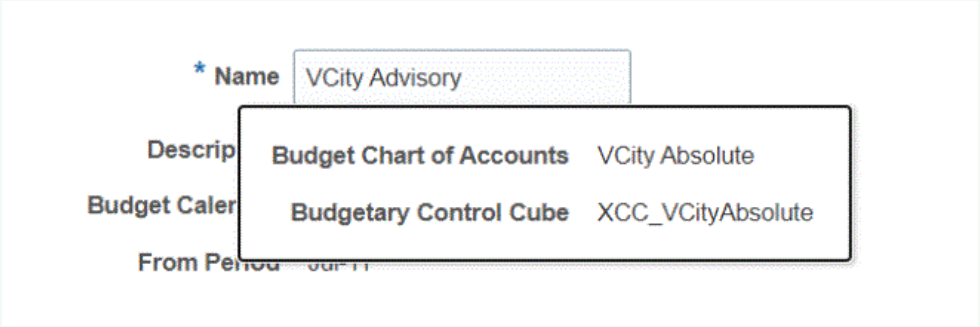
Copyright © 2017, 2026, Oracle and/or its affiliates.



Tip

You must rerun Import Applications when there are changes to your control budget dimensions or when you add new control budgets.

The control budget balances are stored in an Oracle Essbase cube. To obtain the cube name for a control budget, mouse over the control budget name in Manage Control Budget page. The name of the budgetary control cube will be used as the target mapping.



8. Click **OK**.

Integrating Planning and Budgetary Control Consumption Balances

You integrate the Planning application from the approved budget in Planning to the Budgetary Control. The integration enables you to simply pick the Planning application and specify the target Budget Control application and then set up the mappings between the applications.

To create the integration:

1. From the **Data Integration** home page, click .

   The General page is displayed in "Create Integration" view.

Copyright © 2017, 2026, Oracle and/or its affiliates.

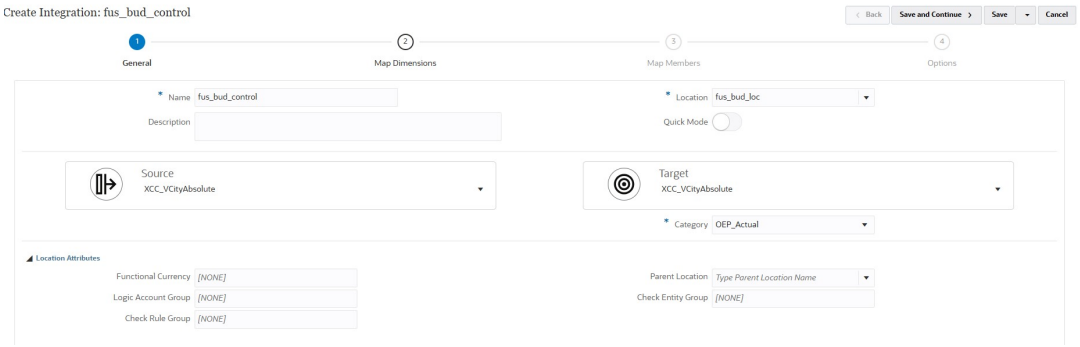
2. In **Name** and **Description**, enter a name and description to identify the integration when you launch the transfer.
3. In **Location**, enter a new location name, or pick an existing location to specify where to load data.
4. Click

   (Select a Source).
5. From the **Select a Source** page, select the Budget Control application.

   The Budget Control source system is registered in Data Integration and appears on the **Select a Source** page.

   Budgetary Control applications are prefixed with "XCC_" for example, **XCC_VCityAbsolute**.
6. Click (Select a Target).

   This is the target that was created in the prior step.
7. From the **Select a Target** page, select the Planning Budgetary Control application.



8. From **Location Attributes**, under **Functional Currency**, specify the currency used for the budget revision.

   For example, to specify the United States dollar, specify **USD**.
9. In **Category**, specify pre-defined scenario member

   For more information about the Category, see Defining Category Mappings.
10. Click **Save and Continue**.

    The next step is to map dimensions.

Defining Category Mappings

You create category mapping for scenario dimension members in the Planning and Planning Budget Revisions application to which Budgetary Controlbalances are loaded.

If you load Budgetary Control budget consumption balances for inquiry in Budget Revisions, load them to the pre-defined OEP_Consumed scenario member.

To define category mappings:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Category Mapping**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

2. On the **Category Mapping**, select **Global Mapping**.
3. Click (**Add**).

   A blank entry row is displayed.
4. In **Category**, enter a name that corresponds to the Planning and Planning Budget Revisions applications Scenario dimension member to which you want to load budget consumption amounts.

   For example, if you want to load the sum of commitments, obligations, other anticipated expenditures, and expenditures from Budgetary Control to one Scenario dimension member in the Planning application, you need one Category mapping entry.

   If you are using Budget Revisions feature, the system-generated Scenario dimension member for this usage is OEP_Consumed.

   If you are not using Budget Revisions and still want to load the Budgetary Control balances, instead of the encumbrance balances from General Ledger balances, to the Planning application, you can create a custom Scenario dimension member.

   Either way, create a category mapping entry and enter this Scenario dimension member name as the Target Category. Its corresponding Category can be named the same for convenience or anything you like, such as Budgetary Control Consumption or just Consumed.
5. Click **Save**.

Mapping Dimensions Between Budgetary Control and Planning

In **Map Dimension**, you specify an import format to set up the integration mapping between the Budgetary Control (Budget Chart of Accounts) segments and dimensions.

When you select the source and target, Data Integration populates the source and target columns automatically.

To map dimensions for a Budgetary Control based source:

1. From the **Data Integration** home page, click to the right of the integration, and then select **Map Dimensions**.
2. Go to the **Import Format Mapping** section.

   The target dimensions are populated automatically.
3. From **Source Column**, map the source dimensions from the source Budgetary Control application that corresponds to the dimensions in the target Planning application.

   If the import format has already been defined, then the source and target columns are mapped automatically.

   If you are adding a new import format or editing an existing import format, complete the following:

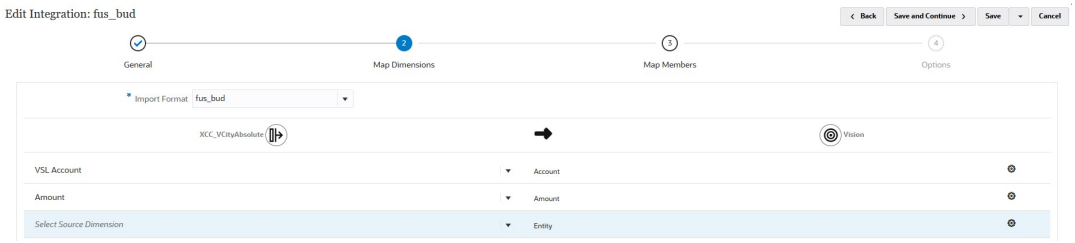
   - In the Source column drop-down, select the name of the source dimension to assign to the target application.

     Multiple source columns of the same dimension can be mapped to target dimensions. For example, you can map four "Account" source columns.
   - Add a source or target expression: assign an expression that operates on values directly from the source or target.

     See Using Source Expressions and Using Target Expressions.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> For Planning dimensions that cannot be mapped from a Budgetary Control dimension, such as the "Version" and "Plan Element" dimension in a Planning application, leave them unmapped. You can specify a single member for those unmapped Planning dimensions later.



4. Click **Save**.

Mapping Members Between Budgetary Control and Planning

When loading Budgetary Control budgets to Planning, you map the Budgetary Control source members to the Oracle Fusion Cloud Enterprise Performance Management target members.

Besides mapping the members for Planning dimensions mapped in the Import Format, also map the members for the unmapped Planning dimensions by specifying the Planning dimension member to which data is loaded, such as the "OEP_Working" version and the "OEP_Load" plan element.

To map members:

1. From the **Data Integration** home page, click to the right of the integration, and then select **Map Members**.
2. If you want to share the map from another integration, select the **Location Reference**.

   You are required to provide a Location reference if you define maps for the current integration.
3. From the **Dimensions** drop-down, select the source dimension to map.

   You must provide mapping for each target Planning dimension.

   For dimensions that are not mapped in the Import Format, you must map to a specific target member, such as "OEP_Working" in the unmapped "Version" dimension and "OEP_Load" in the unmapped "Plan Element" dimension in the Planning application.

   For dimensions that are mapped in Import Format, even if there is no update to the Planning dimensions value prior to the load, it is still necessary to create an "as is" mapping.
4. Select the **Like** tab.
5. In **Source Value**, specify the source dimension member to map to the target dimension member.

   To map all Budgetary Control accounts to Planning "as is" without any modification, in **Source Value**, enter ***, and from **Target Value**, enter ***.
6. In **Target Value**, select the member name to which the source members are mapped.

Copyright © 2017, 2026, Oracle and/or its affiliates.

You can also click the search to display the Member Selector and select a member name from the member list.

7. In **Rule Name**, enter the name of the data load rule used to transfer budget amounts to Budgetary Control.

> **Note**
>
> Rules are evaluated in rule name order, alphabetically. Explicit rules have no rule name. The hierarchy of evaluation is from **Explicit** to (In/Between/Multi) to **Like**.

8. In **Description**, enter a description of the mapping.

   For example, enter a description such as "Map to the ledger".

9. **Optional:** In **Apply to Rule**, check to apply the mapping only to the specific data rule in the location.

10. Click **Save**.

Adding Filters

Use a filter to limit the results from an Oracle ERP Cloud source.

Data Integration automatically creates filters when a integration is created. You can modify the filters as needed but cannot delete them. (If filters are deleted, Data Integration recreates the default value.)

Oracle ERP Cloud General Ledger Balance filters:

Oracle General Ledger Dimension	Filter
Scenario	Actual
Balance Amount	Ending Balance
Amount Type	YTD
Currency Type	Total
All Other Dimensions	'@ILvl0Descendants("All '|| TARGET_DIMENSION_NAME||' Values")'

> **Note**
>
> Drill Through is only supported if you load leaf level data for Oracle General Ledger Chart of Account segments. If you load summary level data, then drill through does not work.

> **Note**
>
> If you want to bring in encumbrance from Oracle General Ledger and combine it with Actual in Planning, modify the default dimension filter in the data load rule to include not only Actual but also Encumbrance.

To assign a filter to an integration:

Copyright © 2017, 2026, Oracle and/or its affiliates.

1. From the **Data Integration** home page, click to the right of the Planning integration, and then select **Options**.
2. Click the **Filter** tab.
3. Click .
4. From the **Dimension Name** drop-down, select the name of the dimension to which to add as a filter.

   Optionally, you can simply select another dimension already assigned to a filter and assign another dimension from the **Dimension Name** drop-down or just change the filter condition.
5. Click In **Filter Condition** and specify the filter condition:

   You can also click and select a member using the member selector from the **Select Members** page.

   For more information about selecting members, see Selecting Members from the Member Selector in *Smart View for Office User's Guide 25.100* .
6. Click **OK**.

Writing Back Cloud EPM Budget Balances to the Budgetary Control

If you want to report budget-to-actual from the General Ledger, you need to write back your budget to Oracle General Ledger. If you want to validate spending online, you need to write back your budget to Budgetary Control.

Use this procedure to write back original and revised budgets prepared using thePlanning feature to Oracle General Ledger.

This procedure is not for writing back budget revisions prepared using the Budget Revisions feature in the Oracle Fusion Cloud Enterprise Performance Management, which automatically updates budget in both General Ledger and EPM type control budget in Budgetary Control through other procedure. For more information, see Setting Up Budget Revisions and Integration with Budgetary Control.

This procedure synchronizes the budget written back to a Planning type control budget in Budgetary Control with the budget in Oracle General Ledger, making it possible to skip the Writing Back Budgets to the Oracle ERP Cloud in *Administering Data Management for Oracle Enterprise Performance Management Cloud* procedure for the portion of your enterprise-wide budget that you write back to Budgetary Control.

For more information, see Using Financials for the Public Sector.

Writing Back Cloud EPM Budget Balances to the Budgetary Control Process Description

At a high level, here are the steps for writing back Planning budgets to the Budgetary Control:

1. Register, configure, and initialize a source connection to the Budgetary Control.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> If you have already registered a source system to connect to the Budgetary Control application in Loading Budgetary Control to the Oracle Fusion Cloud EPM Process Description topic (see Loading Budgetary Control Budget Consumption Balances to Planning Process Description), you must reuse the same source system.

> **Note**
>
> Drill through is not supported in this release.

2. In Applications, select the source budget type as **EPM Financials module** for the Planning source application.

   For more information, see Defining Target Options for Budgetary Control Applications.

3. Create the integration between the Budgetary Control target application to which to write back budgets from the Planning source system.

   For more information, see Creating a Write Back Integration Definition for Budgetary Control Consumption and Planning Balances.

4. Map the dimensions between the Planning application and the Budgetary Control target application by building an import format.

   As a rule, when writing back to a Budgetary Control application, do not change, add, or delete any dimension details on the Map Dimensions page.

   For more information, see Mapping Dimensions Between Planning and Budgetary Control.

5. Map members to assign member mappings to the Control Budget.

   For example, mapping source budget member is just one of the dimensions that need to be mapped.

   For more information, see Mapping Members Between Planning and Budgetary Control

6. In **Options**, select any source and target options for the integration.

   For more information, see Adding Integration Options for Write Back.

7. Modify any filter values to limit any results.

   Data Integration automatically creates default filter values when an integration is created. You can modify the filter values as needed, but you cannot delete them (if you attempt to delete them.)

   For more information, see Adding Filters for the Write Back Integration.

8. Run the integration to execute the load and transfer commitment, obligation and expenditure amounts application to the Budgetary Control application from the Planning application.

   For more information, see Running an Integration.

> **Note**
>
> When running the integration, always choose **Replace** as the **Import Mode** for writing back to Budgetary Control.

Copyright © 2017, 2026, Oracle and/or its affiliates.

9. View the Oracle Fusion Cloud Enterprise Performance Management budgets loaded to Budgetary Control.

   See also Viewing the Cloud EPM Budgets Loaded to Budgetary Control.

10. Optionally, you can write out budget data from the Planning to a data file using the Data Export option. This output file may be used to load data to any other application.

    For more information, see: Creating a Data Export File Integration.

Creating a Write Back Integration Definition for Budgetary Control Consumption and Planning Balances

You create a write back integration definition to specify the Budgetary Control application and location to which the Planning consumption balances are written back.

The integration is created once, but used each time there is a transfer.

To create an integration:

1. From the **Data Integration** home page, click .

   The General page is displayed in "Create Integration" view.

2. In **Name** and **Description**, enter a name and description to identify the integration when you launch the transfer.

3. In **Location**, enter a new location name, or pick an existing location to specify where to load data.

4. Click

   (Select a Source).

5. From the **Select a Source** page, select the Budgetary Control target application to which to write back budgets from the Planning source system.

   Budgetary Control applications are prefixed with "XCC_" for example, **XCC_VCityAbsolute**.

6. Click (Select a Target).

7. From the **Select a Target** page, select the Planning Budgetary Control application.

8. From **Location Attributes**, under Functional Currency, specify the currency used for the budget revision. For example, to specify the United States dollar, specify USD.

9. In **Category**, leave the default category value.

   The categories listed are those that you created in the Data Integration setup.

10. Click **Save**.

Defining Category Mappings

You create category mapping for the scenario dimension member in Planning from which budget is written back.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> Budgetary Control does not have a "Scenario" dimension.

To create category mapping for scenario dimension members in the Planning application from which the budget is written back to the Budgetary Control.

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Category Mapping**.
2. On the **Category Mapping** page, select **Global Mapping**.
3. Click **Add**.

   A blank entry row is displayed.
4. In **Category**, enter a name that corresponds to the Scenario dimension member of the Planning application from which you want to write back the budget.

   For example, if you want to write back budget from the "Plan" Scenario dimension member in the Planning application, you may name the Category "Plan" by the same name as the Scenario dimension member. If the Scenario dimension member from which you want to write back the budget already exists as a Target Category in the existing system-generated category mappings, you do not need to create your own mapping.
5. In **Target Category**, enter the name of the Planning Scenario dimension members from which you want to write back the budget.
6. Click **Save**.

Mapping Dimensions Between Planning and Budgetary Control

In **Map Dimensions**, you specify an import format to set up the integration mapping between the Budgetary Control (Budget Chart of Accounts) segments and Planning dimensions.

When you select the source and target, Data Integration populates the source and target columns automatically.

To map dimensions for a Budgetary Control based source:

1. From the **Data Integration** home page, click to the right of the integration, and then select **Map Dimensions**.
2. Go to the **Import Format Mapping** section.

   The target dimensions are populated automatically.
3. From **Source Column**, map the source dimensions from the source Budgetary Control application that corresponds to the dimensions in the target Planning application.

   If the import format has already been defined, then the source and target columns are mapped automatically.

   If you are adding a new import format or editing an existing import format, complete the following:

   - In the Source column drop-down, select the name of the source dimension to assign to the target application.

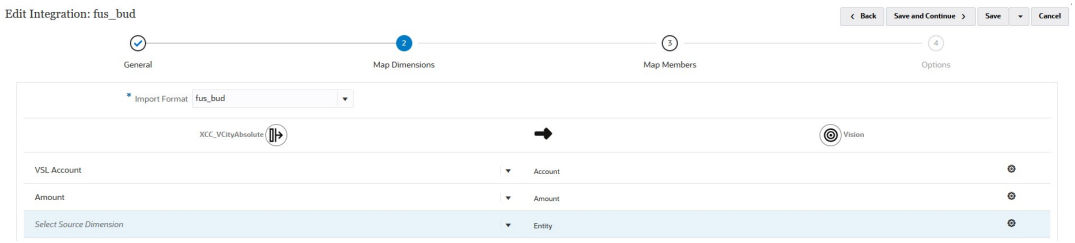
     Multiple source columns of the same dimension can be mapped to target dimensions. For example, you can map four "Account" source columns.

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Add a source or target expression: assign an expression that operates on values directly from the source or target.

  See Using Source Expressions and Using Target Expressions.

> **Note**
>
> For Planning dimensions that cannot be mapped from a Budgetary Control dimension, such as the "Version" and "Plan Element" dimension in a Planning application, leave them unmapped. You can specify a single member for those unmapped Planning dimensions later.



4. Click **Save**.

Mapping Members Between Planning and Budgetary Control

When writing back budgets to Budgetary Control, you assign member mappings to a Control Budget. For example, mapping source budget member is just one of the dimensions that need to be mapped. Member mappings convert data from the Planning dimension members to Budgetary Control during the transfer.

> **Note**
>
> Arbitrarily map the "Control Budget" Budgetary Control dimension to the "Account" Planning dimension. Without a mapping to the Control Budget, the import process fails.

> **Note**
>
> From **Source Column**, specify dimensions in the source Planning that correspond to the dimensions in the target Budgetary Controlapplication.
>
> The Source Column drop-down displays all Planning source system segments available for the Budgetary Control application.

To map members:

1. From the **Data Integration** home page, click to the right of the integration, and then select **Map Members**.
2. If you want to share the map from another integration, select the **Location Reference**.

   You are required to provide a Location reference if you define maps for the current integration.

Copyright © 2017, 2026, Oracle and/or its affiliates.

3. From the **Dimensions** drop-down, select the source dimension to map.

   You must provide mapping for each target Planning dimension.

   For dimensions that are not mapped in the Import Format, you must map to a specific target member, such as "OEP_Working" in the unmapped "Version" dimension and "OEP_Load" in the unmapped "Plan Element" dimension in the Planning application.

   For dimensions that are mapped in Import Format, even if there is no update to the Planning dimensions value prior to the load, it is still necessary to create an "as is" mapping.

4. Select the **Like** tab.

5. In **Source Value**, specify the source dimension member to map to the target dimension member.

   To map all Budgetary Control accounts to Planning "as is" without any modification, in **Source Value**, enter ***, and from **Target Value**, enter ***.

6. In **Target Value**, select the control budget name in Budgetary Control to which the budget is loaded.

   You can also click the search to display the Member Selector and select the control budget name from the member list.

7. In **Rule Name**, enter the name of the data load rule used to transfer budget amounts to Budgetary Control.

   > **Note**
   >
   > Rules are evaluated in rule name order, alphabetically. Explicit rules have no rule name. The hierarchy of evaluation is from Explicit to (In/Between/Multi) to Like.

8. In **Description**, enter a description of the mapping.

   For example, enter a description such as "Map to the ledger."

9. **Optional:** In **Apply to Rule**, select to apply the mapping only to the specific data rule in the location.

10. Click **Save**.

11. From the **Dimensions** drop-down, select the rest of the budget segments (budget dimension such as Fund, Department, Account and so on).

12. Use the **Like** mapping to map all Budgetary Control budget segment values from the correspondingPlanning dimension members "as is" without any modification, by entering * in Source Value and Target Value and specifying a Rule Name.

13. Click **Save**.

Adding Integration Options for Write Back

After you define member mappings for the location, select your options for budgets units in your source system to extract the data from the source Planning application and push it to Budgetary Control.

The integration is created once, but can be used each time for a transfer.

To select options for budget units in the source system

Copyright © 2017, 2026, Oracle and/or its affiliates.

1. From the **Data Integration** home page, click to the right of the integration, and then select **Options**.
2. From **General Options**, then **Category**, select a category.

   The categories listed are those that you created in the Data Integration setup, such as "Actual." See Managing Category Mappings.
3. From **Cube**, select the plan type of the target system.

   Cubes are assigned in Data Integration. You can register a custom cube as a separate application with no limit to the number of custom cube applications that can be registered.
4. In **Period Mapping Type**, select the period mapping type for each data rule.

   Valid options:

   - Default—The system uses the Period Key and Prior Period Key defined in Data Integrationto determine the Source General Ledger Periods mapped to each Data Integration period included in a Data Rule execution.
     Typically, choose the default period mapping type for writing back to Budgetary Control.
   - Explicit—The Data Rule uses the Explicit period mappings defined in Data Integration to determine the source General Ledger Periods mapped to each Data Integration Period included when running the integration. Explicit period mappings enable support of additional General Ledger data sources where periods are not defined by start and end dates.
5. In **Source Plan Type**, select the plan type of the Planning source from which to write back.
6. Click **Save**.

Adding Filters for the Write Back Integration

For data rules used to write back budget from a Planning application to the Budgetary Control, use filters to limit the results.

Data Integration automatically creates default filter values when a rule is created. You can modify the default filter values as needed but cannot delete them. (If the filters are deleted, Data Integration recreates the default value.)

When working with filters to write back budget from Planning to Budgetary Control, note the following:

- For Planning dimensions mapped to Budgetary Control dimensions that correspond to control budget segments such as "Account," specify members that correspond to budget segment values that are either at or below the control budget level as defined in Budgetary Control for the particular control budget to be written back. For ease of use, it is recommended that you select members from the level 0 descendants from these Planning dimensions and let Budgetary Control roll up to the appropriate budget level segment values during the write back process. Do not specify members from multiple levels.
- For the Planning dimensions that are not mapped to the Budgetary Control dimensions, such as "Version" and "Scenario," specify a single member, such as the "Final" version and "Plan" scenarios.

To assign a filter to a write-back integration:

1. From the **Data Integration** home page, click to the right of the Planning integration, and then select **Options**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

2. Click the **Filter** tab.
3. Click .
4. From the **Dimension Name** drop-down, select the name of the dimension to which to add as a filter.

   Optionally, you can simply select another dimension already assigned to a filter and assign another dimension from the **Dimension Name** drop-down or just change the filter condition.
5. Click In **Filter Condition** and specify the filter condition:

   You can also click and select a member using the member selector from the **Select Members** page.

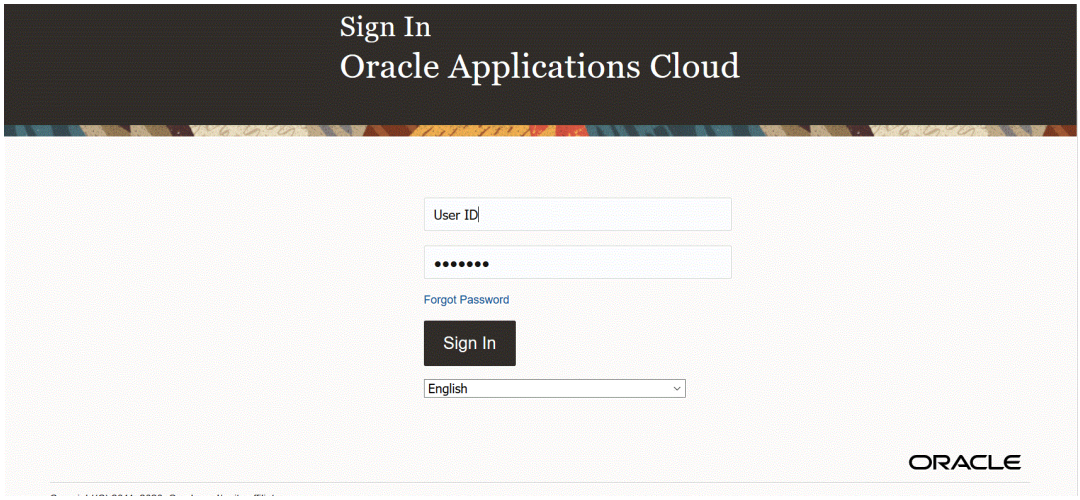
   For more information about selecting members, see Selecting Members from the Member Selector in *Smart View for Office User's Guide 25.100* .
6. Click **OK**.

Viewing the Cloud EPM Budgets Loaded to Budgetary Control

When you have successfully written back budgets from Oracle Fusion Cloud Enterprise Performance Management to the Budgetary Control, you can view the budget detail that was transferred.

To view the budget information transferred to Budgetary Control:

1. Sign into the **Oracle Applications Cloud**.



2. Enter your **User ID** and **Password**.
3. From the Oracle ERP Cloud Navigation menu, under **Budgetary Control**, select **Budgetary Control**.
4. From **Budgetary Control Workspace**, click the **Tasks** icon.
5. From the Tasks page, select **Review Budgetary Control Balances**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

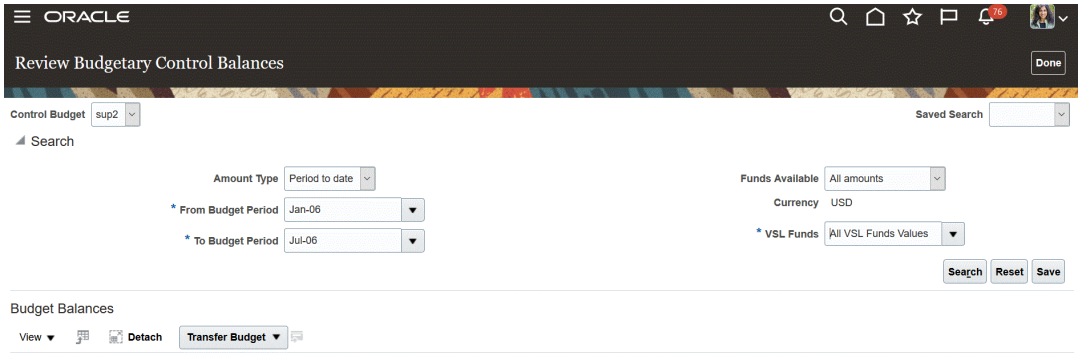
Budgets

- Manage Control Budgets
- Enter Budgets in Spreadsheet
- Review Budget Entries
- Review Budgetary Control Balances
- Review Budgetary Control Transactions

Period Close

- Budget Period Statuses
- Manage Encumbrance Carry Forward Rules
- Carry Forward Funds Available
- Carry Forward Purchase Order Budgetary Control Balances

6. On the **Review Budgetary Control Balances** page, select the **Control Budget**, and any search parameters for the budget that you want to review.



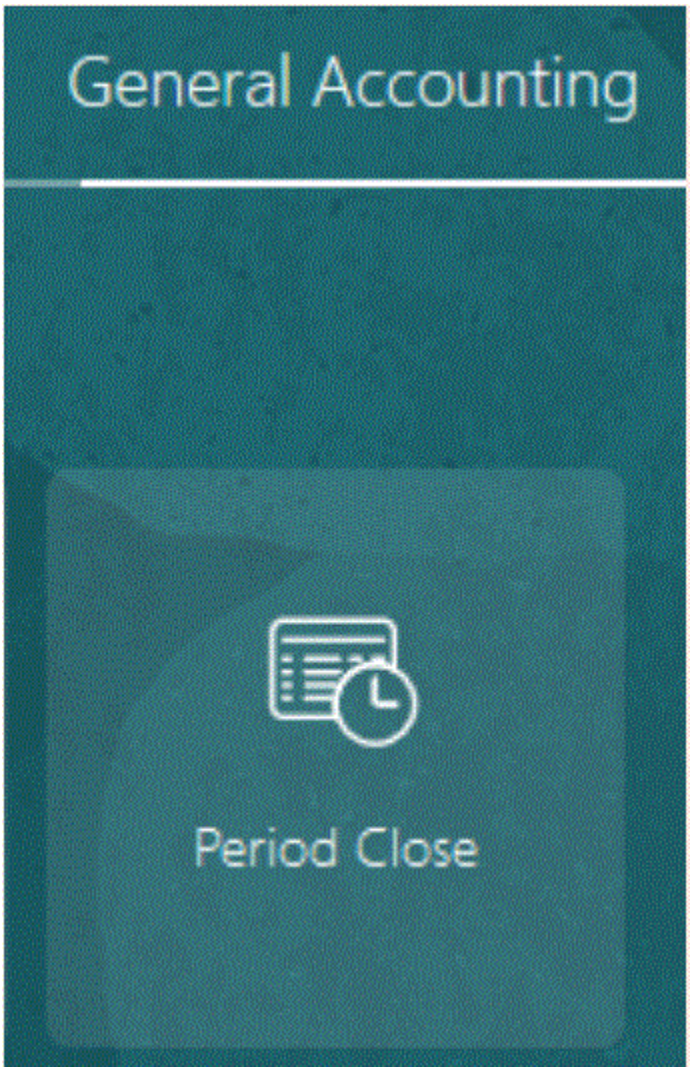
7. Click **Search**.

   The results of the search are shown on a results page.

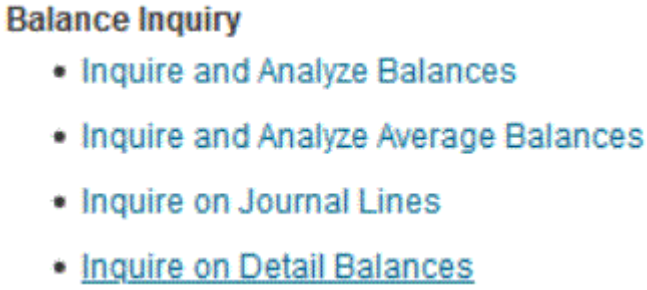
8. **Optional**: If you write back to a control budget with a source budget type classified as an "EPM Financials module,", then the system synchronizes the loaded budget to Oracle General Ledger for you, without having to perform the steps in the **Writing Back Budgets to the Oracle ERP Cloud** topic. You can verify the updated budget in Oracle General Ledger by completing the following:

9. From the Oracle ERP Cloud, from **General Accounting**, select **Period Close**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



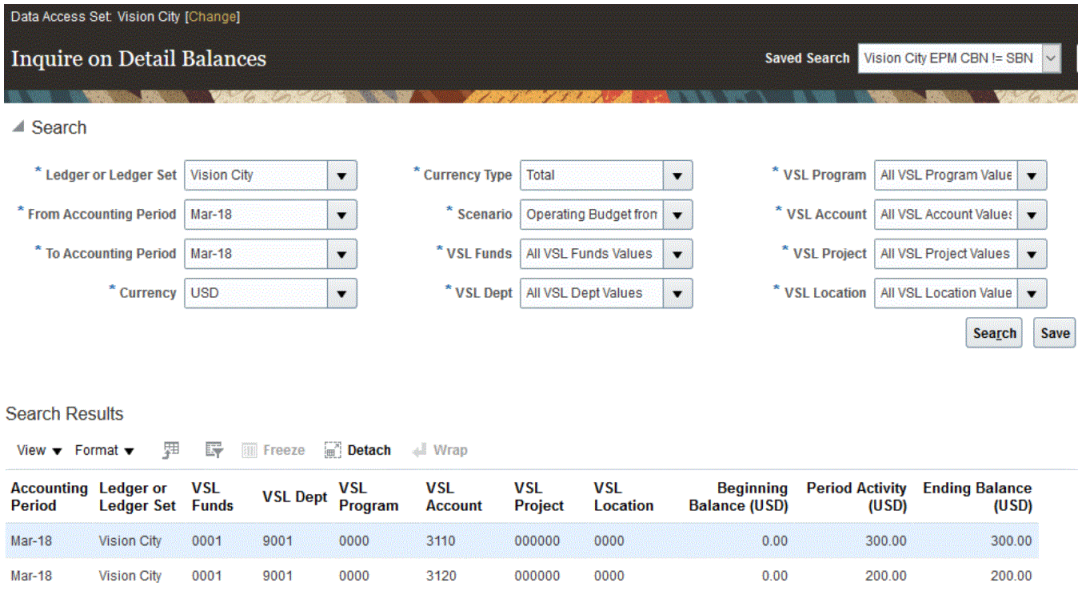
10. From **Period Close Workspace**, select **Tasks**.
11. From the **Tasks** drop-down, select **Inquire on Detail Balances**.



12. On the Inquire on Detail Balances page, select a data access set context if not already done and specify the search parameters for the budget that you want to review.

    Currently the Scenario for the budget auto-synchronized by the EPM Financials module type control budget goes by the same name as that of its source budget name.

Copyright © 2017, 2026, Oracle and/or its affiliates.



Integrating Budget Revisions with Budgetary Control

Write back your Oracle Fusion Cloud EPMbudget to Oracle General Ledger.

If you want to report budget-to-actual from General Ledger, you need to write back your budget to Oracle General Ledger. If you want to validate spending online, you need to write back your budget to Budgetary Control.

Use the procedure in this section to write back budget revisions prepared using the Budget Revisions feature in the Oracle Fusion Cloud Enterprise Performance Management to Budgetary Control, which automatically updates the budget in both General Ledger and EPM type control budget in Budgetary Control.

This procedure is not for writing back original and revised budget prepared using the Planning and Forecast features to the Oracle General Ledger and Budgetary Control.

Process Description

At a high level, these are the steps you complete in Data Integration to prepare to load budget revisions from EPM Planning Financials to Budgetary Control:

1. Complete the setup integration steps between Planning and Planning Budget Revisions and Budgetary Control.

   For more information, see https://docs.oracle.com/en/cloud/saas/planning-budgeting-cloud/epbca/fin_budget_adjustment_setup_102x2f7be273.html.

2. Register your application with **OEP_FS** as an input cube.

   This step enables you to register the Planning Budget Revision application, which stores the preparations or revisions made to the budget and then is loaded to Budgetary Control. This application name is appended with a "BAR" in the name, for example, "PRCVisionCityControlB_BAR"

3. Register **Budgetary Control** as a data source, specify the connection to use, and then import the Budgetary Control balance Essbase cubes into Data Integration.

Copyright © 2017, 2026, Oracle and/or its affiliates.

The Import Applications button brings over Budgetary Control balance Essbase cubes into Data Integration as Budgetary Control target applications. A control budget dimension member within each Budgetary Control target application represents a control budget in Budgetary Control used to load and write back data from and to the Oracle Fusion Cloud Enterprise Performance Management.

See Registering Budgetary Control as a Data Source and Importing Applications.

4. Register the budget adjustment revision by selecting **Data Export** category, then select the **Budget Adjustment Revision** type and then select the Budgetary Control Essbase cube for the application.

   This type of application is appended with a "BAR" in the application name, for example, "PRCVisionCityControlB_BAR."

   See Registering a Budget Adjustment Request.

5. Register the **Planning Budget Revision** as a data source, which stores the preparations or revisions made to the budget and then is loaded to Budgetary Control.

   See Registering the Planning Budget Revision.

6. **Optional:** If you migrate an application using Migration from another environment, then you must import the Control Budget information from the source. In this case, select the Budgetary Control Essbase application and select **Refresh Member** option from the action menu.

7. On the **General** page, create the integration between the **Planning Budget Revision** source application and the **Budget Adjustment Request**.

   See Integrating the Planning Budget Revision and Budget Adjustment Request.

8. On the **Map Dimensions** page, create an import format by mapping the dimensions from the **Planning Budget Revision** source to the **Budget Adjustment Request**.

   In this step you map the Planning Budget Revision dimensions to the Budget Adjustment Request dimensions.

   See Mapping the Planning Budget Revision and the Budget Adjustment Request Dimensions.

9. **Optional**: Map additional attribute columns to which you can import values from the selected EPM Planning Financials source.

   For more information, see Mapping Custom Budget Revision Attributes.

10. On the **Options** page, select the period mapping.

    Period mappings define the mapping between EPM Planning Financials and Budgetary Control. You can define period mappings in two ways:

    - Default Period Processing
    - Explicit Period Processing

    Select default period processing when EPM Planning Financials and Budgetary Control use consistent period definitions and period naming. For example, when a monthly calendar is used in both systems and the names of the periods like Jan-20 match both systems, then you use the Period Mapping Type of Default in the integration. No additional mapping is required.

    Use explicit period processing when EPM Planning Financials and Budgetary Control period definition or period naming conventions are different. In this case, you are required to select a calendar that is associated with an explicit source period mapping.

    See Defining Budget Adjustment Options.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Note the following about the EPM Planning Financials Budget Revisions to Budgetary Control integration:

- You **cannot** use **Member Mapping** for the EPM Planning Financials to Budgetary Control. You can only use an import expression to add a prefix or assign constant values for extra dimensions.
- You **cannot** execute the EPM Planning Financials to Budgetary Control integration from the Run Integration page in Data Integration. The run integration step is invoked only from EPM Planning Financials when performing a Funds Check or Funds Reservation action.
- You **cannot** view the results of the EPM Planning Financials to Budgetary Control integration in the Workbench.

Registering the Planning Application with OEP_FS as an Input Cube

To register a Planning application with OEP_FS as an input cube.

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, click (Add icon).
3. On the **Create Application** page, and then **Category**, select **EPM Local**.
4. From **Application**, select the Planning application associated with the OEP_FS cube.
5. From **Cubes**, select **Input Cubes - OEP_FS**.
6. **Optional**: In **Prefix**, specify a prefix to make the application name unique.

   The prefix is concatenated with the application name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.
7. Click **OK**.

Registering the Oracle ERP Cloud Source System

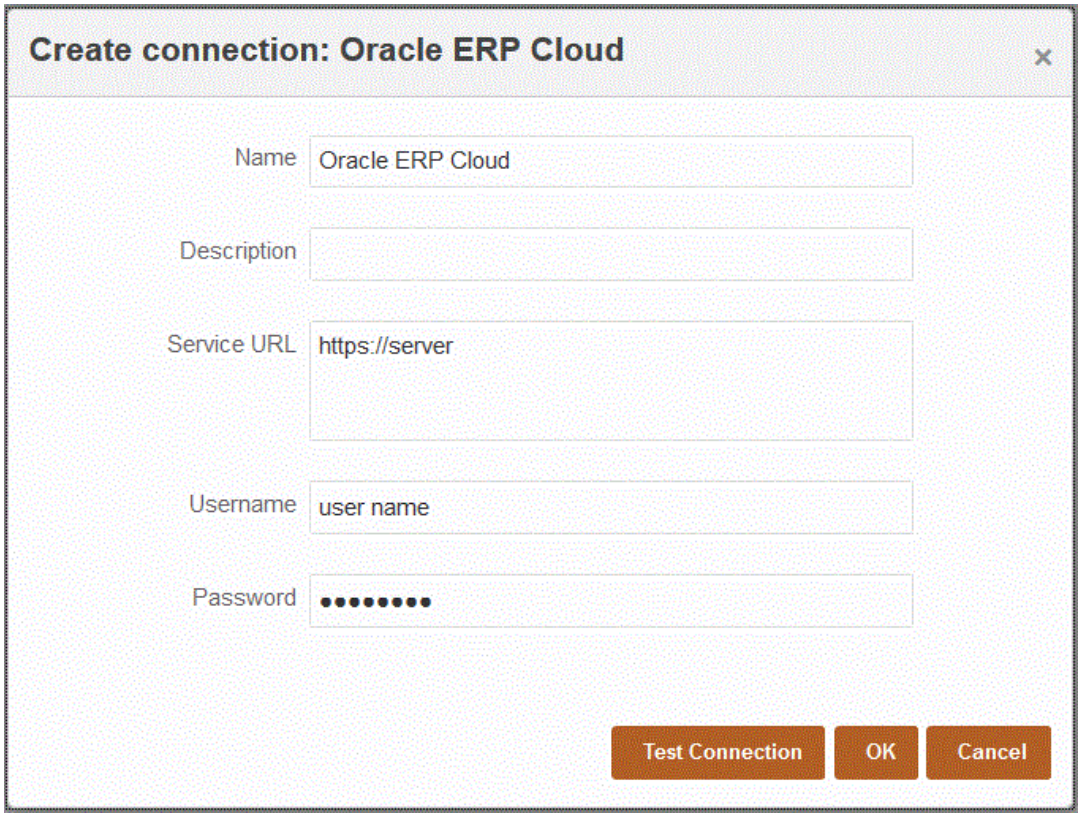
When integrating Planning Budget Revisions with the Budget Adjustment Request, first create a source system with the type: **Oracle ERP Cloud** and then specify the connection information.

To add Oracle ERP Cloud as a source system type:

1. From the Home page, click (**Application** cluster).
2. Click (**Data Exchange** icon), and then select the **Data Integration** tab.

   Optionally, you can launch Data Integration by clicking , and then from **Application**, select Data Exchange .
3. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
4. On the **Applications** page, click (Configure Connections icon).
5. Click (Add drop-down icon) drop-down, select **Oracle ERP Cloud**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



6. In **Name**, enter the source system name.
7. In **Description**, enter a description of the source system.
8. In **Service URL**, enter the server information for the web services.

   For example, enter: `https://server`.
9. In **Username**, enter the Budgetary Control user name.

   Enter the name of the Oracle ERP Cloud user who launches the process requests to send information between Oracle Enterprise Performance Management Cloud and the Oracle ERP Cloud. This user must have an assigned a "Budget Manager" job role.
10. In **Password**, enter the Oracle ERP Cloud password.

    You must update this password anytime you change your Oracle ERP Cloud password.
11. Click **Test Connection**.

    When the connection has been tested correctly, the information message "Connection to [*source system name*] successful" is displayed.
12. Click **OK**.

Registering Budgetary Control as a Data Source and Importing Applications

This step shows you how to register Budgetary Control as a data source and how to import the Budgetary Control applications.

To register Budgetary Control as a data source and import the Budgetary Control balance Essbase cubes into Data Integration:

1. From the Home page, click (**Application** cluster).

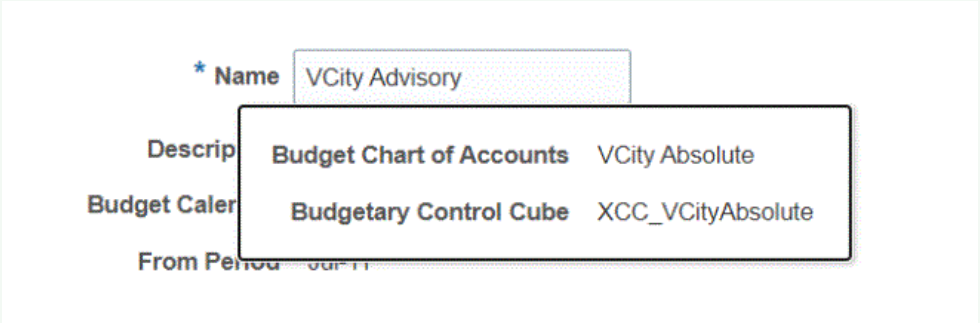
Copyright © 2017, 2026, Oracle and/or its affiliates.

2. Click (**Data Exchange** icon), and then select the **Data Integration** tab.
3. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
4. On the **Applications** page, click (Add icon).

> **Note**
>
> Make sure you click on the Application page and not the home page.

5. From **Category**, select **Data Source.**
6. From **Type**, select **Oracle ERP Cloud**.
7. From **Connection**, select the connection name used to connect to the Oracle ERP Cloud.
8. Enable **Budgetary Control**.
9. Click **Import Applications**.

   The Import Applications button brings over Budgetary Control balance Essbase cubes into Data Integration as Budgetary Control target applications. A control budget dimension member within each Budgetary Control target application represents a control budget in Budgetary Control used to load and write back data from and to the Oracle Fusion Cloud Enterprise Performance Management.

> **Tip**
>
> ou must rerun Import Applications when there are changes to your control budget dimensions or when you add new control budgets.
> The control budget balances are stored in an Oracle Essbase cube. To obtain the cube name for a control budget, mouse over the control budget name in Manage Control Budget page. The name of the budgetary control cube will be used as the target mapping.
>
> 
>

10. Click **OK**.

Registering the Planning Budget Revision

This step enables you to register the Planning Budget Revision from EPM Planning Financials.

To register the Planning Budget Revision application:

1. From the Home page, click (**Application** cluster).

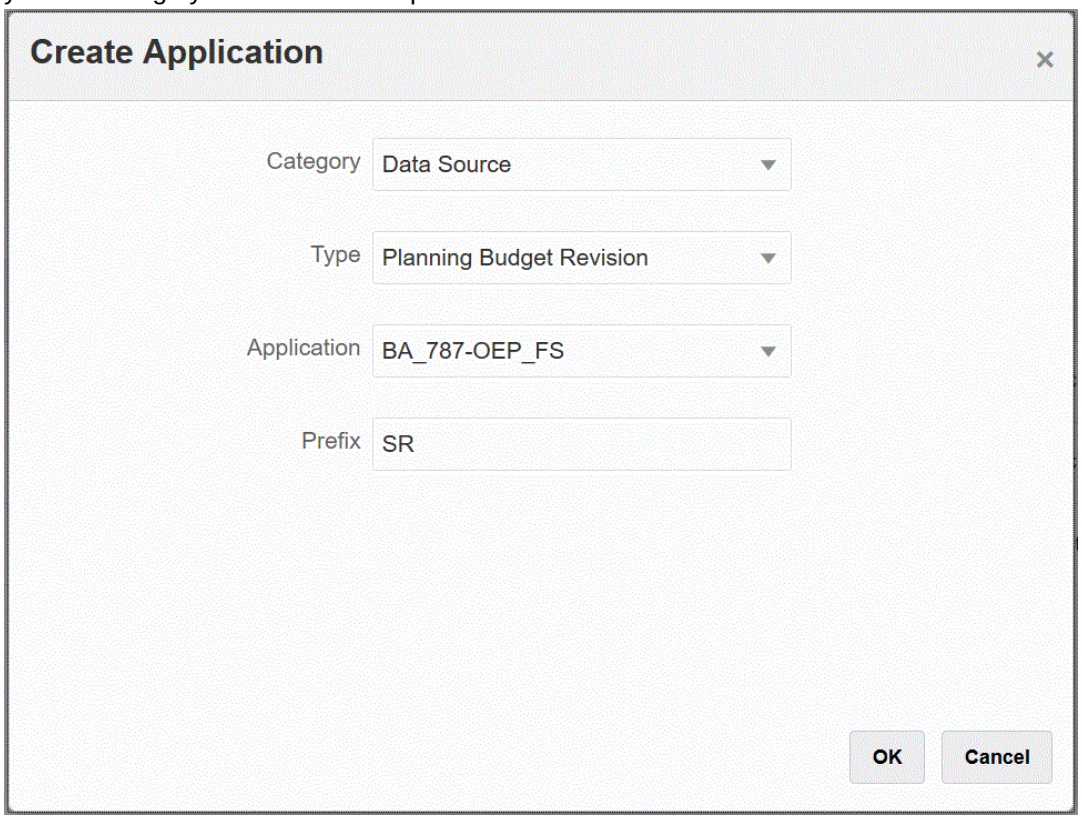
Copyright © 2017, 2026, Oracle and/or its affiliates.

2. Click (**Data Exchange** icon), and then select the **Data Integration** tab.
3. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
4. On the **Applications** page, click (Add icon).

> **Note**
>
> Make sure you click on the Application page and not the home page.

5. From **Category**, select **Data Source**.
6. From **Type**, select **Planning Budget Revision**.
7. From **Application**, select the Planning Input cube suffixed with **OEP_FS**.
8. **Optional**: In **Prefix**, specify a prefix to make the application name unique.

   The prefix is concatenated with the application name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.



9. Click **OK**.

Registering a Budget Adjustment Request

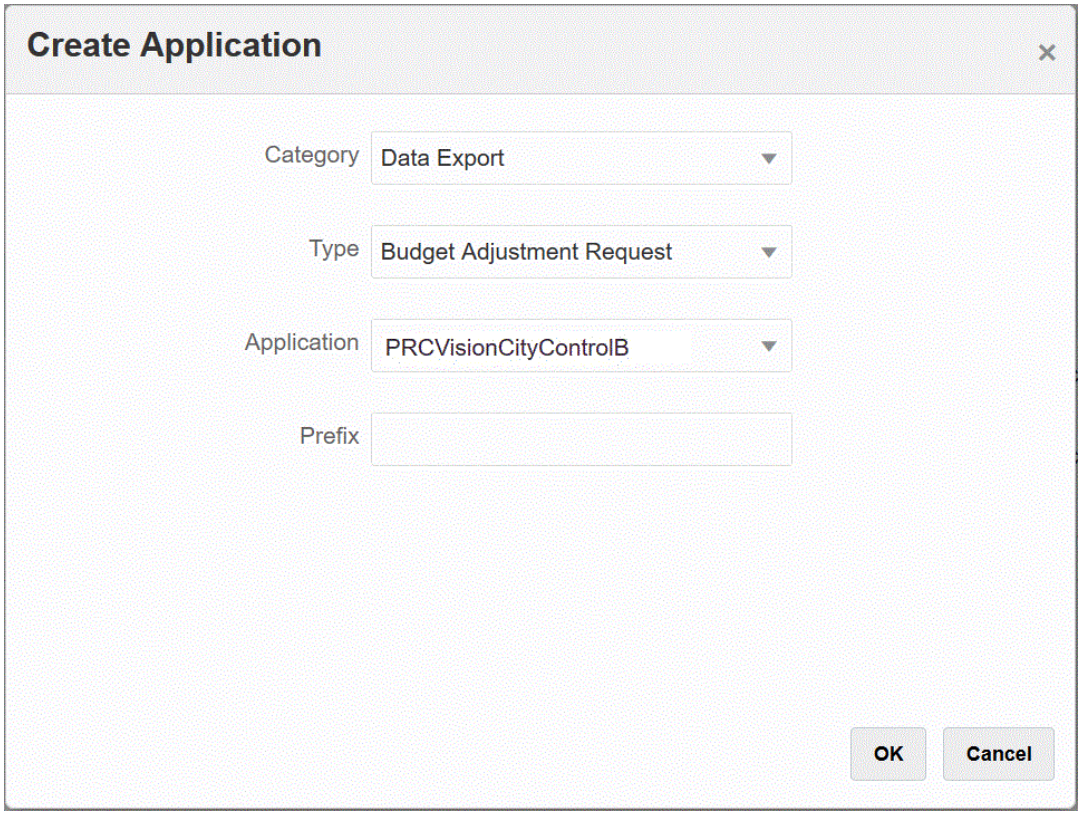
This step enables you to register the Budget Adjustment Request feature to which revisions from the approved Planning Budget Revision are loaded.

To register a Budget Adjustment Request:

Copyright © 2017, 2026, Oracle and/or its affiliates.

1. From the Home page, click (**Application** cluster).
2. Click (**Data Exchange** icon), and then select the **Data Integration** tab.
3. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
4. On the **Applications** page, click (Add icon).
5. On the **Create Application** page, and then **Category**, select **Data Export**.
6. From **Type**, select **Budget Adjustment Request**.
7. From **Application**, select the name of the Budgetary Control cube.
8. **Optional**: In **Prefix**, specify a prefix to make the application name unique.

   The prefix is concatenated with the application name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.



9. Click **OK**.

Integrating the Planning Budget Revision and Budget Adjustment Request

You integrate the Planning Budget revision from the approved budget in the EPM Planning Financials to the Budget Adjustment Request. The integration enables you to simply pick the Planning Budget Revision application and specify the target Budget Adjustment Request and then set up the mappings between the applications.

To create the integration:

1. From the **Data Integration** home page, click .

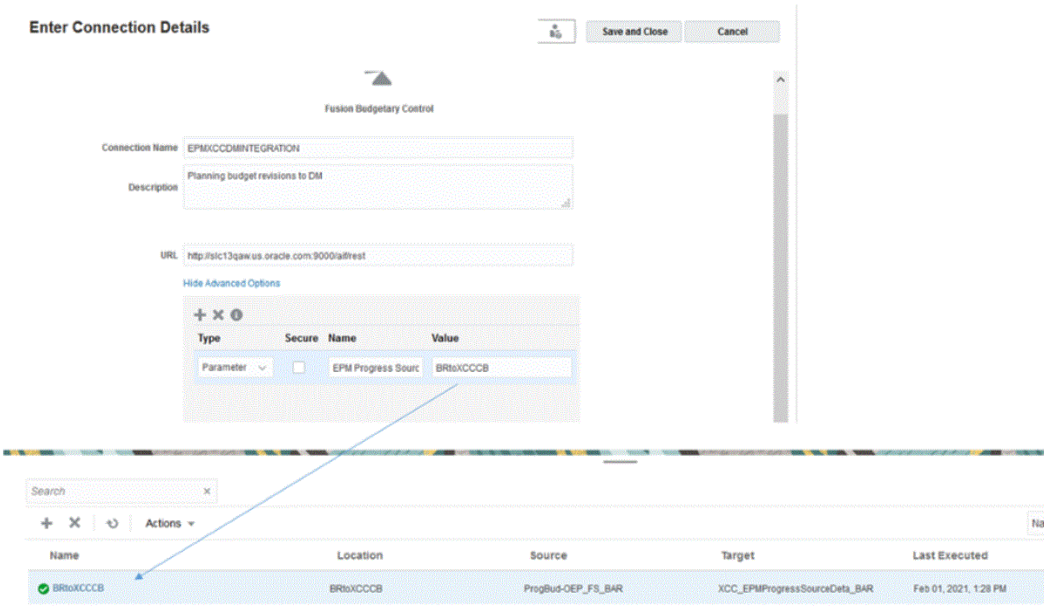
Copyright © 2017, 2026, Oracle and/or its affiliates.

The General page is displayed in "Create Integration" view.

2. In **Name** and **Description**, enter a name and description for the integration of the budget revision.

   The name of the integration is used in connections as a parameter in EPM Planning Financials as shown in the **Value** parameter below.

   When multiple control budgets share the same cube, the same mapping can be used to add additional parameters with the control budget name.



   For more information, see **Setting Up Integration Between EPM Planning Financials and Budgetary Control**.

3. In **Location**, enter a new location name, or pick an existing location to specify where to load data.

4. Click

   (Select a Source).

5. From the **Select a Source** page, select the Planning Budget Revision application.

   The name of the Planning Budget Revision application is suffixed with a "_BAR" in the application name, for example, "OEP_FS_BAR."

   The Planning Budget Revision source system is registered in Data Integration and appears on the **Select a Source** page.

6. Click (Select a Target).

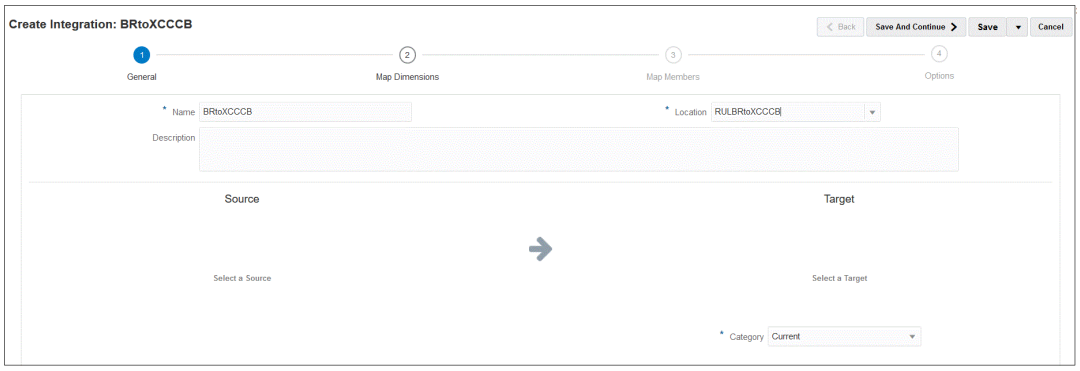
   This is the target that was created in the prior step.

7. From the **Select a Target** page, select the Budgetary Control cube application.

   Target applications are displayed on the **Select a Target** page.

   The budgetary control is suffixed with "BAR."

Copyright © 2017, 2026, Oracle and/or its affiliates.



8. From **Location Attributes**, under **Functional Currency**, specify the currency used for the budget revision.

   For example, to specify the United States dollar, specify **USD**.

9. Click **Save and Continue**.

   The next step is to map dimensions.

Mapping the Planning Budget Revision and the Budget Adjustment Request Dimensions

As part of the integration between the Planning Budget Revision and the Budget Adjustment Request, you map EPM Planning Financials dimensions to Budgetary Control target dimensions. You can also customize and extend these integrations, for example, by applying expressions as needed to meet your business requirements.

To map dimensions:

1. From the **Data Integration** home page, click to the right of the Planning Budget Revision and the Budget Adjustment Request integration, and then select **Map Dimensions**.
2. From **Import Format**, select the name of the import format to use for the integration.

   You can also add a user defined import format name.
3. From **Type**, select the **Delimited - All Data Type** and then click **Save** to refresh the rows.
4. In the mappings grid, map the source columns in the source data-load file to the dimensions in the target application by completing the following:
   a. In **Select Source Dimension**, specify the name of the Planning Budget Revision source dimension to assign to the Budget Adjustment Request target dimension.
   b. Map **Plan Element** to **Plan Element**, **Period** to **Plan Period** and **Year** to **Plan Year**.
   c. Add a target dimension expression for each of the Budgetary Control dimensions. Typical expressions for the most common use cases are:
      - Copysource—use the same member name from EPM Planning Financials in Budgetary Control.
      - LTrim—remove the prefix used in EPM Planning, Financials.
      - Constant—Assign a constant value for an extra dimension in Budgetary Control.

   This is a required step - sources must be mapped to the target and target expression added.

Copyright © 2017, 2026, Oracle and/or its affiliates.

5. Click **Save**.

Mapping Custom Budget Revision Attributes

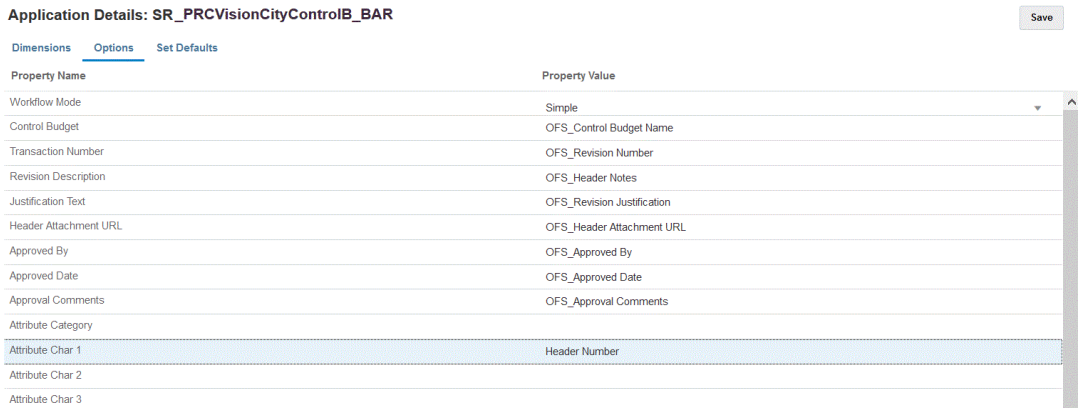
You can map additional attribute columns to which you can import values from the selected EPM Planning Financials source. You can add the value during processing by entering the value for the attribute in the expression field.

To map custom Budget Adjustment Revision attributes:

1. From the Home page, click (**Application** cluster).
2. Click (**Data Exchange** icon), and then select the **Data Integration** tab.
3. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
4. On the **Applications** page, click to the right of the Budget Adjustment Revision application, and then select **Application Details**.

   The Budget Adjustment Revision application is associated with a **Data Export** category.
5. In **Attribute Char 1 -10**, map any additional attributes and click **Save**.

   In the following example, **Header Number** has been added to the **Attribute Char 1** value.



Defining Budget Adjustment Options

You use options to define the type of period mappings used in the Planning Budget Revision and the Budget Adjustment Request integration. You can use either default period mappings or explicit period mappings. Note that you can only select the period mapping type in Options. All other options are preselected and cannot be changed.

> **Tip**
>
> Review the date mapping align with your control budget periods and fiscal year, you may need to modify the mapping created automatically.

To specify options:

Copyright © 2017, 2026, Oracle and/or its affiliates.

1. From the **Data Integration** home page, click to the right of the integration, and then select **Options**.
2. From **Period Mapping Type**, select the mapping type of the period.

   Available options:

   - Default—The integration uses the Period Key and Prior Period Key defined in Data Management to determine the source General Ledger periods mapped to each Data Management period when the integration is run.

     Default period mappings between the Planning Budget Revision and the Budget Adjustment Request are set up automatically during the application creation.

   - Explicit—The integration uses Explicit period mappings defined in Data Management when Explicit period mapping can be used when the EPM Planning Financials periods and Budgetary Control periods are not of same type. For example, EPM Planning Financials is monthly and Budgetary Control is quarterly.

     Explicit mappings are set up using calendars periods in the source system option. You select the actual calendar to use for the integration from the Calendar drop-down. For more information, see Applying Source Mappings.
3. From **Calendar**, select the name of the source period mapping calendar used for integrating data between Planning Budget Revision and the Budget Adjustment Request.
4. Click **Save**.

Integrating Oracle NetSuite

Load data from Oracle NetSuiteto your Oracle Fusion Cloud Enterprise Performance Management business process.

Data Integration is available as the integration mechanism for Cloud EPM customers looking to load data from Oracle NetSuite.

The integration uses the NetSuite EPM Connector SuiteApp for new users and NSPB Sync for existing users to load data from Oracle NetSuite.

Customers who have been using the NSPB Sync to load data from Oracle NetSuite to their application in the Cloud EPM, should continue to use NSPB Sync to do so. For more information about the NSPB Sync SuiteApp, see NSPB Sync SuiteApp.

If you are not a NSPB Sync customer (including existing Oracle NetSuite Account Reconciliation (NSAR) customers), and you want to connect to the Oracle NetSuite, you must use the NetSuite EPM Connector SuiteApp. This connector facilitates the connection between Oracle NetSuite and applications in the Cloud EPM.

For more information, see NetSuite EPM Connector SuiteApp.

The "Account Reconciliation Sync SuiteApp", the "Planning and Budgeting Sync "and "Close Management and Consolidation Sync" provide additional functionality beyond the NetSuite EPM Connector SuiteApp that includes loading metadata with hierarchies, write-back from Oracle Fusion Cloud EPM to NetSuite, predefined saved searches to speed up implementation, ability to run EPM job from NetSuite and loading third party data into Cloud EPM via SFTP.

- For more information about the Planning and Budgeting Sync SuiteApp, see: Planning and Budgeting Sync SuiteApp.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> For information on sending Planningback to Oracle NetSuite, see .Setting Up Your Planning and Budgeting for Budget Data Import

- For more information about the Account Reconciliation Sync SuiteApp, see: Account Reconciliation Sync SuiteApp.
- For more information about the Financial Consolidation and Close Sync SuiteApp, see: Close Management and Consolidation Sync SuiteApp.

Process Description for Integrating Oracle NetSuite

When Oracle NetSuite is used as a data source in the Oracle Fusion Cloud Enterprise Performance Management, Data Integration uses the NSPB Sync or EPM Connector, along with Saved Search or data sets to query data from Oracle NetSuite.

> **Note**
>
> Data generated from the Saved Search is used for importing data only and not for write-back.
> To write back to Oracle NetSuite, check the following link on how to configure data push back from Planning into Oracle NetSuite at: Setting Up Your Planning and Budgeting for Budget Data Import. You can use Account Reconciliation Sync to load data back into Oracle NetSuite.

At a high level, these are the steps for loading data from an Oracle NetSuite data source:

1. To install the SuiteApp, an administrator must install the "EPM Connector Foundation" bundle first. For NSAR customers, the EPM Connector Foundation, the bundle is provisioned automatically.

   Other customers need to work with their account manager to submit a bundle provisioning request for approval. When it is approved, customers can install the "EPM Connector Foundation" bundle.

2. A Oracle NetSuite service administrator installs the NSPB Sync SuiteApp, EPM Connector SuiteApp, Planning and Budgeting Financials Sync SuiteApp, or Account Reconciliation Sync SuiteApp by performing the installation and setup tasks in Oracle NetSuite.

   For detailed information on installing and setting up your SuiteApp, see the table below:

Table 17-1 Netsuite SuiteApp Installation and Setup References

SuiteApp	Reference
Oracle EPM Connector	Oracle EPM Connector Installation and Setup
NSPB Sync	NSPB Sync Setup Guide
Planning and Budgeting Financials	Integration Guidelines
Account Reconciliation Sync	Account Reconciliation Sync SuiteApp Installation and Setup

3. In **Applications**, register the Oracle NetSuite application as a data source, and select the **Import Application** option to import the definition for all saved searches.

Copyright © 2017, 2026, Oracle and/or its affiliates.

When you select this option, Data Integration imports all saved search definitions owned by the user or datasets that are configured to be loaded from Oracle NetSuite. If you don't want to import all saved search definitions, go to target application and select individual saved search definitions one by one. If you have initialized the source system the first time, add incremental saved search definitions also in the target application.

For more information, see: Creating an Oracle NetSuite Data Source and Registering Oracle NetSuite Applications.

For information on updating an Oracle NetSuite application, see Updating An Oracle NetSuite Application.

4. **Optional**: Map columns in the saved search to dimensions.
5. Define the Cloud EPM application as the target application.

   For more information, see: Registering Cloud EPM Applications.
6. Define the import format to map columns from the saved search to dimensions in the target Cloud EPM application dimensions.

   For more information, see: Mapping Dimensions.
7. Define a drill URL specifying the Oracle NetSuite.

   For more information, see: Drilling Through to Oracle NetSuite.
8. Once the initialization process is complete, you can pick a Saved Search when adding a target application. When you select Oracle NetSuite or a dataset as a data source, then you are presented with a list of the saved searches from the selected Oracle NetSuite source.

   You can also provide source filter criteria on the application filters tab. These source filters are the same as Oracle NetSuite "Criteria," which filter the data from Oracle NetSuite Saved Searches.
9. Define source mapping entries in the calendar mapping section to map the Oracle NetSuite periods to the Cloud EPM periods.
10. Create an integration and specify the Oracle NetSuite data source as the source application and your Cloud EPM application as the target application.

    For more information, see: Creating Direct Integrations.
11. Define any period mapping. Available options are explicit or default period mappings.

    For more information on periods mappings available for an Oracle NetSuite integration, see: Managing Periods in Oracle NetSuite.
12. Map members mappings between the Oracle NetSuite data source and the Cloud EPM target application.
13. Apply any Oracle NetSuite filters and options.

    For more information, see: Applying Oracle NetSuite Application Filters.
14. Run the integration between Oracle NetSuite data source and the Cloud EPM target application.

    This pulls the data from the Oracle NetSuite instance into Data Integration, maps the data and then shows the results in the workbench. If the mapping succeeds without errors, the data is loaded to the target application.

    For more information, see: Running an Integration.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Creating an Oracle NetSuite Data Source

Target applications enable Data Integration to store the structure of the source and target applications that can be integrated. In this way, you can map the source and target applications and specify import criteria.

Oracle NetSuite Saved Searches results are registered with an application type of "data source." During the initialization, the system creates the data source automatically based on the available Saved Searches results under the user name registered in the Oracle NetSuite source system. In the target application option, you can create additional Saved Searches in Oracle NetSuite, and then register the custom saved searches on the Application page.

> **Note**
>
> You can click **Update Application** from the **Actions** menu on the Application page to update any saved searches that have been created in Oracle NetSuite after you have initialized the source system in Data Integration.

> **Note**
>
> When you create an Oracle NetSuite data source, dimension details are populated automatically and mapped directly to the target dimension class "Generic." As a rule, when loading data from an Oracle NetSuite data source, do not change, add, or delete any dimension details.

For information on creating an Oracle NetSuite data source, see Registering Oracle NetSuite Applications.

Mapping Columns in the Saved Search to Dimensions

You can manage the dimensions associated with the Oracle NetSuite application. A dimension is a structure that categorizes data in order to enable users to answer business questions. Each dimension usually contains a hierarchy of related members grouped within it. Commonly used dimensions are customers, products, and time.

To map columns in the Saved Search:

1. From the **Application** page, click to the right of the Oracle NetSuite application, and then select **Application Details**.
2. Select the **Dimensions** tab.
3. From **Dimension Name**, select the dimension name of the column to which to classify the dimension.
4. Based on the column type, from the corresponding **Dimension Classification** drop-down, select the classification of the column.
5. Click **Save**.

   For more information on updating a saved search and as well as using SuiteAnswers, see https://www.netsuite.com/portal/landing/suiteanswers.shtml

Copyright © 2017, 2026, Oracle and/or its affiliates.

Adding a Target Dimension for Oracle NetSuite Data Sources

A Oracle NetSuite Saved Search is stored as a data source, and import formats enables you to set up the integration mapping between the data source and the dimensions of your target application in Oracle Fusion Cloud Enterprise Performance Management.

The files generated by the data source application can be in any of the supported format like single column numeric, and multi-column data.

When you select an Oracle NetSuite data source, Data Integration populates the source and target columns automatically.

You map Oracle NetSuite Saved Search data source columns or fields in the import file to dimensions in the Cloud EPM application.

To add a target dimension:

1. From the **Data Integration** home page, click ... to the right of the integration, and then select **Map Dimensions**.
2. On the **Map Dimensions** page, from **Import Format**, select the name of the import format to use for the integration.
3. In **File Type**, select the format of the file to import:

   Available file formats:

   - Delimited - Numeric Data
   - Multi Column - Numeric Data
   - Delimited Data - All Data Type
   - Multi Column - All Data Type
4. In **Drill Through URL**, enter the search type criteria used for the drill-through.
5. In the Mapping Dimension section, map the source columns to the target dimensions by selecting the source column and then selecting target dimension from the target dimension drop-down.
6. To map additional dimensions, in the Mapping section, click , then from the **Add Dimension Row** drop-down, select the dimension row to add.
7. **Optional**: In **Expression**, add any import expressions.

   Data Integration provides a set of powerful import expressions that enable it to read and parse virtually any file into the Data Integration database. You enter advanced expressions in the Expression column of the field. Import expressions operate on the value that is read from the import file.
8. Click **Save**.

Defining Oracle NetSuite Application Options

You can define specific Oracle NetSuite application property options.

To specify application properties:

Copyright © 2017, 2026, Oracle and/or its affiliates.

1. From the **Application** page, click ⋯ to the right of the Oracle NetSuite application, and then select **Application Details**.
2. From **Property Name**, select the value to add or change in the **Property Value** drop-down.

   Available properties:

   - **Postingperiod**—Select the posting period from which to load data.

     > **Note**
     >
     > If you need to load data by a specific period, for example, by a specific month, use a posting period filter that contains a period ID. The period ID is created in the posting period filter and returned by the Saved Search used for filtering data.
     > Oracle NetSuite provides several types of date filters: a named time period, for example, last fiscal year, a custom date range defined by a specific start date and end date, and a relative date range defined by a beginning number of days, weeks, months, quarters, or years ago, or from now, to an ending number of same. See the NetSuite "Search Guide" for more information on setting up filters.

   - **Mainline**—Select **True** to load the **Primary Information** field group in a transaction, as opposed to line item data entered on the transaction's tabs. Otherwise, select **False**.
   - **Posting**
3. Click **Save**.

Applying Oracle NetSuite Application Filters

Search criteria defined in the Oracle NetSuite Saved Searches are registered automatically as application filters. You can edit the filters if needed. For example, you can assign default values or hide the filters from end users by setting the property level.

You can apply filter conditions to the Oracle NetSuite Saved Search results so that only those records that meet selected conditions are returned to Data Integration. You can specify a single filter condition or multiple filters conditions, and additionally specify the exact values that you want returned.

To apply an Oracle NetSuite filter condition:

1. From the **Data Integration** home page, click ⋯ to the right of the Oracle NetSuite integration, and then select **Options**.
2. Click the **Filter** tab.
3. Select the name of the filter condition.
4. From the **Condition** drop-down, select the applicable condition.

   The applicable conditions by filter are derived from the Oracle NetSuite Saved Search.
5. From the **Value** drop-down, select the value to which to apply the filter.
6. Click **Save**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Adding Additional Filters to the Drill URL in the Import Format

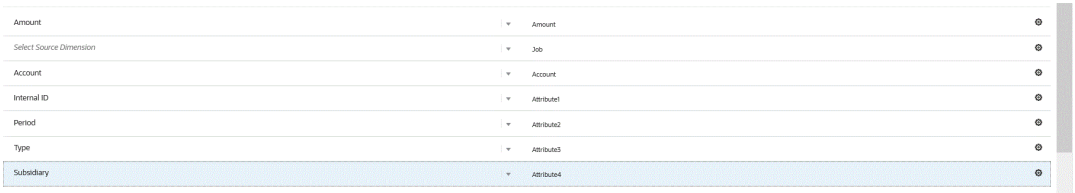
When performing a drill through from Oracle Fusion Cloud Enterprise Performance Management to Data Integration, and then to the Oracle NetSuite Saved Search, sometimes too many records are returned. As a guideline, when you add the amounts in all the records in the drill through, they should total to the number in the data cell you were on when you initiated the drill through in Cloud EPM. To resolve this issue, add additional filters to the drill through URL in the Import Format.

To add additional filters:

1. From the **Data Integration** home page, click to the right of the integration, and then select **Map Dimensions**.
2. On the **Map Dimensions** page, from **Import Format**, select the name of the import format to use for the integration.

   You can also add a user defined import format name.
3. In the **Mapping** section, select the source dimension column to use as a filter and map it to a target **Attribute** column

   For example, you could map a Subsidiary ID source column to the Attribute 4 row.



4. In **Drill Through URL**, click and enter the search type criteria used for the drill-through.

   This is the drill through format:

   ```
   searchtype=Transaction&searchid=customsearch_nspbcs_all_transactions_s_5&Transaction_TYPE&detailname=$ATTR3$&Transaction_ACCOUNT=$ATTR1$&Transaction_POSTINGPERIOD=$ATTR2$&Transaction_POSTING=T&Transaction_MAINLINE=F&
   ```

   If you want to add an additional filter, such as Subsidiary ID, enter **&Transaction_SUBSIDIARY=$ATTR4$** to the list of parameters.

   In this case, specify the entire drill through URL definition as:

   ```
   Searchtype=Transaction&searchid=customsearch_nspbcs_trial_balance&Transaction_ACCOUNT=$ATTR1$&Transaction_POSTINGPERIOD=$ATTR2$&Transaction_SUBSIDIARY=$ATTR4$&Transaction_POSTING=T&
   ```

5. Click **Save**.

Managing Periods in Oracle NetSuite

When loading periods from Oracle NetSuite, you have two options:

- Default Period Mappings—Use this option to load a single period from the POV from the data load rule. In this case, you do not map the period column in the import format or define a source period mapping in period mappings. You can set a source filter in the data rule such as the Current Period or Last period. For example, if you want to load the Feb

Copyright © 2017, 2026, Oracle and/or its affiliates.

2018 period, you set it as the last period. When you run the data rule, periods are not checked and all the data you extract from Oracle NetSuite is imported into the selected POV.

- Explicit Period Mappings—Explicit period mappings enable the loading of General Ledger data when Oracle NetSuite and Planning periods match. Use this option to load multiple periods, or when you do not want users to select the period from the POV.

  To use explicit period mappings, you map the period column in the import format and define source period mappings. The General Ledger period column must match exactly the Oracle NetSuite period. Next in the integration, you select the calendar from the period mapping. When you run the data rule, you specify the start – end periods that you want to load. You can set a filter condition in the data rule to extract specific period data, for example, the Fiscal Year Current to Fiscal Quarter Current period. Data Integration matches the period column from the search result with the period in the period mapping and loads the data to the appropriate period.

Adding a Target Dimension for Oracle NetSuite Data Sources

A Oracle NetSuite Saved Search is stored as a data source, and import formats enables you to set up the integration mapping between the data source and the dimensions of your target application in Oracle Fusion Cloud Enterprise Performance Management.

The files generated by the data source application can be in any of the supported format like single column numeric, and multi-column data.

When you select an Oracle NetSuite data source, Data Integration populates the source and target columns automatically. You map Oracle NetSuite Saved Search data source columns or fields in the import file to dimensions in the Cloud EPM application.

To add an import format to an Oracle NetSuite source:

1. From the **Data Integration** home page, click to the right of the integration, and then select **Map Dimensions**.
2. On the **Map Dimensions** page, from **Import Format**, select the name of the import format to use for the integration.
3. In **File Type**, select the format of the file to import:

   Available file formats:

   - Delimited - Numeric Data
   - Multi Column - Numeric Data
   - Delimited Data - All Data Type
   - Multi Column - All Data Type
4. In **Drill Through URL**, enter the search type criteria used for the drill-through.
5. In the Mapping Dimension section, map the source columns to the target dimensions by selecting the source column and then selecting target dimension from the target dimension drop-down.
6. To map additional dimensions, in the Mapping section, click , then from the **Add Dimension Row** drop-down, select the dimension row to add.
7. **Optional**: In **Expression**, add any import expressions.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Data Integration provides a set of powerful import expressions that enable it to read and parse virtually any file into the Data Integration database. You enter advanced expressions in the Expression column of the field. Import expressions operate on the value that is read from the import file.

For more information, see .

8. Click **Save**.

Adding Additional Filters to the Drill URL in the Import Format

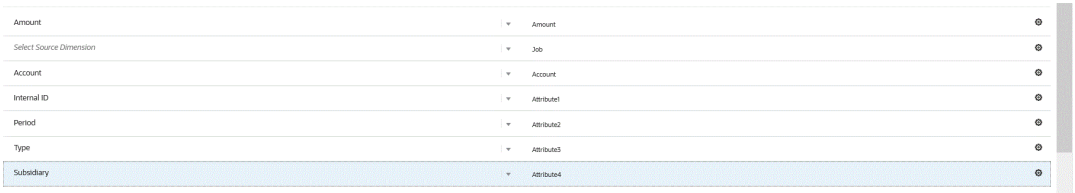
When performing a drill through from Oracle Fusion Cloud Enterprise Performance Management to Data Integration, and then to the Oracle NetSuite Saved Search, sometimes too many records are returned. As a guideline, when you add the amounts in all the records in the drill through, they should total to the number in the data cell you were on when you initiated the drill through in Cloud EPM. To resolve this issue, add additional filters to the drill through URL in the Import Format.

To add additional filters:

1. From the **Data Integration** home page, click to the right of the integration, and then select **Map Dimensions**.
2. On the **Map Dimensions** page, from **Import Format**, select the name of the import format to use for the integration.

   You can also add a user defined import format name.
3. In the **Mapping** section, select the source dimension column to use as a filter and map it to a target **Attribute** column

   For example, you could map a Subsidiary ID source column to the Attribute 4 row.



4. In **Drill Through URL**, click and enter the search type criteria used for the drill-through.

   This is the drill through format:

   ```
   searchtype=Transaction&searchid=customsearch_nspbcs_all_transactions_s_5&Transaction_TYPE&detailname=$ATTR3$&Transaction_ACCOUNT=$ATTR1$&Transaction_POSTINGPERIOD=$ATTR2$&Transaction_POSTING=T&Transaction_MAINLINE=F&
   ```

   If you want to add an additional filter, such as Subsidiary ID, enter **&Transaction_SUBSIDIARY=$ATTR4$** to the list of parameters.

   In this case, specify the entire drill through URL definition as:

   ```
   Searchtype=Transaction&searchid=customsearch_nspbcs_trial_balance&Transaction_ACCOUNT=$ATTR1$&Transaction_POSTINGPERIOD=$ATTR2$&Transaction_SUBSIDIARY=$ATTR4$&Transaction_POSTING=T&
   ```

5. Click **Save**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Drilling Through to Oracle NetSuite

Data Integration enables you to drill through to Oracle NetSuite using an externally available URL that you specify within Data Integration.

When drilling back to the Oracle NetSuite, the drill URL is constructed by passing the stored values from Data Integration as parameters in the drill URL.

The drill through URL to Oracle NetSuite must be configured manually.

Defining Drill Through Parameters to Oracle NetSuite

The parameters required to drill back to Oracle NetSuite are described below.

The server URL for drilling to Oracle NetSuite takes the drill URL address format of: `https://<NetSuite Domain>/app/common/search/searchresults.nl?`.

On the detail side (where you specify the search type components), the drill URL to Oracle NetSuite requires the following parameters:

- "search type"
- "search ID"
- Optionally, you can specify additional parameters to filter the drill based on the Account and Period.

Search Type

The drill through parameter list includes the search type of "Transaction." It is specified in the drill through URL as:

```
Searchtype=Transaction&searchid=customsearch_nspbcs_trial_balance&Transaction_ACCOUNT=$ATTR1$&Transaction_POSTINGPERIOD=$ATTR2$&Transaction_DEPARTMENT=$ATTR5$&Transaction_CLASS=$ATTR4$&Transaction_INTERNALID=$ATTR3$&Transaction_POSTING=T&
```

The Transaction search types include:

- Transaction_POSTINGPERIOD
- Transaction_DEPARTMENT
- Transaction_SUBSIDIARY
- Transaction_CLASS
- Transaction_LOCATION
- Transaction_INTERNALID
- Transaction_POSTING=T

Search ID

The drill through list also includes "Search ID." Specify the parameter by using Search StringID.

You can find the value from the Search Definition in Oracle NetSuite.

`https://<NetSuite Domain>/app/common/search/searchresults.nl?searchtype=Transaction&searchid=customsearch_nspbcs_all_transactions_det`.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Additional Parameters

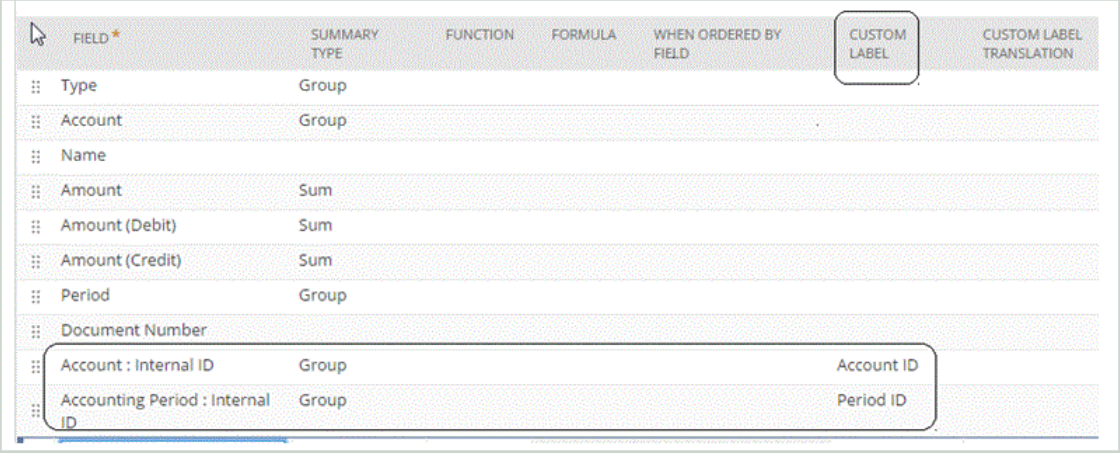
You can specify additional parameters to filter the drill based on account and period. Below are some commonly used parameters:

Table 17-2 Additional Filter Parameters

Label	Parameter	Value	Example
Account	Transaction_ACCOUNT	Account internal ID	`&Transaction_ACCOUNT=54`
Main Line	Transaction_MAINLINE	T or F	`&Transaction_MAINLINE=T`
Period	Transaction_POSTINGPERIOD	Period ID or key of the predefined period	`&Transaction_POSTINGPERIOD=21 or &Transaction_POSTINGPERIOD=LP`
Posting	Transaction_POSTING	T or F	`&Transaction_POSTING=T`
Subsidiary	Transaction_SUBSIDIARY	Subsidiary internal ID	`&Transaction_SUBSIDIARY=1`

Saved Search Requirements in the Drill Through

To use the context sensitive drill through, the Saved Searches must include an Internal ID for the account and period fields in the search Output. You can add these fields if they are missing in the Results section of your Saved Search in Oracle NetSuite as shown below:



For more information about Internal IDs, see: Using Internal IDs, External IDs, and References.

Adding the Drill Through URL

To add a drill through URL:

1. From the **Data Integration** home page, click ⋯ to the right of the integration, and then select **Map Dimensions**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

2. On the **Map Dimensions** page, from **Import Format**, select the name of the import format to use for the integration.

   You can also add a user defined import format name.

3. From the **Import Format Summary** section, select the import format.

4. In the **Import format Detail** section, map the Internal ID for Account, Period, and Transaction Type to the Attribute columns as shown below.

   In this example, the drill URL format for the Saved Search is:

   ```
   searchtype=Transaction&searchid=<NAME OF SAVED
   SEARCH>&Transaction_TYPE&detailname=$<ATTR COLUMN FOR TRANSACTION
   TYPE>$&Transaction_ACCOUNT=$<ATTR COLUMN FOR ACCOUNT
   ID>$&Transaction_POSTINGPERIOD=$=$<ATTR COLUMN FOR PERIOD
   ID>$&Transaction_POSTING=T&Transaction_MAINLINE=F&
   ```

5. In the **Import Format Detail** section, in the **Drill URL**, click .

6. In the **Drill URL** edit window, specify the attributes of the drill through.

   You specify the drill URL as:

   ```
   searchtype=Transaction&searchid=customsearch_nspbcs_all_transactions_sum&Trans
   action_TYPE&detailname=$ATTR3$&Transaction_ACCOUNT=$ATTR1$&Transaction_POSTING
   PERIOD=$ATTR2$&Transaction_POSTING=T&Transaction_MAINLINE=F&
   ```

7. Click **Save**.

Adding Additional Filters to the Drill URL in the Import Format

When performing a drill through from Oracle Fusion Cloud Enterprise Performance Management to Data Integration, and then to the Oracle NetSuite Saved Search, sometimes too many records are returned. As a guideline, when you add the amounts in all the records in the drill through, they should total to the number in the data cell you were on when you initiated the drill through in Cloud EPM. To resolve this issue, add additional filters to the drill through URL in the Import Format.

To add additional filters:

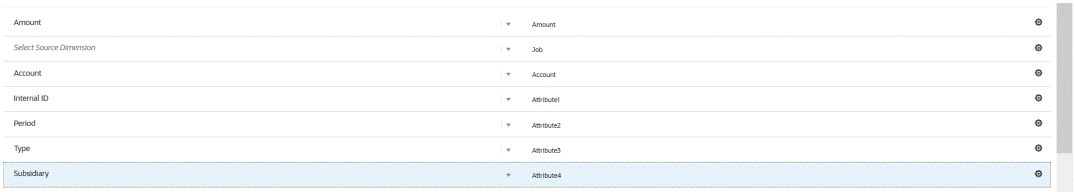
1. From the **Data Integration** home page, click to the right of the integration, and then select **Map Dimensions**.

2. On the **Map Dimensions** page, from **Import Format**, select the name of the import format to use for the integration.

   You can also add a user defined import format name.

3. In the **Mapping** section, select the source dimension column to use as a filter and map it to a target **Attribute** column

   For example, you could map a Subsidiary ID source column to the Attribute 4 row.



Copyright © 2017, 2026, Oracle and/or its affiliates.

4. In **Drill Through URL**, click and enter the search type criteria used for the drill-through.

   This is the drill through format:

   ```
   searchtype=Transaction&searchid=customsearch_nspbcs_all_transactions_s_5&Transaction_TYPE&detailname=$ATTR3$&Transaction_ACCOUNT=$ATTR1$&Transaction_POSTINGPERIOD=$ATTR2$&Transaction_POSTING=T&Transaction_MAINLINE=F&
   ```

   If you want to add an additional filter, such as Subsidiary ID, enter **&Transaction_SUBSIDIARY=$ATTR4$** to the list of parameters.

   In this case, specify the entire drill through URL definition as:

   ```
   Searchtype=Transaction&searchid=customsearch_nspbcs_trial_balance&Transaction_ACCOUNT=$ATTR1$&Transaction_POSTINGPERIOD=$ATTR2$&Transaction_SUBSIDIARY=$ATTR4$&Transaction_POSTING=T&
   ```

5. Click **Save**.

Integrating with the Oracle HCM Cloud

load Human Resources data from the Oracle Human Capital Management Cloud to use in the Oracle Workforce Planning business process.

You can load Human Resources data from the Oracle HCM Cloud to use in the Workforce Planning business process of Planning Modules or Strategic Workforce Planning.

You use Data Integration to drive the integration of data between Oracle HCM Cloud and the Workforce Planning or Strategic Workforce Planning. Data Integration provides an out of the box solution that enables Workforce Planning customers to apply predefined mappings from the Oracle HCM Cloud data model to target dimensions. You can also customize and extend these integrations, for example, by applying other mapping as needed to meet their business requirements.

Watch this video to learn how to integrate data between Oracle HCM Cloud and Workforce Planning using Data Integration. Integrating selected data from your Oracle HCM Cloud application allows you to prepare workforce plans using data directly from your Oracle HCM Cloud as the basis for your plan.

For more information, see: Integrating Oracle HCM Cloud with Workforce Planning.

Process Description for Integrating Data from Oracle HCM Cloud

Data Integration provides an out of the box solution that enables customers to apply predefined mappings from the Oracle Human Capital Management Cloud data model to target dimensions in Workforce or Strategic Workforce Planning.

Data Integration along with Oracle HCM Cloud provides templates that the user must load to the Oracle HCM Cloud instance so that data can be extracted.

Oracle HCM Cloud extracts are used to extract high volumes of data from Oracle HCM Cloud. Each of the predefined Oracle HCM Cloud extracts are defined as "Data Sources" in Data Integration.

When Data Integration performs the data extraction portion of the integration, it designates each predefined Oracle HCM Cloud extract as a data source entity. In turn, these data sources are mapped directly to the Workforce or Strategic Workforce Planning data model in Planning Modules.

Copyright © 2017, 2026, Oracle and/or its affiliates.

There is one template named "EPBCS Initialize.xml" that is not an extract template, but which is used by the test connection process and initialize process on the Data Integration side. If a user does not upload the full set of templates provided by Data Integration, they must still upload "EPBCS Initialize.xml " so that Data Integration can successfully initialize the content from the Oracle HCM Cloud system, and so that it can perform the test connection step.

The following is a description of metadata templates that are provided as part of this integration:

- Entity data including organization code, name, and available organization structure
- Employee data including person number, full name, and available demographic information
- Job data including job code, name, and available job structure
- Position data including position code, name, and available position structure
- Component data including grade code and name
- Union Code data including bargaining unit code and name

The following is a description of the data that is provided by way of the extract templates:

- FTE
- Salary Basis
- Salary Rate for defined basis
- Union Code

Customers can also pull the position start date which Data Integration converts to the start month for loading into the Workforce application. They can also customize and extend these integrations, for example, by applying other mapping as needed to meet their business requirements.

> **Note**
>
> Drill through and write-back are not supported in Oracle HCM Cloud.

At a high level, the steps for loading data from an Oracle HCM Cloud extract data source are:

1. Make sure that you have been assigned a Human Capital Management Integration Specialist job role.

   A Human Capital Management Integration Specialist job role is required to manage Human Capital Management extracts. The Human Capital Management Integration Specialist (Job Role) is the individual responsible for planning, coordinating, and supervising all activities related to the integration of human capital management information systems.

   For more information, see Human Capital Management Integration Specialist (Job Role).

2. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
3. From the **Application** page, select the application corresponding to the Workforce application, and then on the **Dimension Detail** tab, assign classifications for the seeded dimensions in Planning Modules.

   Classifications for the seeded dimensions include "Employee", "Job", "Property," and "Union" dimensions.

4. From the **Application** option, from the **Actions** option, download the Oracle HCM Cloud extracts by clicking **Download Extract**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

For more information, see Downloading Oracle HCM Cloud Extracts.

Extracts included in the zip:

An **EPBCS HCM Extract.zip** file containing the following Oracle HCM Cloud extract definitions is created. These files are a subset of the data that can be extracted and loaded into the Workforce application.

- EPBCS Assignment_<Release>.xdoz

> **Note**
>
> You must import EPBCS Assignment_<Release>.xdoz into the /Custom folder of BI Publisher and not Oracle HCM Cloud.

> **Note**
>
> If you require non-English characters, download the EPBCS HCM Extract.zip file and then unzip the zip. Next go to BI Publisher Document Repository and import the EPBCS Assignment.xdoz file.

- EPBCS Entity Metadata_<Release>.xml
- EPBCS Employee Metadata_<Release>.xml
- EPBCS Position Metadata_<Release>.xml
- EPBCS Location Metadata_<Release>.xml
- EPBCS Job Metadata_<Release>.xml
- EPBCS Initialize.xml

> **Note**
>
> In all cases, the EPBCS Initialize.xml must always be imported in Oracle HCM Cloud.

> **Note**
>
> All extracts must be imported without the Legislative Group. That is, the Legislative Group must be blank.

- EPBCS Component Salary Metadata.xml
- EPBCS Assignment Data.xml
- EPBCS Account Merit Metadata.xml

Data Integration ships the template that is needed for the extract as content that is uploaded to the Oracle HCM Cloud application. This content is provided by Data Integration because it is not seeded with the Oracle HCM Cloud.

5. Save the zip to a temporary folder.

Copyright © 2017, 2026, Oracle and/or its affiliates.

6. Import the Oracle HCM Cloud extract definitions saved in the temporary folder to the Oracle HCM Cloud.

   For more information, see Importing the Oracle HCM Cloud Extract Definition.

7. Import BI Publisher etext template in the Shared/Customer folders.

   This template converts the data into the required format.

   For more information, see Importing the BI Publisher eText Templates.

8. Validate and submit Oracle HCM extracts.

   You must validate submit an extract before submitting an integration. An integration uses this submitted extraction for fetching data while running.

   For more information, see Validating and Submitting the Oracle HCM Cloud Extract Definition.

9. From the **Application** option, select the **Connections** option and configure a connection to the Oracle HCM Cloud.

   For more information, see Configuring an Oracle HCM Cloud Connection .

10. In Data Integration in the **Application** option, register each individual Oracle HCM Cloud extract (source entity) to be used in an integration under the data source category and Oracle HCM Cloud type.

    When this step is completed, Data Integration creates:

    - the dimensions (import format) to map Oracle HCM Cloud columns to Planning Modules dimensions,
    - a location,
    - mappings to import data to the correct Oracle Workforce Planning accounts,

    For more information, see Registering an Oracle HCM Cloud Application.

11. Create the integration between the Oracle HCM Cloud source application and the target Oracle Fusion Cloud Enterprise Performance Management application.

    > **Note**
    >
    > Each Oracle HCM Cloud extract imported is registered automatically as a target data source application.

12. If necessary, modify any dimension details.

    All columns from the Oracle HCM Cloud extract are mapped to the EPM target dimensions class with the type "Generic."

    For more information, see Mapping Dimensions.

    > **Note**
    >
    > As a rule, when loading data from an Oracle HCM Cloud data source, do not change, add, or delete any dimension details on the Map Dimensions page.

13. Member mappings are predefined when the Oracle HCM Cloud extract is registered as an application.

    Additionally, Oracle HCM Cloud extracts support the transformation of actual data imported from Oracle HCM Cloud in the Data dimension column.

Copyright © 2017, 2026, Oracle and/or its affiliates.

For example, in the Oracle HCM Cloud, the employee type might be "F" (for full time employee type), or "T" (for temporary employee) while in Planning Modules, the same designations are shown as "FULLTIME," or "TEMP."

14. Any application filters associated with the Oracle HCM Cloud data source are predefined when the application is registered.

    You can select any specific criteria on the Source Filters tab to filter the results that are loaded.

    For more information about filters, see Defining Direct Integration Options.

15. Run the integration.

    For more information, see Running an Integration.

Downloading Oracle HCM Cloud Extracts

For an out-of-box integration with the Oracle Human Capital Management Cloud, you download the Oracle HCM Cloud extracts.

Data Integration loads the extract definitions on the **Applications** page. These files are a subset of the data that can be extracted and loaded into the Workforce business process application of the Planning Modules or Strategic Workforce Planning. .

To download the Oracle HCM Cloud extracts:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. From the **Applications** page, and then from the **Actions** menu, select **Download HCM Extract**.
3. An **EPBCS HCM Extract.zip** file containing the following Oracle HCM Cloud extract definitions is created. These files are a subset of the data that can be extracted and loaded into the Planning Modules application.
   - EPBCS Assignment_<Release>.xdoz

     > **Note**
     >
     > You must import EPBCS Assignment_<Release>.xdoz into the /Custom folder of BI Publisher and not Oracle HCM Cloud.

     > **Note**
     >
     > If you require non-English characters, download the EPBCS HCM Extract.zip file and then unzip the zip. Next go to BI Publisher Document Repository and import the EPBCS Assignment.xdoz file.

   - EPBCS Entity Metadata_<Release>.xml
   - EPBCS Employee Metadata_<Release>.xml
   - EPBCS Position Metadata_<Release>.xml
   - EPBCS Location Metadata_<Release>.xml
   - EPBCS Job Metadata_<Release>.xml
   - EPBCS Initialize.xml

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> In all cases, the EPBCS Initialize.xml must always be imported in Oracle HCM Cloud.

> **Note**
>
> All extracts must be imported without the Legislative Group. That is, the Legislative Group must be blank.

- EPBCS Component Salary Metadata.xml
- EPBCS Assignment Data.xml
- EPBCS Account Merit Metadata.xml

Data Integration ships the template that is needed for the extract as content that is uploaded to the Oracle HCM Cloud application. This content is provided by Data Integration because it is not seeded with the Oracle HCM Cloud.

4. Save the zip to a temporary folder.

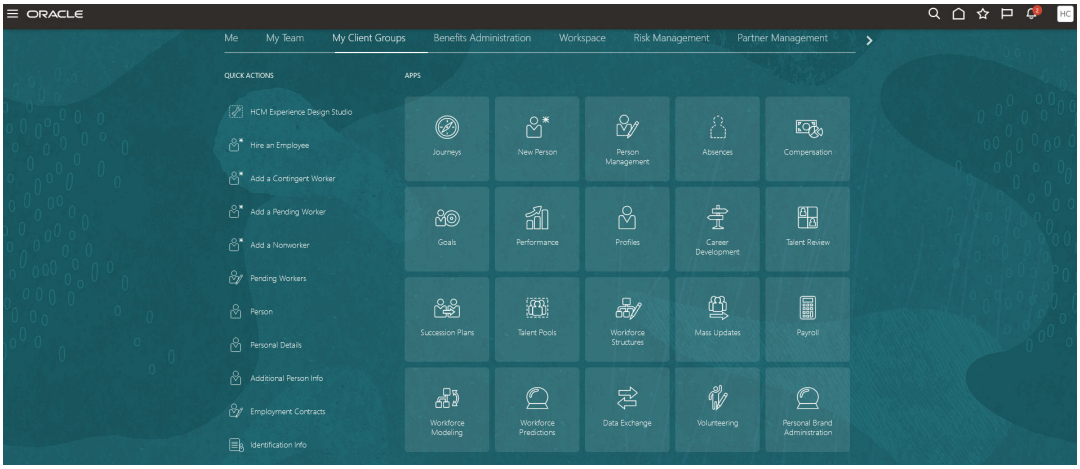
Importing Oracle HCM Cloud Extract Definitions to Oracle HCM Cloud

To begin seeding extract definitions in the Oracle Human Capital Management Cloud application, you import Oracle HCM Cloud extract definition XML files and BI Publisher e-Text template XDOZ files where the extract output is saved in a comma separated value (CSV) file. The CSV file format is defined as a BI Publisher report, and this report template is specified as part of the extract definition in the Oracle HCM Cloud application.

Importing the Oracle HCM Cloud Extract Definition

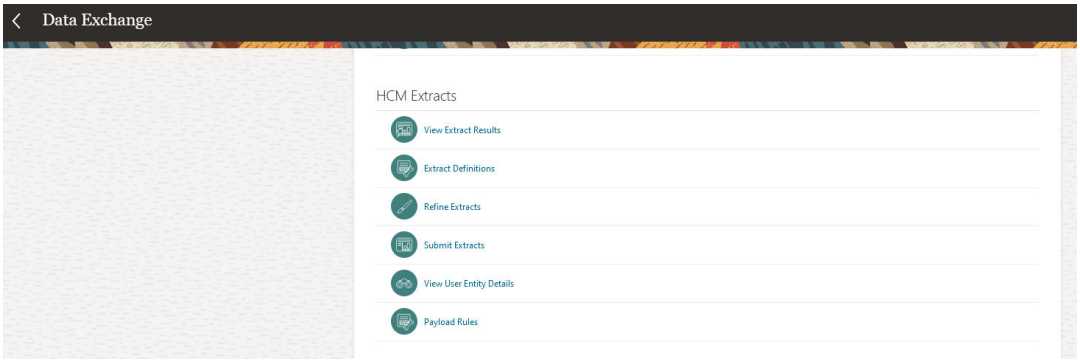
To import the Oracle Human Capital Management Cloud extract definitions that were exported from Data Integration:

1. Log on to the Oracle HCM Cloud application.
2. From the **Fusion Navigation** menu, and then from **My Client Groups**, select the **Data Exchange** application.

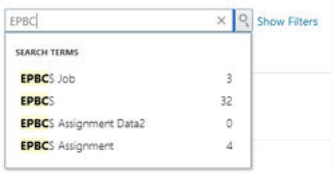


3. From the **Task** page, **HCM Extracts**, select **Extract Definitions**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



4. On the **Extract Definitions** page, click Import (**Import**) to import the pre-defined Oracle HCM Cloud extract definition XML files.

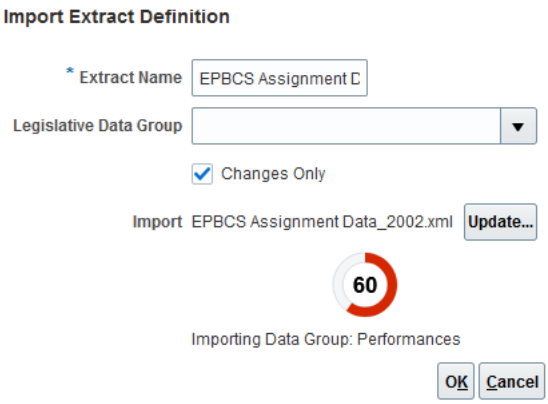
> **Note**
>
> You can search on an extract definition name by typing the extract name in the Search field:
>
> 
>

5. On the **Import Extract Definition** page, complete the following, then click **Update**, and then **OK** (when the throbber shows the extract import is complete):

   - **Extract Name**—Specify the name of the Oracle HCM Cloud extract to be imported.

     When importing the extract definitions, the extract name must be same as the first name of the file name. For example, when importing "EPBCS Assignment Data_2002.xml," the extract name must be specified as "EPBCS Assignment Data_2002."

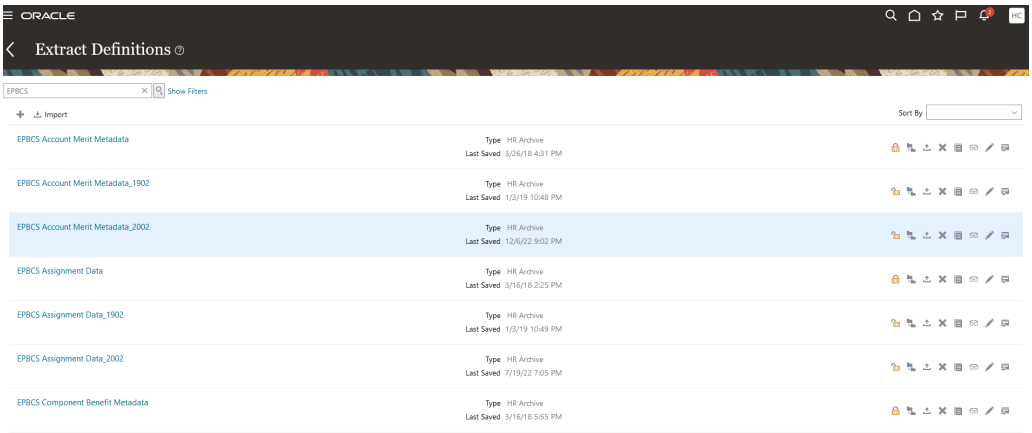
   - **Legislative Data Group**—Leave blank.
   - **Changes Only**—Select this option to create an extract that captures the incremental changes only and not a full extract.



When the extract definition has been imported successfully, a confirmation message is displayed. You can also download a log file of the import from the confirmation page.

Copyright © 2017, 2026, Oracle and/or its affiliates.

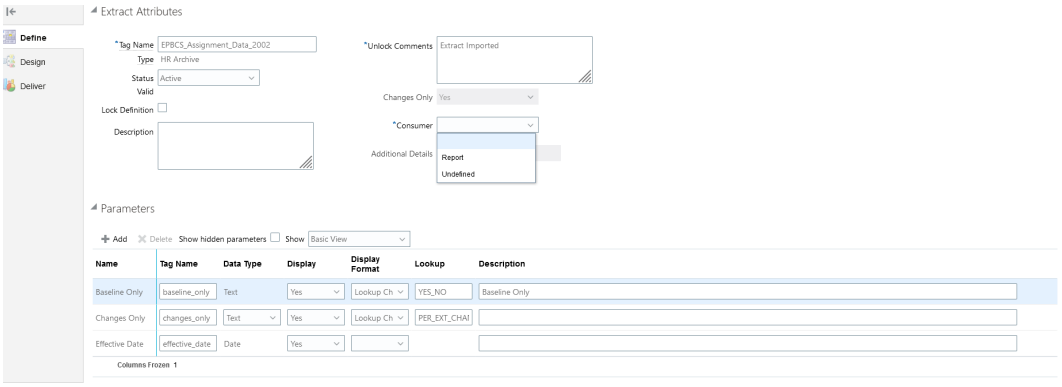
6. Import all the pre-defined Oracle HCM Cloud extract definitions:
   - EPBCS Account Merit Metadata—EPBCS Account Merit Metadata_<Release>.xml
   - EPBCS Assignment Data—EPBCS Assignment Data_<Release>.xml
   - EPBCS Component Salary Metadata—EPBCS Component Salary Metadata_<Release>.xml
   - EPBCS Employee Metadata—EPBCS Employee Metadata_<Release>.xml
   - EPBCS Entity Metadata—EPBCS Entity Metadata_<Release>.xml
   - EPBCS Job Metadata—EPBCS Job Metadata_<Release>.xml
   - EPBCS Location Metadata—EPBCS Location Metadata_<Release>.xml
   - EPBCS Position Metadata—EPBCS Position Metadata_<Release>.xml



   You can import only the extract definitions that you plan to load from Oracle HCM Cloud, or all extracts even if you are not using them. All XML files that are loaded need to be validated after all files are loaded.

7. For each imported extract, on the **Extract Attribute** page, then from **Consumer**, select **Report** to indicate that the extract is used for reporting purposes and that the data is not interfaced with any external applications. Otherwise, select **Undefined**.

8. Modify any other attributes as necessary.



   For more information about Oracle HCM Cloud extracts, see HCM Extracts.

9. Click **OK**.

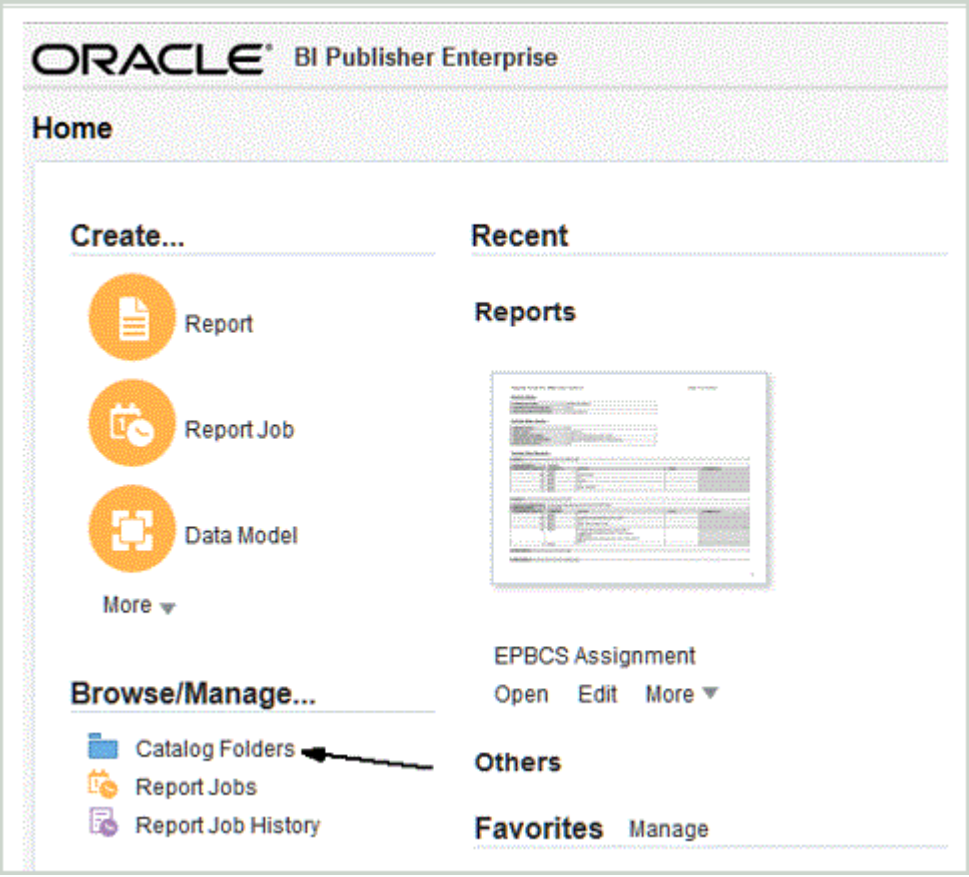
   The application saves the extract definition and automatically generates the parameters based on the type of extract. The parameters control the output of an extract.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Importing the BI Publisher eText Templates

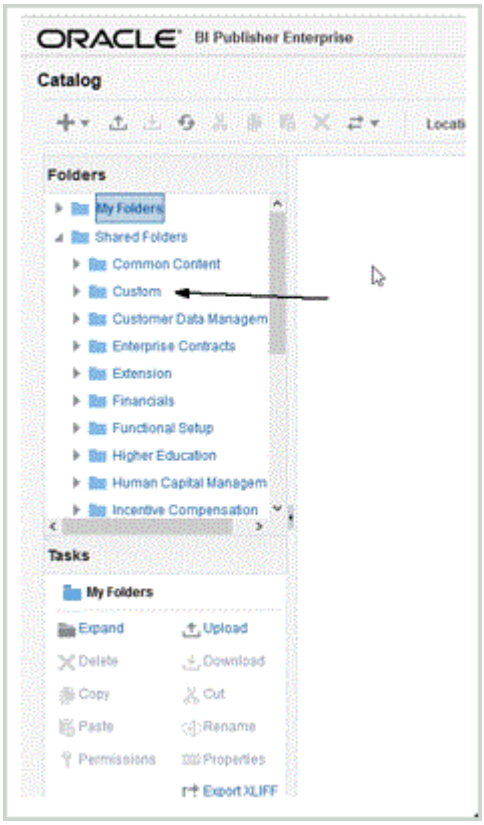
To import the BI Publisher eText templates:

1. Log on to the **BI Publisher Enterprise** by specifying the following address in a browser: `https://server`.
2. Click **Catalog Folders**.

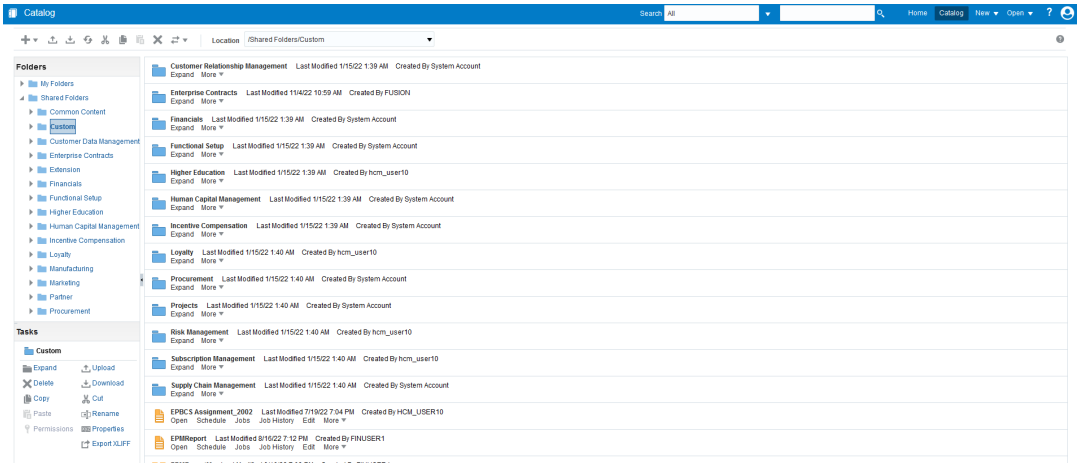


3. From the **Catalog** screen, and then under **Shared Folders**, select **Custom**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



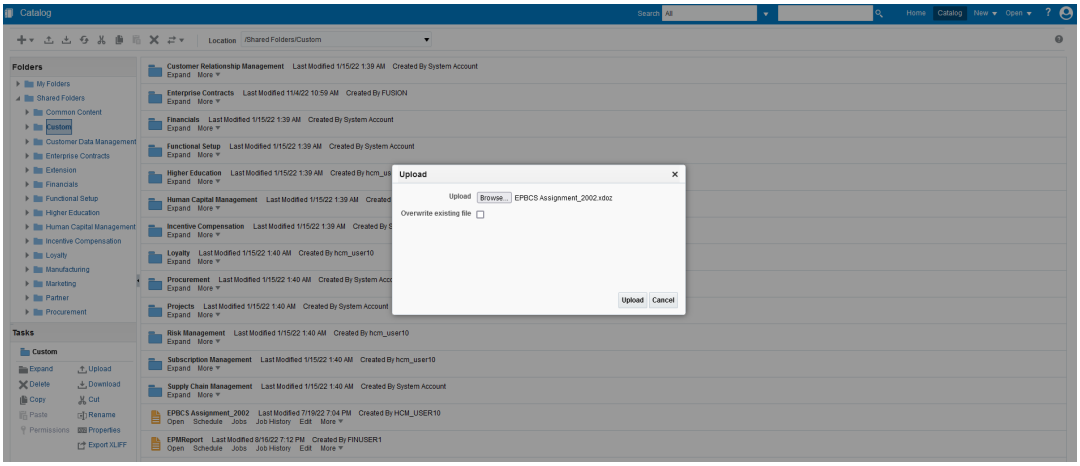
4. Expand the **Custom** folder.



5. Browse to and select the predefined template files and then click **Upload**.

   For example, upload the EPBCS Assignment_<Release>.xdoz template file.

Copyright © 2017, 2026, Oracle and/or its affiliates.



Validating and Submitting the Oracle HCM Cloud Extract Definition

An extract definition automatically creates an extract process (payroll flow) with the same name as the extract. The extract process enables you to define an execution sequence of multiple tasks, including pre and post tasks using the extract process. You can use the Refine HCM Extracts task to view and modify the extract process submission parameters, if required.

When you validate an extract, the application performs validations on the extract design to ensure that:

- The root data group is defined.
- All non-root data groups are linked to the root data group directly or indirectly (for example, through another non-root data group that is linked to the root data group).
- If a sequence of data groups is defined, then the next data group is also defined for processing.
- All Fast Formulas used in the extract exist and are compiled or valid.
- There are no issues detected during BI Publisher validations.

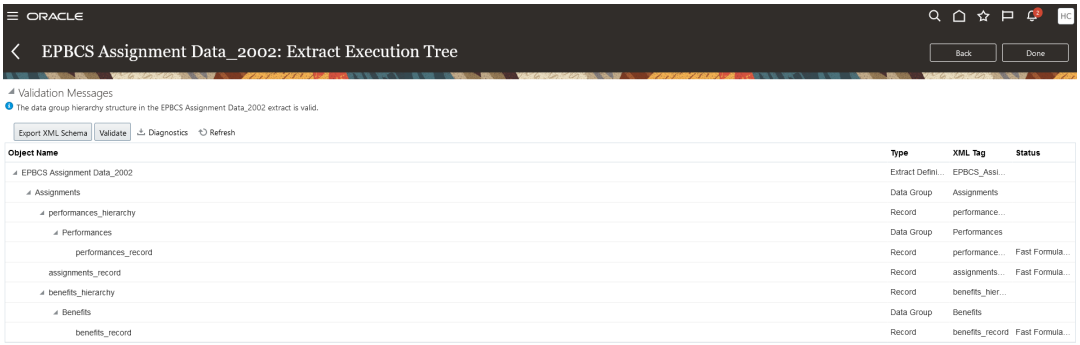
If the validation is successful, the application marks the extract as valid. However, if the validation fails, the application marks the extract as invalid and submitting the extract results in errors.

You can run the flow pattern directly using the Submit an HCM Process option.

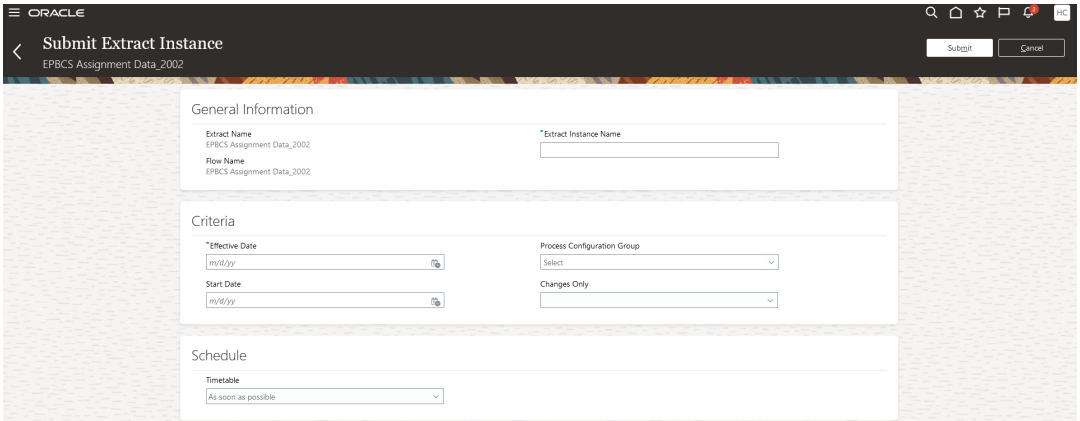
To validate and submit an extract:

1. On the **Extract Definitions** page, double click the Oracle Human Capital Management Cloud extract definition.
2. On the **Extract Attribute** page, click **Validate**.
3. On the **Extract Execution Tree** page, click **Validate**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



4. If your fast formulas need to be compiled, you get a message that your fast formulas have been submitted for compiling. If necessary, click **Refresh** until all fast formula are compiled as annotated by a green checkmark in the **Status** column.
5. On the **Validation Messages** page, click **OK**.
6. When the validation has been successfully completed, a message indicates that the data group hierarchy structure in that the <extract_name> extract is valid.
7. Click **Done**.
8. Repeat steps 1-7 for all imported pre-defined Oracle HCM Cloud extract definitions:
9. From the **Tasks** menu, and then under **HCM Extracts**, click **Submit Extracts**.
10. Narrow down the list of available definition to the one imported by entering a few characters of the definition name in the Search text box.
11. Double-click the extract definition and click **Next**.
12. On the **Submit Extract Instance** page, enter the extract name in **Extract Instance Name** and an effective date in the **Effective date** of the extract.



13. Click **Submit** to submit the extract.
14. Click **Done**.

Configuring an Oracle HCM Cloud Connection

Define a Oracle Human Capital Management Cloud source connection to connect to a Oracle HCM Cloud data source.

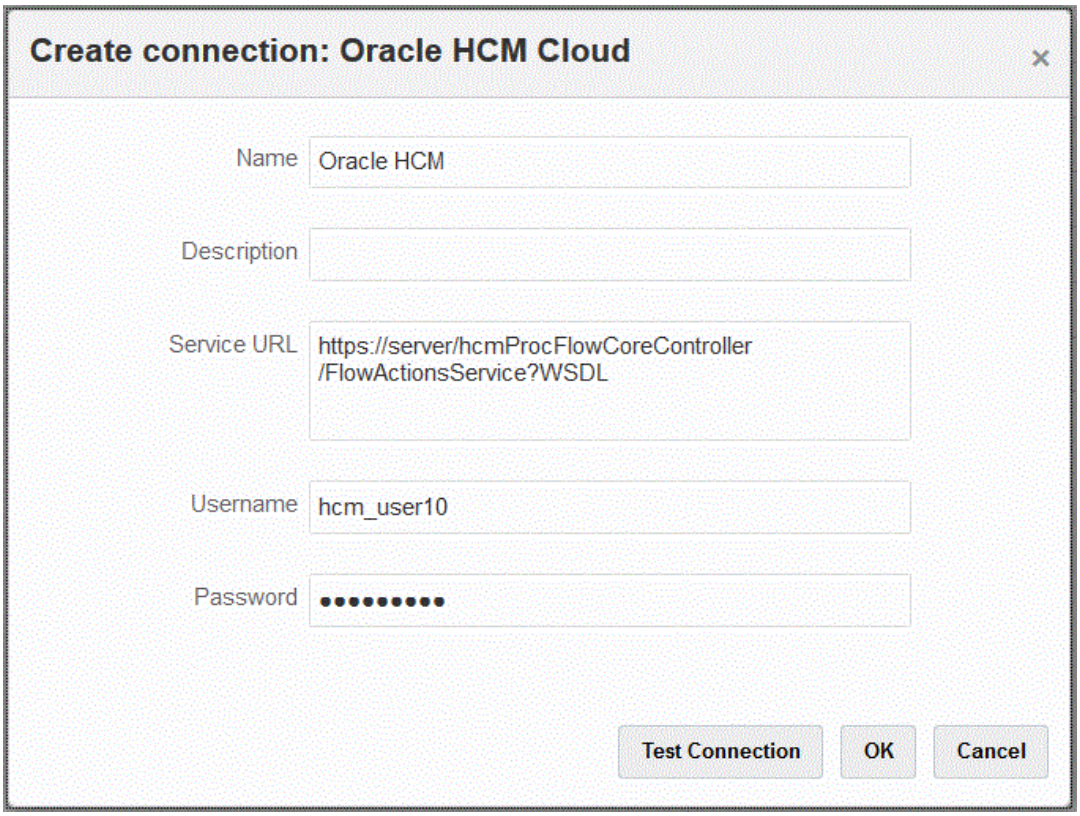
To begin integrating Oracle HCM Cloud with the Workforce business process of Planning Modules or Strategic Workforce Planning, first create and register the source system with the type "Oracle HCM Cloud."

Copyright © 2017, 2026, Oracle and/or its affiliates.

After the source system and connection information are specified, you initialize the source system. This process creates a target application record for each Oracle HCM Cloud extract.

To configure an Oracle HCM Cloud connection:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, from the **Actions** menu, click **Configure Connections**.
3. On the **Connections** page, click ⋯, and then click **Oracle Cloud HCM**.



4. In **Name**, enter the source system name.
5. In **Description**, enter a description of the source system.
6. In **Service URL**, enter the server information for the web services.
7. In **Username**, enter the Oracle HCM Cloud user name.
8. In **Password**, enter the Oracle HCM Cloud password.

   You must update this password anytime you change your Oracle HCM Cloud password.
9. Click **Test Connection**.

   When the connection has been tested correctly, the information message "Connection to [*source system name*] successful" is displayed.
10. Click **OK**.

Registering an Oracle HCM Cloud Application

When integrating human resources data from Oracle Human Capital Management Cloud with Oracle Workforce Planning business processes, register the Oracle HCM Cloud extracts as a "data source" application type.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Data Integration provides an out of the box solution that enables customers to apply predefined mappings from the Oracle HCM Cloud data model.

Data Integration and Oracle HCM Cloud provide templates that the user must load to the Oracle HCM Cloud instance so that data can be extracted.

Oracle HCM Cloud extracts are used to extract high volumes of data from Oracle HCM Cloud. Each of the predefined Oracle HCM Cloud extracts are defined as "Data Sources" in Data Integration.

When Data Integration performs the data extraction portion of the integration, it designates each predefined Oracle HCM Cloud extract as a data source entity. In turn, these data sources are mapped directly to the Workforce Planning data model.

> **Note**
>
> A Human Capital Management Integration Specialist job role is required to manage Human Capital Management extracts. The Human Capital Management Integration Specialist (Job Role) is the individual responsible for planning, coordinating, and supervising all activities related to the integration of human capital management information systems.
>
> For more information, see Human Capital Management Integration Specialist (Job Role).

To register an Oracle HCM Cloud application:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. From the **Applications** page, click (Add icon).
3. From the **Create Application**, then **Category**, select **Data Source**.
4. From **Type**, select **Oracle HCM Cloud**.
5. From **Connection**, select the name of the connection to the Oracle HCM Cloud type.

   Set up the Oracle HCM Cloud source system and specify the connection information in the Configure Connection option. See Configuring an Oracle HCM Cloud Connection.
6. In **File**, specify the file name (in XML format) of the source entity to register as a data source.

   The file is the Oracle HCM Cloud extract application name/definition that has already been uploaded to the *inbox*.
7. Click **OK**.

> **Note**
>
> The following steps are required only if you are registering a custom extract. For an out-of-box integration, you simple perform the steps 1-7 above .

1. From the **Data Integration** home page, and then **Actions**, select **Applications**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

2. From the **Application** page, click to the right of the Workforce Planning application, and then select **Application Detail**.
3. On the **Application Detail** page, assign classifications for the seeded dimensions in Planning.

   Classifications for the seeded dimensions include "Employee," "Job,""Property," and "Union."

   For more information, see Defining Application Dimension Details.
4. Click **Save** and then click **Return**.

Updating Existing Oracle HCM Cloud Extracts

When an Oracle Human Capital Management Cloud extract is updated, new functionality or definitions are provided and a new file is created and made available in the EPBCS HCM Extract.zip. This feature enables you to take the latest updates when new extract definitions become available and shipped.

You can retain all existing mappings and definitions for a selected or all target applications based on an extract definition in Data Integration when the extract is updated.

Any new files show the release version in the file name. For example, if extracts are updated in Release 19.02, the extract file and name are as follows:

- File name—EPBCS Assignment Data_1902.xml
- Extract name—EPBCS Assignment Data_1902
- XML Tag Name—EPBCS Assignment Data (no release version added)

> **Note**
>
> The Update Application step is provided to update an application when new versions of the out-of-box applications are provided by Oracle. When you select **Update Application**, it imports the latest version of the XML shipped out of the box.
>
> To customize and add a new attribute, copy the extract in Oracle HCM Cloud and create a new extract definition in then Oracle HCM Cloud and then modify it. Then export the XML and create a new application in the Oracle Fusion Cloud Enterprise Performance Management. Do not use **Update Application** to update your custom application definition.
>
> You should not use Update Application to update your custom application definition.

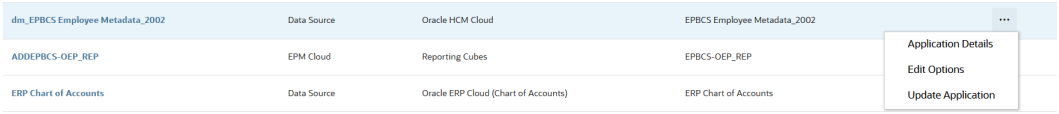
To select an updated version of an Oracle HCM Cloud extract:

> **Note**
>
> These steps assume that you have configured the source system, defined the source connection, and downloaded the EPBCS HCM Extract.zip. For information about any of these processes, see Downloading Oracle HCM Cloud Extracts.

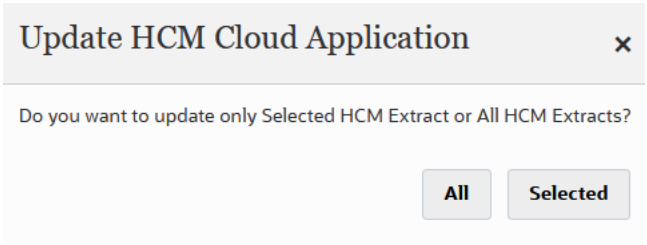
Copyright © 2017, 2026, Oracle and/or its affiliates.

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.

2. On the **Applications** page, click ... to the right of the Oracle HCM Cloud application, and then select **Update Application**.



3. From the **Update HCM Cloud Application**, when prompted with the message, "Do you want to update only Selected Extract or All HCM Extracts?", click **All** to update all extracts, or click **Selected** to update just the individual extract.



Integrating Oracle HCM Cloud Metadata

You can integrate Oracle Human Capital Management Cloud metadata to use in the Workforce business process of Planning Modules or Strategic Workforce Planning.

As the integration mechanism, Data Integration can align and move the following metadata from the Oracle HCM Cloud to your target applications and business processes in the Planning Modules:

- Account Merit
- Component Salary Metadata
- Employee Metadata
- Entity Metadata
- Job Metadata
- Location Metadata
- Position Hierarchy

During the initialization of the Oracle HCM Cloud source system, Data Integration creates an application for each metadata source. You can map each application to your metadata application and then execute the load. Note that the system does not create mappings automatically.

> **Note**
>
> For detailed information on the Oracle HCM Cloud fields belonging to each predefined extract definition, seeOracle HCM Cloud Extract Definition Field Reference.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Loading Oracle HCM Cloud Metadata

> **Note**
>
> Drill through and write-back are not supported in an integration with Oracle Human Capital Management Cloud.

At a high level, the steps for loading data from an Oracle HCM Cloud extract data source are:

1. Make sure that you have been assigned a Human Capital Management Integration Specialist job role.

   A Human Capital Management Integration Specialist job role is required to manage Human Capital Management extracts. The Human Capital Management Integration Specialist (Job Role) is the individual responsible for planning, coordinating, and supervising all activities related to the integration of human capital management information systems.

   For more information, see Human Capital Management Integration Specialist (Job Role).

2. In Data Integration, then from **Application** option, select the application corresponding to the Workforce application, and then on the **Dimension Detail** tab, assign classifications for the seeded dimensions in Planning Modules.

   Classifications for the seeded dimensions include "Employee", "Job", "Property," and "Union" dimensions.

3. From **Applications** page and then from the **Actions** menu, download the Oracle HCM Cloud extracts by clicking **Download HCM Extract**.

   For more information, see Downloading Oracle HCM Cloud Extracts.

   An **EPBCS HCM Extract.zip** file containing the following Oracle HCM Cloud extract definitions is downloaded. These files are a subset of the data that can be extracted and loaded into the Workforce application.

   - EPBCS Assignment_<Release>.xdoz

     > **Note**
     >
     > You must import EPBCS Assignment_<Release>.xdoz into the /Custom folder of BI Publisher and not Oracle HCM Cloud.

     > **Note**
     >
     > If you require non-English characters, download the **EPBCS HCM Extract.zip** file and then unzip it. Next go to **BI Publisher Document Repository** and import the **EPBCS Assignment.xdoz** file.

   - EPBCS Entity Metadata_<Release>.xml
   - EPBCS Employee Metadata_<Release>.xml
   - EPBCS Position Metadata_<Release>.xml
   - EPBCS Location Metadata_<Release>.xml

Copyright © 2017, 2026, Oracle and/or its affiliates.

- EPBCS Job Metadata_<Release>.xml
- EPBCS Initialize.xml

> **Note**
>
> In all cases, the EPBCS Initialize.xml must always be imported to the Oracle HCM Cloud.

> **Note**
>
> All extracts must be imported without the Legislative Group. That is, the Legislative Group must be blank.

- EPBCS Component Salary Metadata.xml
- EPBCS Assignment Data.xml
- EPBCS Account Merit Metadata.xml

Data Integration ships the template that is needed for the extract as content that is uploaded to the Oracle HCM Cloud application. This content is provided by Data Integration because it is not seeded with the Oracle HCM Cloud.

4. Save the zip to a temporary folder.
5. Import the Oracle HCM Cloud extract definitions saved in the temporary folder to the Oracle HCM Cloud.

   For more information, see Importing the Oracle HCM Cloud Extract Definition.
6. Import BI Publisher etext template in the Shared/Customer folders.

   This template converts the data into the required format.

   For more information, see Importing the BI Publisher eText Templates.
7. Validate and submit the Oracle HCM Cloud extracts.

   You must validate submit an extract before submitting an integration. An integration uses this submitted extraction for fetching data while running.

   For more information, see Validating and Submitting the Oracle HCM Cloud Extract Definition.
8. From the **Application** page, select the **Connections** option and configure a connection to the Oracle HCM Cloud.

   For more information, see Configuring an Oracle HCM Cloud Connection.
9. In Data Integration from the **Application** page, register each individual Oracle HCM Cloud extract (source entity) to be used in an integration under the data source category and Oracle HCM Cloud type.

   When this step is completed, Data Integration:

   - creates the dimensions (import format) to map Oracle HCM Cloud columns to Planning Modules dimensions,
   - creates a location,
   - creates mappings to import data to the correct Oracle Workforce Planning accounts,

Copyright © 2017, 2026, Oracle and/or its affiliates.

For more information, see Registering an Oracle HCM Cloud Application.

10. From the **Application** page, click to the right of the Oracle HCM Cloud application, and then select **Application Detail**.
11. Select the **Options** tab.
12. Select any applicable application detail options and click **Save**.

    Options include:

    Any source filters associated with the data source are created automatically during the integration. You can select any specific criteria on the Source Filters tab to filter the results that are loaded.

    Depending on the Oracle HCM Cloud metadata category, the following source filters apply:

    - Effective Date—Select the date on which you want the trees to be effective.
    - Legislative Data Group—Legislative data groups are a means of partitioning payroll and related data. At least one legislative data group is required for each country where the enterprise operates. Each legislative data group is associated with one or more payroll statutory units.
    - Tree Code—Tree code for hierarchy in Oracle HCM Cloud (for objects with hierarchy, for example: Org, Position)
    - Tree Version—Tree Version for hierarchy in Oracle HCM Cloud
    - Changes Only—Controls the extract mode. Valid options are **N** or **Y**.
      The following table describes the different extract modes, their lookup values and descriptions:

Mode	Lookup Value	Description
N	All attributes	Includes all data in the extract. A full extract is run which produces the full data output at that point of time. The archived data is utilized as a baseline.
Y	Changed attributes	Compares this extract run with the previous extract runs and by comparing against the baseline (to identify the incremental data), displays the data that has changed only.

    > **Note**
    >
    > You can also select the above options on the **Filters** tab in Options.

13. Create the integration between the Oracle HCM Cloud source application and the target Oracle Fusion Cloud Enterprise Performance Management application.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> Each Oracle HCM Cloud extract imported is registered automatically as a target data source application.

14. If necessary, modify any dimension details.

    All columns from the Oracle HCM Cloud extract are mapped to the EPM target dimensions class with the type "Generic."

    For more information, see Mapping Dimensions.

> **Note**
>
> As a rule, when loading data from an Oracle HCM Cloud data source, do not change, add, or delete any dimension details on the Map Dimensions page.

15. Member mappings are predefined when the Oracle HCM Cloud extract is registered as an application.

    Additionally, Oracle HCM Cloud extracts support the transformation of actual data imported from Oracle HCM Cloud in the Data dimension column.

    For example, in the Oracle HCM Cloud, the employee type might be "F" (for full time employee type), or "T" (for temporary employee) while in Planning Modules, the same designations are shown as "FULLTIME," or "TEMP."

16. Any application filters associated with the Oracle HCM Cloud data source are predefined when the application is registered.

    You can select any specific criteria on the Source Filters tab to filter the results that are loaded.

    For more information about filters, see Defining Direct Integration Options.

17. Run the integration.

    For more information, see Running an Integration.

Oracle HCM Cloud Extract Definition Field Reference

The tables in this section list the Oracle Human Capital Management Cloud fields belonging to each predefined extract definition. These fields are a subset of the data that can be extracted and loaded into a Oracle Workforce Planning or Strategic Workforce Planning application from each extract definition.

- Account Merit Extract Definition Fields
- Assignment Extract Definition Fields
- Component Extract Definition Fields
- Employee Extract Definition Fields
- Entity Extract Definition Fields
- Job Extract Definition Fields
- Location Extract Definition Fields
- Position Extract Definition Fields

Copyright © 2017, 2026, Oracle and/or its affiliates.

Account Merit Extract Definition Fields

The following table shows the Account Merit Extract Definition fields that can be extracted and loaded to the Workforce or Strategic Workforce Planning application from the extract definition.

Account: Merit Extract Definition Fields
Extract Performance Rating Short Description
Extract Performance Rating Description

Assignment Extract Definition Fields

The following table shows the Assignment Extract Definition fields that can be extracted and loaded to the Workforce or Strategic Workforce Planning application from the extract definition.

Assignment Extract Definitions Fields
Assignment Bargaining Unit Code
Assignment Bargaining Unit Code Name
Assignment FTE Value
Assignment Full Part Time
Assignment Type
Assignment Salary Amount
Assignment Salary Basis Code
Assignment Salary Currency Code
Assignment Organization Code
Assignment Organization Name
Extract Assignment Organization Classification
Person Number
Person Start Date
Person Date of Birth
Person Gender
Person Highest Education Level
Assignment Job Code
Job Family Name
Assignment Grade Code
Assignment Position Code
Assignment Number
Assignment Position Type
Assignment Job Function Code
Person Full Name
Assignment Grade Name
Assignment Job Name
Assignment Location Code
Assignment Location Name
Assignment Employee Category
Assignment Employment Category
Assignment Name
Assignment Position Name
Extract Assignment Effective Start Date

Copyright © 2017, 2026, Oracle and/or its affiliates.

Assignment Extract Definitions Fields
Person Gender Meaning
Person Highest Education Level Meaning
Assignment Grade Type
Assignment Business Unit Name
Assignment Legal Employer Name
Assignment Legislation Code
Assignment Legislation Name
Assignment General Ledger Code Combination Segment1
Assignment General Ledger Code Combination Segment2
Assignment General Ledger Code Combination Segment2
Assignment General Ledger Code Combination Segment3
Assignment General Ledger Code Combination Segment4
Assignment General Ledger Code Combination Segment5
Assignment General Ledger Code Combination Segment6
Organization GL Cost Center - Company
Organization GL Cost Center - Cost Center
Extract Performance Rating Short Description
Extract Performance Rating Description
Benefit Amount
Benefit Type
Plan - Name
Plan Type - Name
Annual Rate Value
Unit of Measure
Option Name

Component Extract Definition Fields

The following table shows the Component Extract Definition fields that can be extracted and loaded to the Workforce or Strategic Workforce Planning application from the extract definition.

Component Extract Definition Fields
Extract Grade Code
Grade Name
Extract Grade Type

Employee Extract Definition Fields

The following table shows the Employee Extract Definition fields that can be extracted and loaded to the Workforce or Strategic Workforce Planning application from the extract definition.

Employee Extract Definition Fields
Extract Employee Person Number
Person Full Name
Person Date of Birth
Person Enterprise Hire Date

Copyright © 2017, 2026, Oracle and/or its affiliates.

Employee Extract Definition Fields
Person Gender
Person Gender Meaning
Person Highest Education Level
Person Highest Education Level Meaning
Assignment Employee Category
Assignment Employee Category Meaning
Assignment Bargaining Unit Code
Assignment Bargaining Unit Code Name
Person First Name
Person Last Name
Assignment Employment Category
Assignment Employment Category Meaning
Location Code

Entity Extract Definition Fields

The following table shows the Entity Extract Definition fields that can be extracted and loaded to the Workforce or Strategic Workforce Planning application from the extract definition.

Entity Extract Definition Fields
Extract Organization Tree Code
Extract Organization Tree Version Name
Extract Organization Tree Distance
Extract Organization Tree Depth
Extract Organization Tree Is Leaf
Extract Organization Tree Code
Extract Organization Tree Name

Job Extract Definition Fields

The following table shows the Job Extract Definition fields that can be extracted and loaded to the Workforce or Strategic Workforce Planning application from the extract definition.

Job Extract Definition Fields
Extract Job Code
Job Name
Job Family Name
Extract Job Function Code
Extract Job Function Code Meaning

Location Extract Definition Fields

The following table shows the Job Extract Definition fields that can be extracted and loaded to the Workforce or Strategic Workforce Planning application from the extract definition.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Location Extract Definition Fields
Extract Location Code
Extract Location Name
Extract Location Country
Extract Location Town or City
Extract Location Region1
Extract Location Region2
Extract Location Region3

Position Extract Definition Fields

The following table shows the Position Extract Definition fields that can be extracted and loaded to the Workforce or Strategic Workforce Planning application from the extract definition.

Position Extract Definition Fields
Extract Position Tree Code
Extract Position Tree Version Name
Extract Position Tree Depth
Extract Position Tree Distance
Extract Position Tree Is leaf
Extract Position Code
Position Name

Integrating SAP

When using Oracle Fusion Cloud EPM, it is easy to integrate with SAP source systems to load the required data for your selected business process.

Data may be integrated from SAP to Cloud EPM using the following methods:

- Export data from your SAP system to a file, and then import into Cloud EPM.
- Configure the EPM Integration Agent to either connect directly to the source SAP database, or use the scripting feature in the agent to call the necessary SAP Business Application Programing Interface (BAPI).
- Use the EPM Integration Agent with a JDBC driver to directly access SAP tables. (Check SAP contract for direct access details).
- Define an integration using an SAP adapter available in Oracle Integration Cloud (OIC) to extract data from SAP, and then call the EPM REST API to load the data into your Cloud EPM instance. For more information, see the SAP Adapter Capabilities in the *Using the SAP Adapter with Oracle Integration Generation 2* guide.
- Agent scripting calls to SAP Open Data Protocol (Odata)

  Allows REST call to ABAP to return data in JSON or XML

These integration methods may be used with any version of SAP including: ECC, S/4 HANA on-premises, S/4 HANA cloud or SAP Business by Design.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Integrating with JD Edwards

You can easily load JD Edwards source system data for your selected business process using these methods using one of the following methods:

- Flat File
  - JD Edwards Team provides data extract.
  - Staged for used by the Oracle Fusion Cloud Enterprise Performance Management.
  - Orchestrate with EPMAutomate or a REST API.
- EPM Integration Agent
  - Use Agent with Data Access Driver (DAD).
  - Learning path is available that provides integration example at: Integrating Oracle Cloud Enterprise Performance Management with JD Edwards EnterpriseOne
- Oracle Integration Cloud
  - JD Edward Adapter documentation is available at: Understand the Oracle JD Edwards EnterpriseOne Adapter.
  - Use OIC REST adapter to connect to Cloud EPM.

Integrating Account Reconciliation Data

This section describes how to integrate Account Reconciliation data.

Loading Reconciliation Compliance Balances

Load data from a source system business process to your Reconciliation Compliance Balances target application.

You use Data Integration as the integration mechanism between a source system business process and your Reconciliation Compliance Balances target application. You can set up mapping rules to assign the account balances to the reconciliations, and when balances are imported, ensure they appear in the correct reconciliation based on these rules.

Loading Reconciliation Compliance Balances Process Description

At a high level, this is how you load balances to Account Reconciliation using Data Integration.

To set up the load for Reconciliation Compliance Balances in Data Integration:

1. In Data Integration, set up Global and Application mappings in Period Mappings.

   Both Global and Application Mappings with a Reconciliation Compliance Balances application are required.

   For more information, see Global Mappings and Application Mappings.
2. Create an integration between the source system or file and the target Reconciliation Compliance Balances application.

   When creating the integration, you also specify location information.

   See Creating a Reconciliation Compliance Balances Integration.

Copyright © 2017, 2026, Oracle and/or its affiliates.

3. Map the dimensions between the source file to the dimensions in the Reconciliation Compliance Balances.
4. Test the integration.

   See Running the Reconciliation Compliance Balances Integration.
5. Import balances into Account Reconciliation from **Application** and then **Periods**.

Tutorials

Tutorials provide instructions with sequenced videos and documentation to help you learn a topic.

Your Goal	Learn How
Import balances with Data Integration and load them into Account Reconciliation.	Managing Account Reconciliation balance load

Creating a Reconciliation Compliance Balances Integration

You can load balances imported as source system and sub system data sources to a Reconciliation Compliance Balances target application by creating an integration.

1. From the **Data Integration** home page, click , and then select **Integration**.

   The General page is displayed in Create Integration view.
2. In **Name** and **Description**, enter a name and description for the Reconciliation Compliance Balances integration.
3. In **Location**, enter a new location name, or pick an existing location to specify where to load data..
4. Click (Select a Source).
5. From the **Select a Source** page, select the source system to use for the integration. **File**.

   The source can be an ERP, an on-premises database, a file, or any source system.
6. **(Optional)**: If the source system in step 5 is file-based, from the **File Browser**, select the transactions file from which to load to the Reconciliation Compliance Balances application and click **OK**.

   You can double-click the **inbox** or **outbox** folders, or any other folders to see a list of files in folders.

   You can also click **Upload** and navigate to a file to upload it.

   Optionally, you can create a file-based load in which you designate the file to be loaded at runtime. However, the column number and column name are designated only when you map the dimensions. See Mapping Dimensions.

   You can select the following types of files.

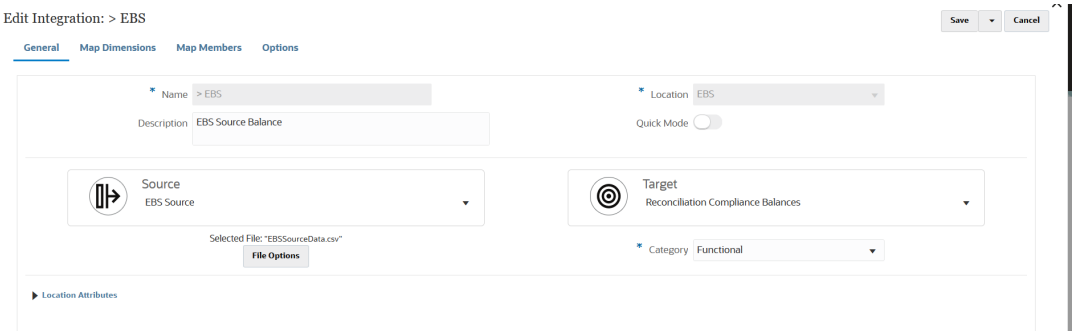
   - comma separated file (CSV)
   - tab separated file (TXT)
   - custom delimiter file
7. Click (Select a Target).

Copyright © 2017, 2026, Oracle and/or its affiliates.

8. From the **Select a Target** page, select the Reconciliation Compliance Balances target application.
9. From **Category**, select a currency bucket.

   Options include:

   - Entered
   - Functional
   - Reporting
10. Click **Save**.



See also: GL and Subledger Balances from a File using Data Integration

Mapping Dimensions for a Reconciliation Compliance Balances Application

By default "Profile" is mapped to the "Account" (Reconciliation Account ID) target dimension class and "Period" is mapped to the "Period" target dimension class.

To define dimension details for a Reconciliation Compliance Balances application:

1. From the **Data Integration** home page, click ⋯ to the right of the Reconciliation Compliance Balances integration, and then select **Map Dimensions**.
2. From **Import Format**, select the name of the import format to use for the integration.

   You can also add a user defined import format name.
3. From **Type**, select **Delimited**.

   Delimited enables you to load numeric data from a delimited file format.
4. From **Delimiter**, select the character to use for delimiting columns in the output file.

   A delimited file contains one or more records set off from each other by a specified delimiter.

   Available options:

   - Comma (,)
   - Pipe (|)
   - Exclamation (!)
   - Semicolon (;)
   - Colon (:)
   - Tab
   - Tilde (~)

Copyright © 2017, 2026, Oracle and/or its affiliates.

5. In the mappings grid, map the source columns in the source data-load file to the appropriate Reconciliation Compliance Balances dimensions.

   If the import format has already been defined for the file, then the source and target columns are mapped automatically.

6. Add the **Currency Dimension** to map the currency code.
7. **Optional**: Add additional **Profile** dimensions to concatenate Profile Segments in the Profile dimension.
8. Edit the source expression on the **Amount** column and add the **NZP** expression type to load zero balances.

   For more information about source expressions, see Using Source Expressions.

9. Click **Save**.

Adding an Account Reconciliation Merge Balance Dimension

When loading data, Account Reconciliation customers can merge only changed balances and retain existing data for the same location. This option eliminates the need to run an entire data file load when only a few balances have changed since the last time data was loaded into Account Reconciliation. The process requires that customers set up a merge ID dimension for each new balance to be loaded to the target application. All the individual dimensions that make up the profile segments along with currency should be mapped to the Merge ID.

> **Note**
>
> To ensure correct drill back results from Account Reconciliation to Data Integration, it is required to add a new merge ID for each new balance to load.

To map a merge balance dimension:

1. From the **Data Integration** home page, click to the right of the integration to which to add the merge balance, and then select **Map Dimensions**.
2. From the Map Dimensions, select the name of the import format associated with the integration.

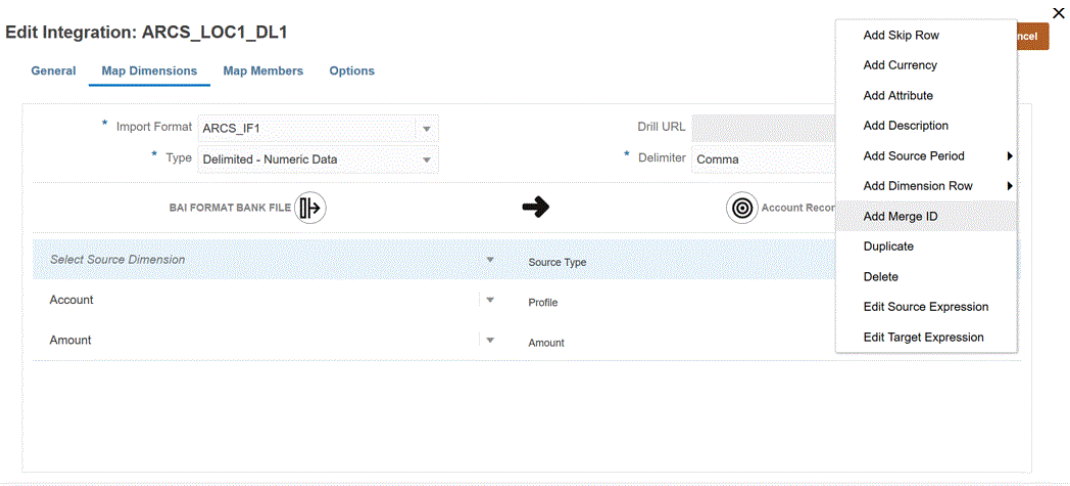
   You can also add a user defined import format name.

3. Click and click **Add Merge ID**.

   A blank source dimension row is added with a source type of **Merge ID**.

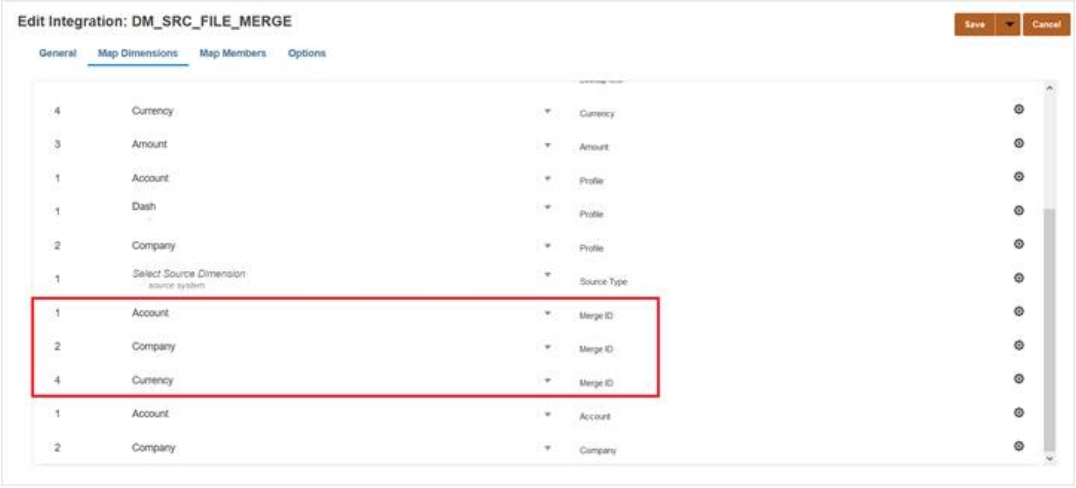
> **Note**
>
> Drill back to source for file-based sources and Fusion ERP GL is not supported if you use the Merge ID. You can only drill from Account Reconciliation to Data Integration.

Copyright © 2017, 2026, Oracle and/or its affiliates.



4. From the **Select Source Dimension** drop-down, map all individual dimensions that make up the profile segments along with currency to Merge ID.

   For example if Account and Company make up the profile segments, add the Merge ID row 3 times and map Merge ID for 3 rows in total Account, Company and then again for Currency.



5. Click **Save**.
6. From the **Data Integration** home page, select the integration, and then click ▶.
7. On the **Run Integration** page, select the **Options** tab.
8. From **Import Mode** drop-down, select **Merge**.
9. From **Export Mode** drop-down, select **No Export**.
10. Select periods as needed.
11. Click **Run**.

Mapping Members for Reconciliation Compliance Balances

To map members for Reconciliation Compliance balances, see Adding Member Mappings.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Running the Reconciliation Compliance Balances Integration

When you execute an integration in Data Integration that has a Reconciliation Compliance Balances target application, Data Integration imports data from the source and stages it.

Then Account Reconciliation pulls the data from Data Integration, processes the data and loads them to the appropriate reconciliations in the selected period.

To test an integration to a Reconciliation Compliance Balances target application:

1. From the **Data Integration** home page, select an Reconciliation Compliance Balances integration, and then click ▶.
2. From **Import Mode**, select the method for importing data.

   Available import modes:

   - **Append**—Keep existing rows for the POV but append new rows to the POV. For example, a first-time load has 100 rows and second load has 50 rows. In this case, 50 rows are appended. After this load, the row total for the POV is 150.
   - **Replace**—Clears all data for the POV in the target, and then loads from the source or file. For example, a first-time load has 100 rows, and a second load has 70 rows. In this case, 100 rows are removed, and 70 rows are loaded to TDATASSEG. After this load, the row total is 70.
   - **Merge**—(Account Reconciliation only). Merge changed balances with existing data for the same location.

     Merge mode eliminates the need to load an entire data file when only a few balances have changed since the last time data was loaded into Account Reconciliation. If mappings change between two loads, customers must reload the full data set.

     For example, a customer might have 100 rows of existing balances for one number account IDs, each which has an amount of $100.00. If the customer runs the integration in merge mode and the source has one row for one account ID with an amount of $80, then after running integration, there are one hundred rows of balances, 99 each of which have a balance of $100.00, and 1 which has a balance of $80.00.

     > **Note**
     >
     > Merge mode for Account Reconciliation is applicable for:
     >
     > a. File based sources only
     > b. Reconciliation Compliance Balances Target application only.

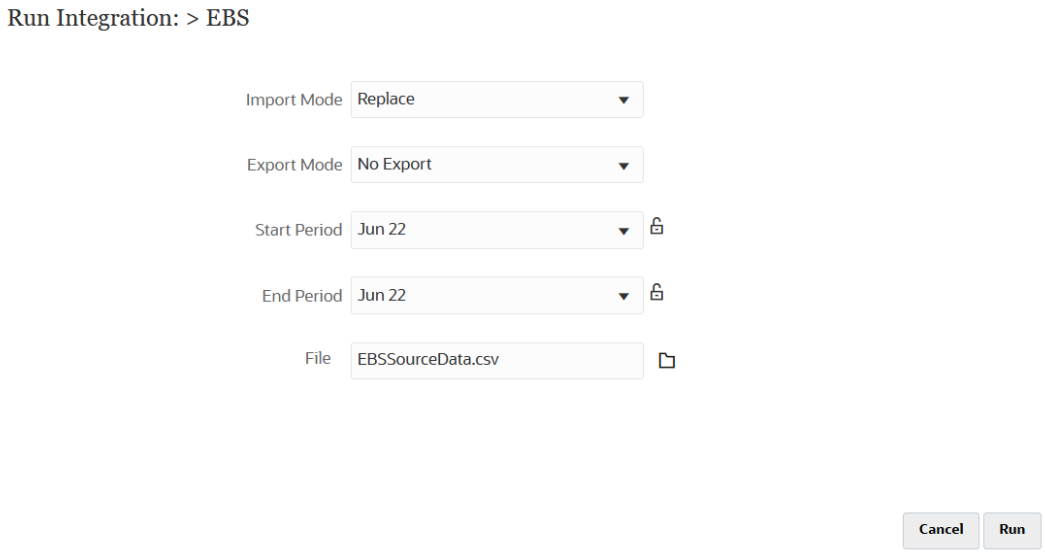
   - **No Import**—Skip the import of data entirely.
   - **Map and Validate**—Skip importing the data but reprocess the data with updated mappings.
3. If the source system is file based, in **File Name**, select the data file name that contains the data you are loading.

   When the file name is provided, then data must be entered for a single period on the Rules Execution window.

   To navigate to a file, click 🗀.

Copyright © 2017, 2026, Oracle and/or its affiliates.

4. From **Start Period** and **End Period**, select the period defined for Reconciliation Compliance Balances.
5. Click **Run**.



6. To import balances into Account Reconciliation, complete the following:
   a. Click **Application**, then **Configuration**, and then select **Data Load**.
   b. Create a Data Load definition.
   c. Import balances by selecting **Application** and then **Period**.
   d. From the **Action** menu, click **Import Data**.
   e. Click , then from **New Data Load Execution**, select **Use saved data load**, and select the data load created in the previous step.
   f. Click **OK** to import balances.

      For more information, see Defining and Saving a Data Load Definition in *Administering Oracle Account Reconciliation*.

Integrating BAI, SWIFT MT940, and CAMT.053 Format Bank File Balances

When loading Bank balances, you create a data source associated with a Bank file source system. Data Integration pulls each account's end of day balance from the BAI, SWIFT MT940, and CAMT.053 file formats so they can be loaded as a sub-system balance in Account Reconciliation.

The source application for a BAI Format Bank File Balance file has the following pre-defined constant columns and headers:

- Closing Balance
- Currency
- Transaction Type
- Currency
- Statement Date
- Account

Copyright © 2017, 2026, Oracle and/or its affiliates.

The source application for a Swift MT940 Format Bank File Balances file has the following pre-defined constant columns and headers:

- Closing Balance
- Currency
- Transaction Type
- Currency
- Statement Date
- Account

The source application for a CAMT.053 Format Bank File Balances file has the following pre-defined constant columns and headers:

- Account
- Closing Balance
- Currency
- As Of Date
- CdtDbtInd (Credit Debit Indicator)

To add a BAI Format Bank File, SWIFT MT940, or CAMT.053 Format Bank File Balances source system:

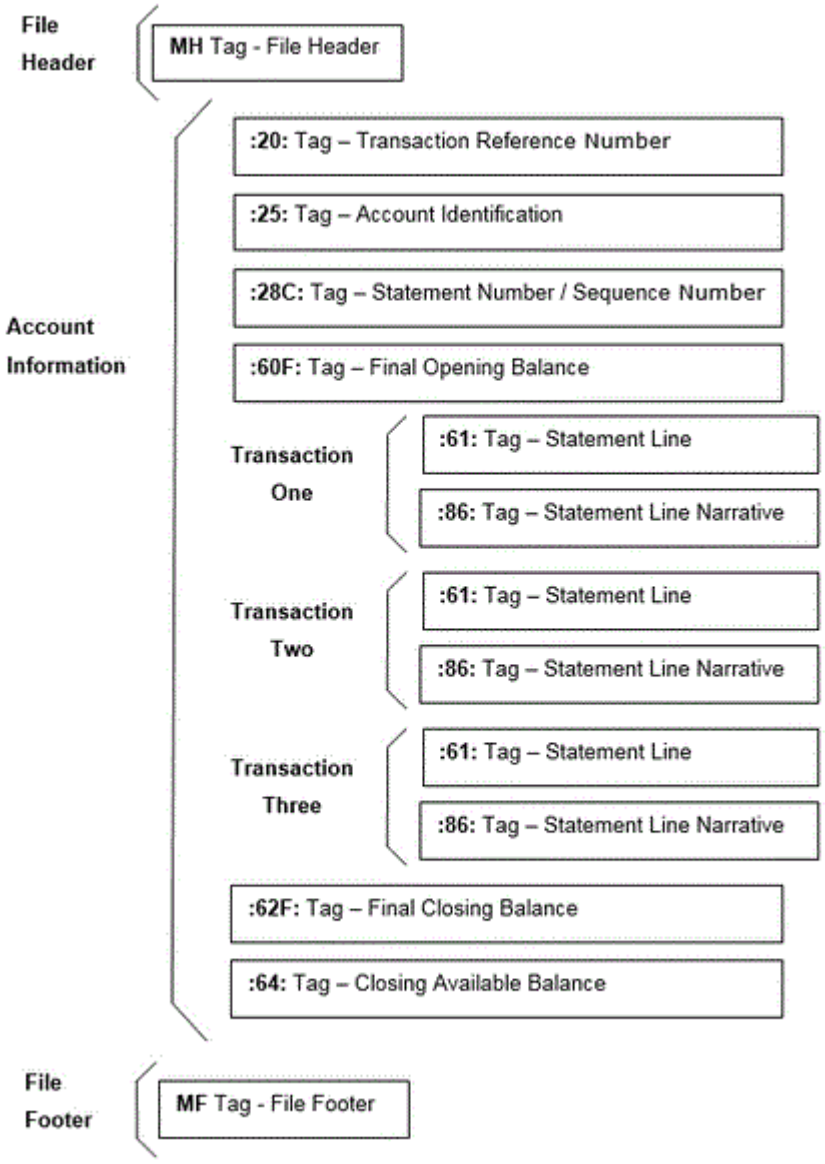
1. Upload the BAI, SWIFT MT940 or camt.053 bank statement files using the file browser when registering the target application.

   The following shows a BAI Format Bank File statement file:

```
01,JSLTEST,JSLTEST,150828,1353,004,80,,2/
02,JSLTEST,JSLTEST,1,150827,1435,USD,2/
03,9000000009124,USD,010,00000006850000,,/
88,015,00000008430000,,/
88,100,00000004190000,,/
88,400,00000002610000,,/
16,354,004000000,,KB0075 IMG8650153019,,Int Inc
16,654,000100000,,KB0075 IMG8650153019,,Int exp
16,116,100000,,KB0075 IMG8650153019,1800000040,Check(s) In,/
88,rcode056,invoice number, text info,/
16,116,90000,,KB0075 IMG8650153019,1800000041,Check(s) In,/
16,698,0010000,,KB0075 IMG8650153019,,Bank Fee
16,475,1000000,,KB0075 IMG8650153019,100007,Check(s) Paid
16,475,1100000,,KB0075 IMG8650153019,100008,Check(s) Paid
16,469,4000000,,20150840796815,/
49,0000000000010000,000007/
98,0000000000010000,00001,000009/
99,0000000000010000,00001,000011/
```

   The following shows the structure of a SWIFT MT940 Format Bank File statement file:

Copyright © 2017, 2026, Oracle and/or its affiliates.

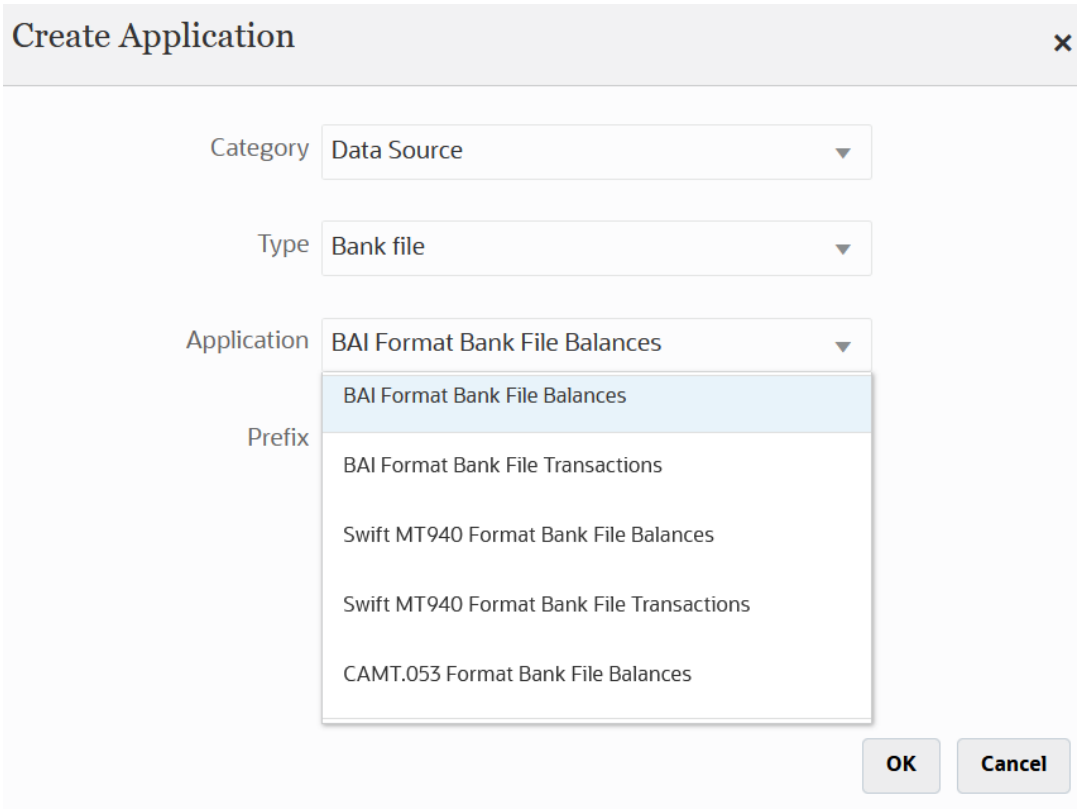


2. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
3. On the **Applications** page, click + (Add icon).
4. From **Category**, select **Data Source**.
5. From **Type**, select **Bank file**.
6. From **Application**, select an application name from the list of values.

   Available types of applications include:

   - BAI Format Bank File Balances
   - SWIFT MT940 Format Bank File Balances
   - CAMT.053 Format Bank File Balances

Copyright © 2017, 2026, Oracle and/or its affiliates.



For a BAI Format Bank File Balances, select **BAI Format Bank File Balances**.

For a SWIFT MT940 file, select **SWIFT MT940 Format Bank File Balances**.

For a CAMT.053 file, select **CAMT.053 Format Bank File Balances**.

7. In **Prefix**, specify a prefix to make the source system name unique.

   Use a prefix when the source system name you want to add is based on an existing source system name. The prefix is joined to the existing name. For example, if you want to name a Bank file application the same name as the existing one, you might assign your initials as the prefix.

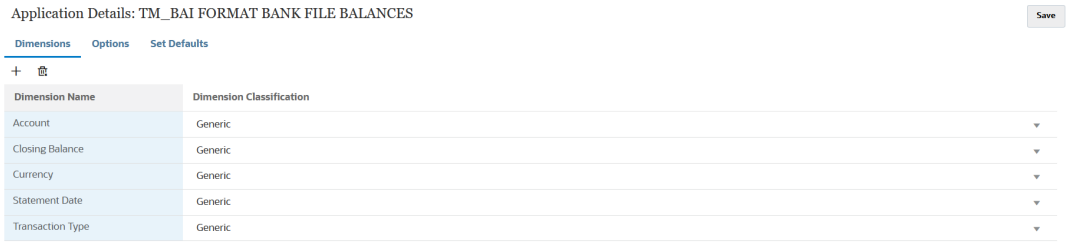
8. Click **OK**.

9. From the **Application** page, click ⋯ to the right of the application, and then select **Application Details**.

10. Select the **Dimensions** tab.

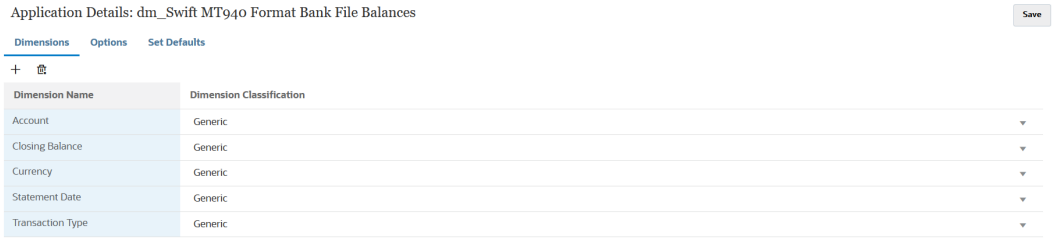
11. To view dimensions in the BAI Format Bank File Balances file source system, select the **Dimension Details** tab.

    The dimension details for a BAI Format Bank File Balances file application are shown below:

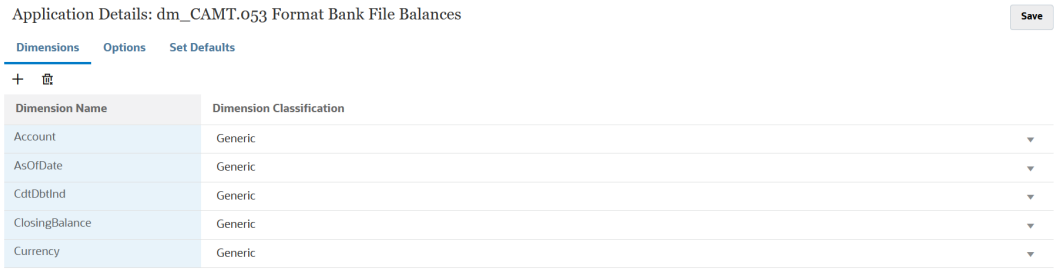


Copyright © 2017, 2026, Oracle and/or its affiliates.

The dimension details for a SWIFT MT940 Format Bank File Balances application are shown below:



The dimension details for a CAMT.053 Format Bank File Balances application are shown below:



12. Set up the integration mapping between BAI Format Bank File Balances source system and the Reconciliation Compliance Balances target application by building an import format:

    a. On the **Create Integration** page, then **Name** and **Description**, enter a name and description for the new integration.

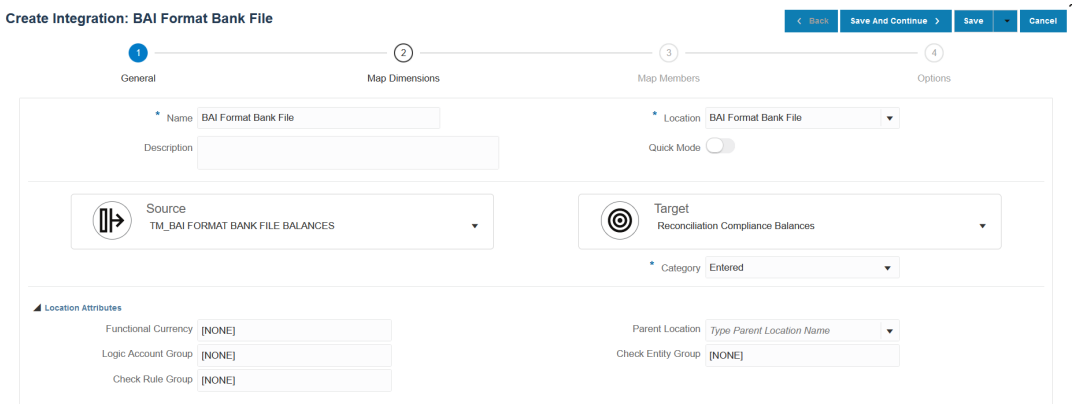
    b. In **Location**, enter a new location name, or pick an existing location to specify where to load data.

    c. From the **Source** ( ) drop-down, select the Bank File source application.

    d. From the **Target** ( ) drop-down, select the target Reconciliation Compliance Balances application.

    e. From **Category**, select the currency bucket.

    f. **Optional**: Select any applicable location attributes for the integration. For more information, see Selecting Location Attributes.



Copyright © 2017, 2026, Oracle and/or its affiliates.

13. From the **Data Integration** home page, click ⋯ and then select **Map Dimensions**.
14. On the **Map Dimensions** page, set up the integration mapping between BAI Format Bank File Balances source system and the Reconciliation Compliances Balances target application by building an import format.
    a. In **Import Format**, select the name of the import format to use for the integration.

       You can also add a user-defined import format name.

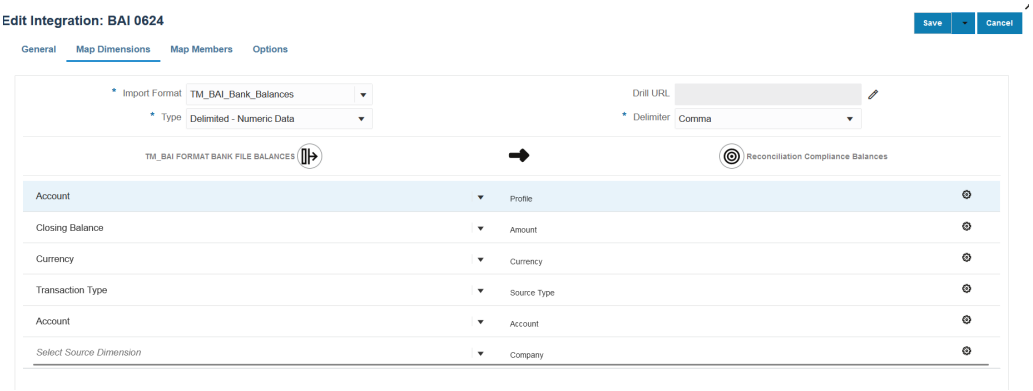
    b. In the mappings grid, map the source columns in the source to the dimensions in the Reconciliation Compliance Balances Profiles.

> **Note**
>
> All Bank files require that the Profile dimension is mapped to the corresponding Profile in Account Reconciliation.

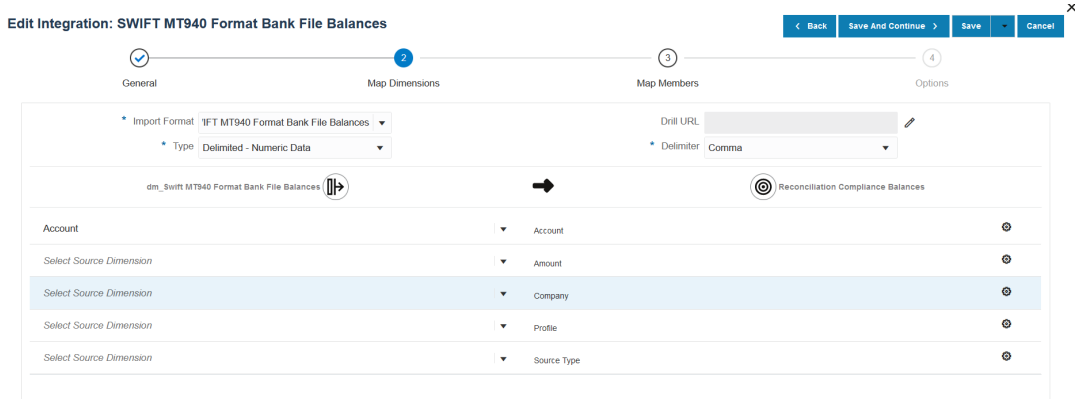
The dimensions from the target application are populated automatically.

If the import format has already been defined for the integration, then the source and target columns are mapped automatically.

An example of the import format for a BAI Format Bank File Balances application is shown below:

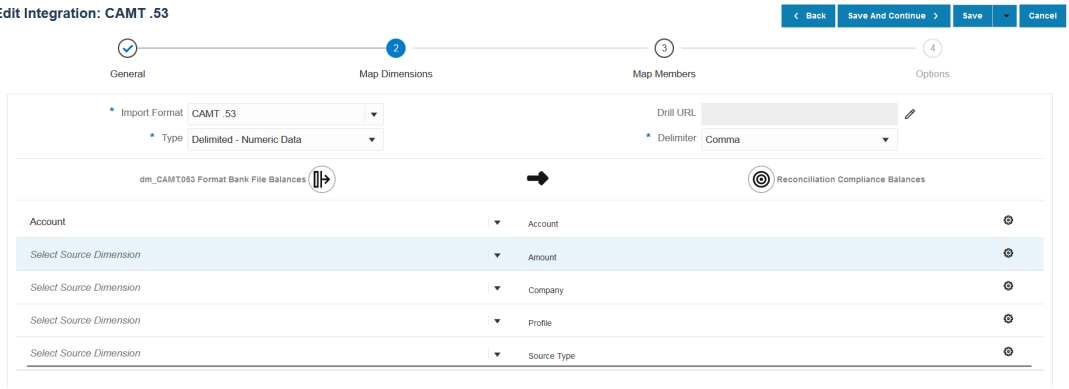


An example of the import format for a SWIFT MT940 Format Bank File Balances application is shown below:

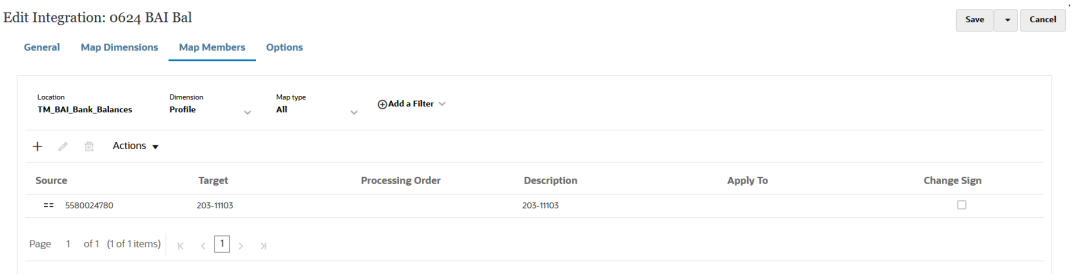


An example of the import format for a CAMT .053 Format Bank File Balances application is shown below:

Copyright © 2017, 2026, Oracle and/or its affiliates.



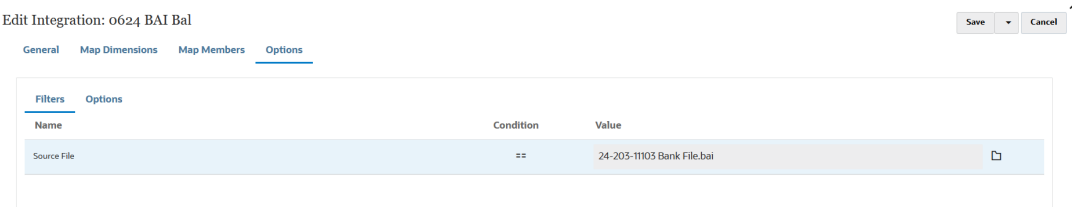
15. From the **Data Integration** home page, click and then select **Map Members**.
16. On the **Map Members** page, map the account numbers in the file to the appropriate Reconciliation names.



> **Note**
>
> All Bank files require that the Profile dimension is mapped to the corresponding Profile in Account Reconciliation.

17. From the **Data Integration** home page, click to the right of the integration, and then select **Options**.
18. Select the **Filter** tab.
19. Map the source type dimension **Source type *** to the hard coded "source system" or "sub-system" target value and then click **Save**.

> **Note**
>
> Bank Files are usually configured as sub-systems.



20. Click the **Options** tab, in **Category**, specify the preferred category.

Copyright © 2017, 2026, Oracle and/or its affiliates.

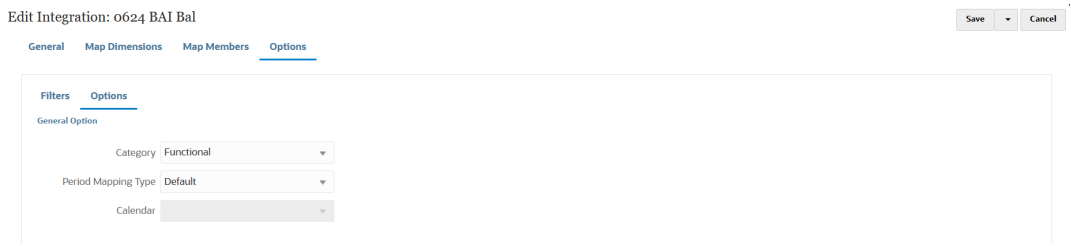
Default categories include:

- Reporting
- Functional
- Entered

Typically bank balances would be loaded into "Entered," but depending on your requirements, you can select "Functional" or "Reporting."

21. In **Period Mapping Type**, select **Default**

    When **Default** is selected, period mappings default to the list of source application periods using the application or global period mappings based on the period key. The list of source periods is added as Year and Period filters.



22. Click **Save**.
23. Go to Account Reconciliation home page.
24. Click **Application**, then **Configuration**, and then **Data Load**.
25. Create a data load definition.

    For more information, seeDefining and Saving a Data Load Definition in *Administering Oracle Account Reconciliation* .

26. Import balances by selecting **Application** and then **Period**.
27. Click to go to the action menu and then click **Import Data**.
28. Click **+**, then from **New Data Load Execution**, select **Use saved data load**, and select the data load created in the previous step.
29. Click **OK** to import balances from Bank Files.

    Bank balances are typically loaded as sub-system balances, but can also be loaded as source balances if needed.

Adding a Transaction Matching Target Application

Export a file to a Transaction Matching Target application.

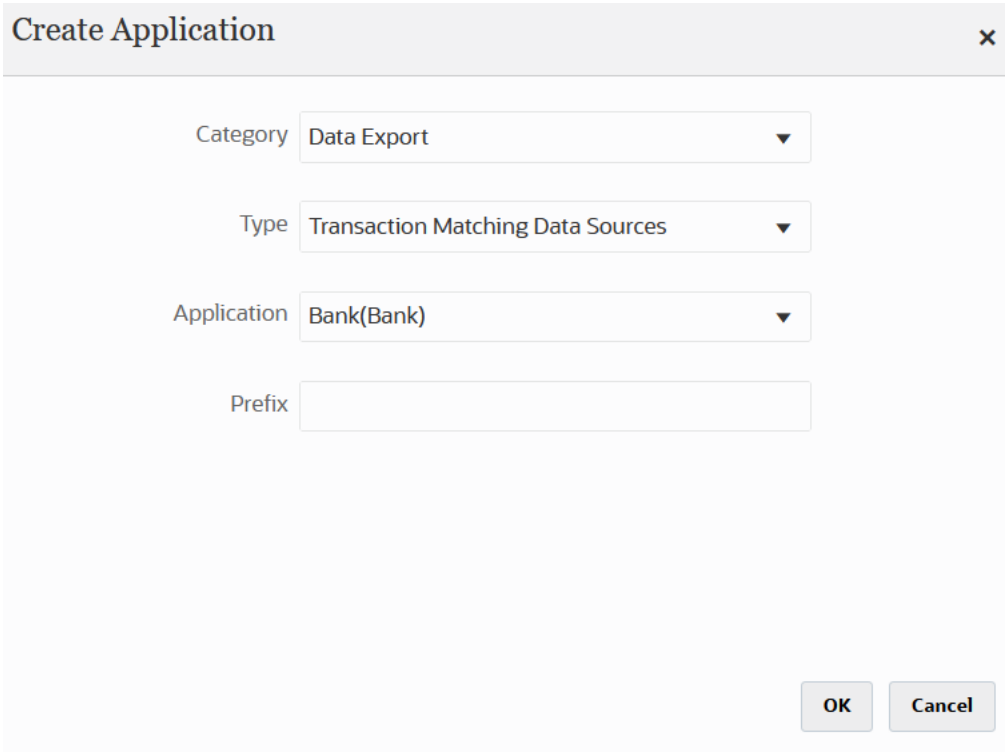
Any file, including a Bank file, can easily be exported to a Transaction Matching Target application. In this case, you create a data export application, which uses a Transaction Matching data source as the application type, and then you select the application name.

To add a Transaction Matching target application:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, click (Add icon).
3. From **Category**, select **Data Export**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

4. From **Type**, select **Transaction Matching Data Sources**.
5. From **Application**, select the application name for the Transaction Matching data source.
6. Click **OK**.



7. To view dimensions in the Transaction Matching target application, select the **Dimension Details** tab.

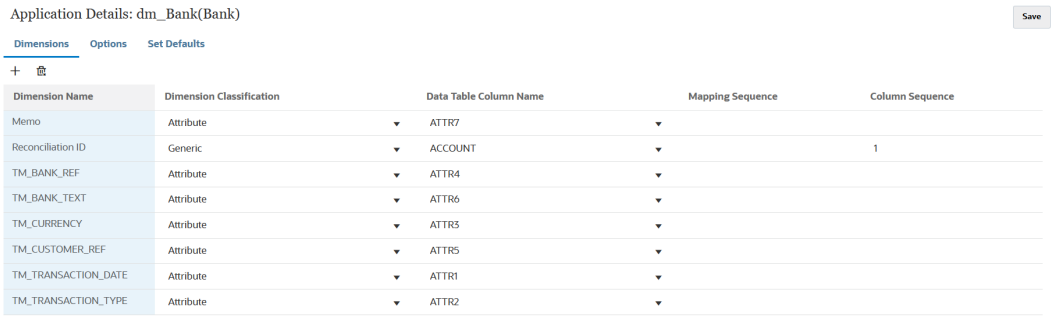
   When you a Transaction Matching target application, the dimension details of the application are populated automatically on the Dimension Detail tab. Subsequently, when you update the data source in Account Reconciliation, the dimension details are updated automatically on the **Dimension Details** tab. By default, the currency attribute defined in the data source in Account Reconciliation appears as a Dimension.

   > **Note**
   >
   > Data source attribute updates made by Service Administrators and users with the **Match Types - Manage granular role** automatically update the dimension details of the Transaction Matching target application in Data Integration. Verify the integrations when making updates to the associated data source in Transaction Matching.

   Each dimension class (except Reconciliation ID) is assigned the Attribute Dimension class. Attribute dimensions are pass-through, therefore no mappings are required. If you need to apply transformation, switch the Dimension Classification to Generic and select a UD or User Defined from the Data Table Column Name.

   The dimension class is a property that is defined by the dimension type.

Copyright © 2017, 2026, Oracle and/or its affiliates.



8. Click **Save**.
9. Set up the source and target mapping between the source system and the Transaction Matching target application by building an import format.
    a. On the **Create Integration** page, then **Name** and **Description**, enter a name and description for the new integration.
    b. In **Location**, enter a new location name, or pick an existing location to specify where to load data.
    c. From the **Source** ( ) drop-down, select the source application.
    d. From the **Target** ( ) drop-down, select the Transaction Matching target application.
    e. From **Category**, select a currency bucket.

    > **Note**
    >
    > Category Mappings are not applicable while loading data into Transaction Matching Targets. However, as a best practice, choose a currency bucket that aligns with the lowest enabled currency bucket on your profiles.

    For information on Account Reconciliation currency buckets, see Defining Currency in the *Setting Up and Configuring Account Reconciliation*.

    f. **Optional**: Select any applicable location attributes for the integration. For more information, see Selecting Location Attributes.
10. From the **Data Integration** home page, click ... to the right of the integration, and then select **Map Dimensions**.
11. On the **Map Dimensions** page, complete the following:
    a. In **Import Format**, select the name of the import format to use for the integration.

    You can also add a user-defined import format name.
    b. In the mappings grid, map the source columns in the source to the dimensions in the target application.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> All transaction matching files require the Reconciliation Id dimension to be mapped to the corresponding Transaction Matching Profile.

The dimensions from the target application are populated automatically. Do not the map the currency attribute from the source to the default Currency dimension in Data Integration. Instead, map it to the dimension that is associated with the data source's currency attribute (defined in Step 7).

If the import format has already been defined for the integration, then the source and target columns are mapped automatically.

12. From the **Data Integration** home page, click to the right of the integration, and then select **Map Members**.
13. On the **Map Members** page, define members mapping to map the members from the source to target.

> **Note**
>
> All transaction matching files require the Reconciliation Id dimension to be mapped to the corresponding Transaction Matching Profile.

14. From the **Data Integration** home page, click to the right of the integration, and then select **Workbench**.
15. In the **Workbench**, test and validate the data by executing the integration to ensure that the load is running properly, and your data looks correct. Data Integration transforms the data and stages it for Account Reconciliation to use. For more information about the impact of importing transactions with associated currencies, see "Importing Transactions" in Importing Pre-mapped Transactions for Transaction Matching in *Administering Account Reconciliation* .

Aggregating Transaction Matching Data

Data Integration enables users to aggregate transactions for Transaction Matching. For example, using this aggregation option, you can load transactions at a detail credit card swipe level and roll them up to a summary total per location per day total for matching.

To set the Aggregation option:

1. Add a new Transaction Matching target application.

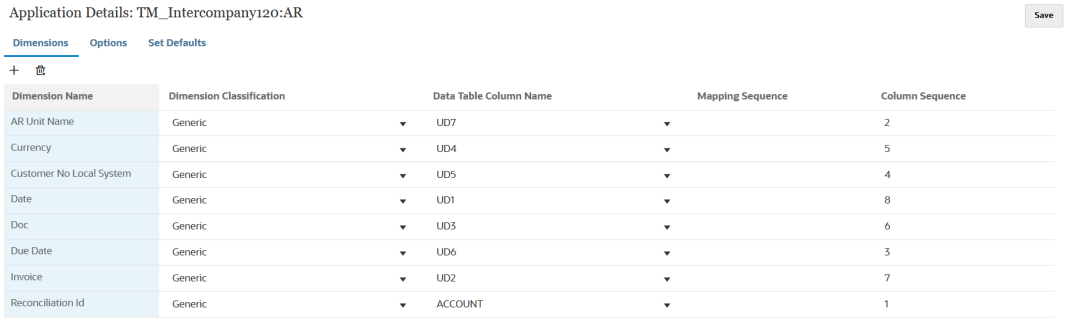
   For information on adding a Transaction Matching target application, see Adding a Transaction Matching Target Application.
2. Select the **Dimension Detail** tab.

   When you select a Transaction Matching target application, the dimension details of the application are populated automatically on the Dimension Detail tab.
3. Select the **Dimension Class** or click to select the target dimension class.

   The dimension class is a property that is defined by the dimension type.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Include only those dimensions that you want to aggregate when mappings dimensions. For example, if you want to roll up only the merchant number, bank reference, credit card type, or transaction date, include only these corresponding dimensions in your mappings.



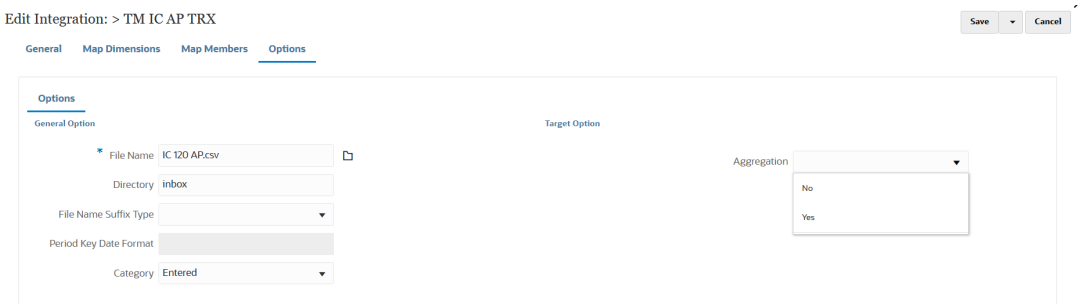
4. Click **Save**.

5. From the Data Integration home page, click to the right of the integration with the Transaction Matching data to aggregate, and then from the Edit Integration page, select the **Options** tab.

6. From **Aggregation**, select **Y** to aggregate imported transactions.

   To leave imported transaction unaggregated, select **N**.

   The default aggregation setting is **N**.



   You can also enable the aggregation option by selecting **Application Detail**, then from **Options**, then **Aggregation** and then from **Property Value**, select **Y** (for yes).

   The Aggregate option chosen on the Options tab in step 6 overrides the option chosen in the Application Details.

7. Click **Save**.

8. In the Workbench, test and validate the data by executing the data load rule to ensure that the data load rule is running properly, and your data looks correct. Data Integration transforms the data and stages it for Account Reconciliation to use.

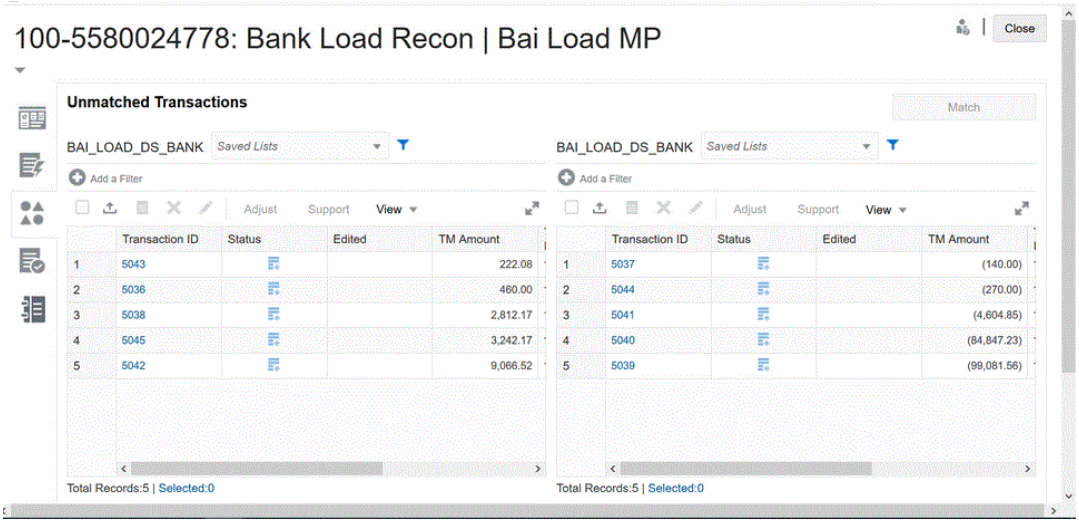
9. To view the Transaction Matching with the aggregated imported data, from the **Accounts Reconciliation** home page, click **Matching**.

10. Click the **Account Id** to which the source accounts were mapped.

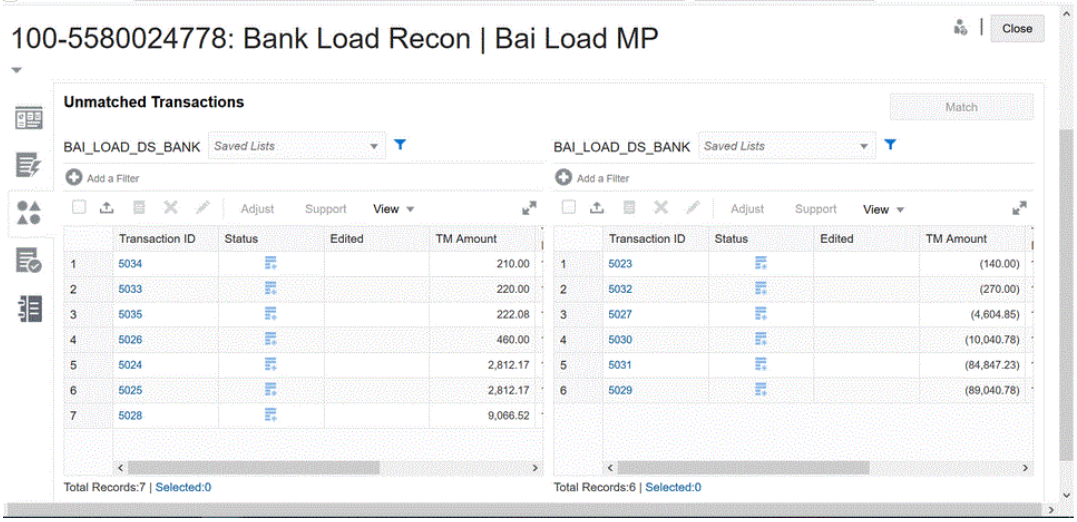
11. Click the **Unmatched Transactions** icon ( )

    In the example, ten transactions are loaded here when aggregation is enabled:

Copyright © 2017, 2026, Oracle and/or its affiliates.



If the aggregation option is disabled, a total of thirteen transactions are loaded:



Integrating BAI, SWIFT MT940, and CAMT.053 Format Bank File Transactions

Converts the BAI, SWIFT MT940, or CAMT.053 file formats to CSV format for loading into Transaction Matching.

When loading bank file data, you create a data source associated with the bank file source system. Data Integration converts the BAI, SWIFT MT940, or CAMT.053 file formats to CSV format for loading into Transaction Matching. The CSV load file can be viewed in the Transaction Matching Jobs results.

The source application for BAI Format Bank File Transactions has the following pre-defined constant columns and headers:

- Account
- Amount
- Transaction Type
- Currency

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Transaction Date
- Bank Reference
- Customer Reference
- Bank Text
- Bank Text2, Bank Text3, and Bank Text4
- FundsType1, FundsType2 and FundsType3: If Funds Type is V (date equal value), then copies the following two date fields into FundsType1 and FundsType2 - as is from the , Optionally, if Funds Type is S, then copy the next three fields into FundsType1, FundsType2, and FundsType3 - as is.

The source application for a Swift MT940 Format Bank File Transactions has the following pre-defined constant fields:

Table 17-3 Swift MT940 Fields and Descriptions

Field	Description
Transaction Reference Number	
StatementIdentification	Unique identification to unambiguously identify the account statement.
CreateDate	Date on which the statement was created.
StatementFromDate	Date on which the period starts, for which the account statement is issued.
StatementToDate	Date on which the period ends, for which the account statement is issued.
Account	Account Unambiguous identification of the account to which credit and debit entries are made.
Amount	Amount of money in the cash entry.
Currency	Currency code of the currency in which the cash entry was made.
Closing Balance	
Currency	
Transaction Date	
Transaction Type	
Currency	
Statement Date	
Statement Number	
Amount	
Customer Reference	
Bank Reference	
Bank Text	
Bank Text2, Bank Text3, and Bank Text4	When using BAI format files, the Bank Text column above supports a maximum of 300 characters. If your description in the Bank Text column exceeds more than 300 characters, the Bank Text1, Bank Text2, and Bank Text 3 columns are used to store additional text up to a maximum of 1,200 characters for all Bank Text columns (Bank Text, Bank Text1, Bank Text2, and Bank Text 3).
FundsType1, FundsType2 and FundsType3	If Funds Type is V (date equal value), then copies the next two date fields into FundsType1 and FundsType2 - as is, Optionally, if Funds Type is S, then copy the following three fields into FundsType1, FundsType2, and FundsType3 - as is.
Additional Info1	

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 17-3 (Cont.) Swift MT940 Fields and Descriptions

Field	Description
Additional Info2	
Additional Info3	

The source application for CAMT.053 Format Bank File Transactions file has the following pre-defined constant columns and headers:

Table 17-4 CAMT.053 Format Bank Transaction Field Descriptions

Field	Description (based on ISO definitions)	CAMT.053 TAG
StatementIdentification	Unique identification to unambiguously identify the account statement.	<Stmt> <Id>
CreateDate	Date at which the statement was created.	<Stmt> <CreDtTm>
StatementFromDate	Date at which the period starts, for which the account statement is issued.	<Stmt> <FrToDt> <FrDtTm>
StatementToDate	Date which the period ends, for which the account statement is issued.	<Stmt> <FrToDt> <ToDtTm>
Account	Unambiguous identification of the account to which credit and debit entries are made.	<Stmt> <Acct> <Id> <Othr> <Id> or <Stmt> <Acct> <Id> <IBAN>
AccountOwner	Party that legally owns the account	<Stmt> <Acct> <Ownr>
Amount	Amount of money in the cash entry.	<Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <Amt Ccy="ISO Currency Code"> or <Stmt> <Acct> <Ntry> <Amt Ccy="ISO Currency Code">
Currency	Currency code of the currency in which the cash entry was made.	<Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <Amt Ccy="ISO Currency Code"> or <Stmt> <Acct> <Ntry> <Amt Ccy="ISO Currency Code">
CdtDbtInd	Credit Debit Indicator	<Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <NtryDtls> <TxDtls> <CdtDbtInd> or <Stmt> <Acct> <Ntry> <CdtDbtInd> Except for batch bookings, the <TxDtls> tag is mandatory. The bank statement may not contain detailed transaction information for batch bookings because of the sensitive nature of the data.
BookingDate	Date and time when an entry is posted to an account on the account servicer's books.	<Stmt> <Acct> <Ntry> <BookgDt> <Dt>

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 17-4 (Cont.) CAMT.053 Format Bank Transaction Field Descriptions

Field	Description (based on ISO definitions)	CAMT.053 TAG
ValueDate	Date and time at which assets become available to the account owner in case of a credit entry, or cease to be available to the account owner in case of a debit entry.	<Stmt> <Acct> <Ntry> <ValDt> <Dt>
EntryRef	Unique reference for the entry	<Stmt> <Acct> <Ntry> <NtryRef>
AccountServicerReference	Unique reference as assigned by the account servicing institution to unambiguously identify the entry	<Stmt> <Acct> <Ntry> <AcctSvcrRef> or <Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <Refs> <AcctSvcrRef>
DomainCode	Part of Bank Transaction Code. It specifies the business area of the underlying transaction.	<Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <BkTxCd> <Domn> <Cd> or <Stmt> <Acct> <Ntry> <BkTxCd> <Domn> <Cd>
FamilyCode	Specifies the family within a domain.	<Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <BkTxCd> <Domn> <Fmly> <Cd> or <Stmt> <Acct> <Ntry> <BkTxCd> <Domn> <Fmly> <Cd>
SubFamilyCode	Specifies the sub-product family within a specific family.	<Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <BkTxCd> <Domn> <Fmly> <SubFmlyCd> or <Stmt> <Acct> <Ntry> <BkTxCd> <Domn> <Fmly> <SubFmlyCd>
Proprietary	Bank transaction code in a proprietary form, as defined by the issuer.	<Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <BkTxCd> <Prtry> <cd> or <Stmt> <Acct> <Ntry> <BkTxCd> <Prtry> <cd>
ReversalIndicator	Indicates whether or not the entry is the result of a reversal. This element should only be present if the entry is the result of a reversal.	<Stmt> <Acct> <Ntry> <RvslInd>
BankTransactionIdentification	Unique identification that can be used for reconciliation, tracking or to link tasks relating to the transaction on the interbank level.	<Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <Refs> <TxId>
EndToEndIdentification	Unique identification, as assigned by the initiating party, to unambiguously identify the transaction.	<Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <Refs> <EndToEndId>
ChequeNumber	Unique and unambiguous identifier for a cheque as assigned by the agent.	<Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <Refs> <ChqNb>
InstructionIdentification	Unique identification, which is a point to point reference that can be used between the instructing party and the instructed party to refer to the individual instruction.	<Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <Refs> <InstrId>

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 17-4 (Cont.) CAMT.053 Format Bank Transaction Field Descriptions

Field	Description (based on ISO definitions)	CAMT.053 TAG
PaymentInformationIdentification	Unique identification, as assigned by a sending party, to unambiguously identify the payment information group within the message.	<Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <Refs> <PmtInfId>
MandateIdentification	Unique identification, as assigned by the creditor, to unambiguously identify the mandate.	<Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <Refs> <MndtId>
ClearingSystemReference	Unique reference, as assigned by a clearing system, to unambiguously identify the instruction.	<Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <Refs> <ClrSysRef>
SourceCurrency	Currency from which an amount is to be converted in a currency conversion.	<Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <AmtDtls> <TxAmt> <CcyXchg> <SrcCcy> or <Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <AmtDtls> <InstdAmt> <CcyXchg> <SrcCcy>
TargetCurrency	Currency into which an amount is to be converted in a currency conversion.	<Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <AmtDtls> <TxAmt> <CcyXchg> <TrgtCcy> or <Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <AmtDtls> <InstdAmt> <CcyXchg> <TrgtCcy>
UnitCurrency	Currency in which the rate of exchange is expressed in a currency exchange. In the example, 1GBP = xxxCUR, the unit currency is GBP.	<Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <AmtDtls> <TxAmt> <CcyXchg> <UnitCcy> or <Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <AmtDtls> <InstdAmt> <CcyXchg> <UnitCcy>
ExchangeRate	Factor used to convert an amount from one currency into another. This reflects the price at which one currency was bought with another currency. Usage: ExchangeRate expresses the ratio between UnitCurrency and QuotedCurrency (ExchangeRate = UnitCurrency/ QuotedCurrency).	<Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <AmtDtls> <TxAmt> <CcyXchg> <XchgRate> or <Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <AmtDtls> <InstdAmt> <CcyXchg> <XchgRate>
ContractIdentification	Unique identification to unambiguously identify the foreign exchange contract	<Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <AmtDtls> <TxAmt> <CcyXchg> <CtrctId> or <Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <AmtDtls> <InstdAmt> <CcyXchg> <CtrctId>
QuotationDate	Date and time at which an exchange rate is quoted.	<Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <AmtDtls> <TxAmt> <CcyXchg> <QtnDt> or <Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <AmtDtls> <InstdAmt> <CcyXchg> <QtnDt>

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 17-4 (Cont.) CAMT.053 Format Bank Transaction Field Descriptions

Field	Description (based on ISO definitions)	CAMT.053 TAG
ReturnReasonCode	Specifies the reason for the return, as per SWIFT external codes.	<Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <RtrInf> <Rsn> <Code>
ReturnReasonProprietary	Specifies the reason for the return, in a proprietary form.	<Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <RtrInf> <Rsn> <Prtry>
Originator	Name of the party that issues the return.	<Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <RtrInf> <Orgtr> <Nm>
OrgnlBkTxCdDomain	Part of the Original Bank Transaction Code. It specifies the business area of the underlying transaction.	<Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <RtrInf> <OrgnlBkTxCd> <Domn> <Cd>
OrgnlBkTxCdFamily	Specifies the family within a domain of the Original Bank Transaction Code.	<Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <RtrInf> <OrgnlBkTxCd> <Domn> <Fmly> <Cd>
OrgnlBkTxCdSubFamily	Specifies the sub-product family within a specific family of the Original Bank Transaction Code.	<Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <RtrInf> <OrgnlBkTxCd> <Domn> <Fmly> <SubFmlyCd>
OrgnlBkTxCdProprietary	Original Bank transaction code in a proprietary form, as defined by the issuer	<Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <RtrInf> <OrgnlBkTxCd> <Prtry> <Cd>
InitiatingParty	Party that initiated the payment that is reported in the entry.	<Stmt> <Ntry> <NtryDtls> <TxDtls> <RltdPties> <InitgPty> <Pty> <Nm>
Creditor	Party to which an amount of money is due	<Stmt> <Ntry> <NtryDtls> <TxDtls> <RltdPties> <Cdtr> <Pty> <Nm>
CreditorAccount	Unambiguous identification of the account of the creditor to which a credit entry has been posted as a result of the payment transaction	<Stmt> <Ntry> <NtryDtls> <TxDtls> <RltdPties> <CdtrAcct> <Id> <IBAN> or <Stmt> <Ntry> <NtryDtls> <TxDtls> <RltdPties> <CdtrAcct> <Id> <Othr> <Id>
UltimateCreditor	Ultimate party to which an amount of money is due	<Stmt> <Ntry> <NtryDtls> <TxDtls> <RltdPties> <UltmtCdtr> <Pty> <Nm>
Debtor	Party that owes an amount of money to the (ultimate) creditor.	<Stmt> <Ntry> <NtryDtls> <TxDtls> <RltdPties> <Dbtr> <Pty> <Nm>
DebtorAccount	Unambiguous identification of the account of the debtor	<Stmt> <Ntry> <NtryDtls> <TxDtls> <RltdPties> <DbtrAcct> <Id> <IBAN> or <Stmt> <Ntry> <NtryDtls> <TxDtls> <RltdPties> <DbtrAcct> <Id> <Othr> <id>
UltimateDebtor	Ultimate party that owes an amount of money to the (ultimate) creditor	<Stmt> <Ntry> <NtryDtls> <TxDtls> <RltdPties> <UltmtDbtr> <Pty> <Nm>
TradingParty	Party that plays an active role in planning and executing the transactions that create or liquidate investments of the investors assets, or that move the investor's assets from one investment to another.	<Stmt> <Ntry> <NtryDtls> <TxDtls> <RltdPties> <TradgPty> <Pty> <Nm>

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 17-4 (Cont.) CAMT.053 Format Bank Transaction Field Descriptions

Field	Description (based on ISO definitions)	CAMT.053 TAG
RemittanceInformation	Information supplied to enable the matching/reconciliation of an entry with the items that the payment is intended to settle, such as commercial invoices in an accounts receivable system, in an unstructured form.	<Stmt> <Acct> <Ntry> <NtryDtls> <TxDtls> <RmtInf> <Ustrd>
AdditionalRemittanceInformation	Additional information, in free text form, to complement the structured remittance information	<Stmt> <Ntry> <NtryDtls> <TxDtls> <RmtInf> <Strd> <AddtlRmtInf>
CreditorReferenceInformation	Reference information provided by the creditor to allow the identification of the underlying documents.	<Stmt> <Ntry> <NtryDtls> <TxDtls> <RmtInf> <Strd> <CdtrRefInf> <Ref>
CreditorReferenceInformationCode	Type of creditor reference, in a coded form	<Stmt> <Ntry> <NtryDtls> <TxDtls> <RmtInf> <Strd> <CdtrRefInf> <Tp> <CdOrPrtry> <Cd>
CreditorReferenceInformationPrtry	Creditor reference type, in a proprietary form	<Stmt> <Ntry> <NtryDtls> <TxDtls> <RmtInf> <Strd> <CdtrRefInf> <Tp> <CdOrPrtry> <Prtry>
Invoicee	Identification of the party to whom an invoice is issued, when it is different from the debtor or ultimate debtor	<Stmt> <Ntry> <NtryDtls> <TxDtls> <RmtInf> <Strd> <Invcee> <Nm>
Invoicer	Identification of the organisation issuing the invoice, when it is different from the creditor or ultimate creditor	<Stmt> <Ntry> <NtryDtls> <TxDtls> <RmtInf> <Strd> <Invcr> <Nm>
ReferredDocumentInformation	Provides the identification and the content of the referred document	<Stmt> <Ntry> <NtryDtls> <TxDtls> <RmtInf> <Strd> <RfrdDocInf> <Nb>

To add a BAI, SWIFT MT940, or CAMT.053 Format Bank File Transactions application:

1. From the **Data Integration** home page, and then **Actions**, select **Applications**.
2. On the **Applications** page, click .
3. From **Category**, select **Data Source**.
4. From **Type**, select **Bank file**.
5. From **Application**, select an application name from the list of values.

   Available types of application include:

   - BAI Format Bank File Transactions
   - SWIFT MT940 Format Bank File Transactions
   - CAMT.053 Format Bank File Transactions

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> The Data Integration connection to the BAI, SWIFT MT940, camt.053 source file fails under the following circumstances:
>
> - The match type is changed in Transaction Matching.
> - The data source ID changes.
> - The data source attribute ID changes, or is added and removed.
>
> In this case, you need to recreate the application (including the entire target application, import format, location, mapping) in Data Integration.

For a BAI file, select **BAI Format Bank File Transactions**.

For a SWIFT MT940 file, select **SWIFT MT940 Format Bank File Transactions**.

For a CAMT.053 file, select **CAMT.053 Format Bank File Transactions**.

6. In **Prefix**, specify a prefix to make the source system name unique.

   Use a prefix when the source system name you want to add is based on an existing source system name. The prefix is joined to the existing name. For example, if you want to name a Bank file application the same name as the existing one, you might assign your initials as the prefix.

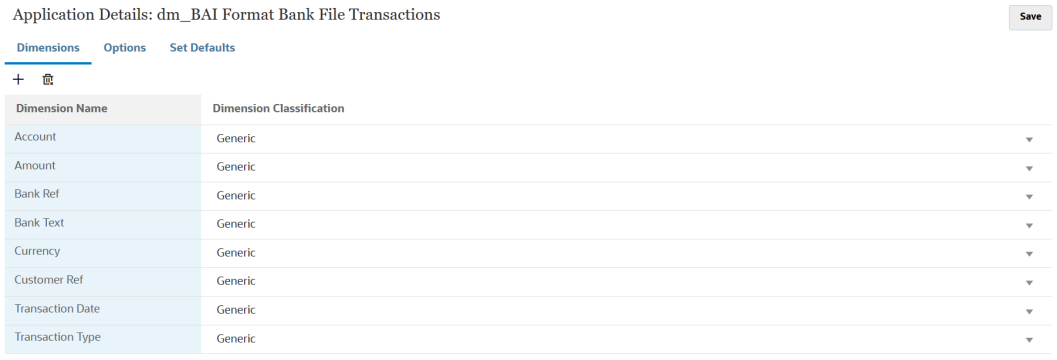
7. Click **OK**.

8. From the **Application** page, click ⋯ to the right of the application, and then select **Application Details**.

9. Select the **Dimensions** tab.

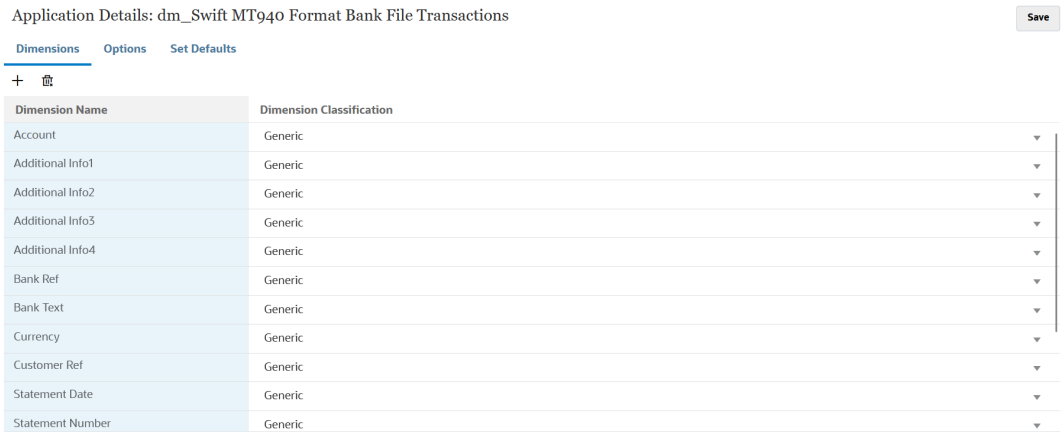
10. To view dimensions in the Bank file source system, select the **Dimension Details** tab.

    The dimension details for a BAI Format Bank File Transactions application are shown below:

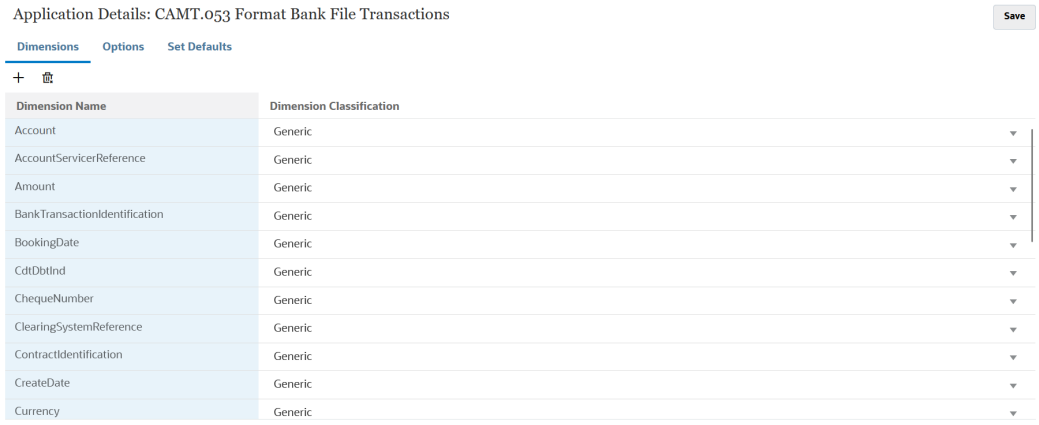


An example of dimension details for a SWIFT MT940 Bank File Transactions application is shown below:

Copyright © 2017, 2026, Oracle and/or its affiliates.



An example of the dimension details for a CAMT.053 Bank File Transactions application is shown below:



11. Set up the integration mapping between Bank file source system and the Account Reconciliation target application.

    a. From the **Data Integration** home page, click to add a new integration.

    b. On the **Create Integration** page, then **Name** and **Description**, enter a name and description for the new integration.

    c. In **Location**, enter a new location name, or pick an existing location where to load data.

    d. From the **Source** ( ) drop-down, select the Account Reconciliation source application.

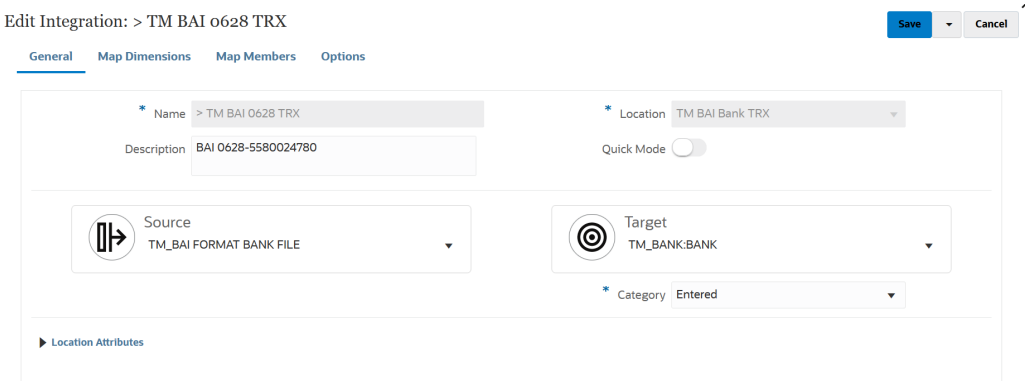
    e. From the **Target** ( ) drop-down, select the target Transaction Matching data source application.

    f. From **Category**, select the appropriate Currency Bucket, typically the lowest enabled Currency Bucket on the Profile.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> Category mappings are not relevant for Transaction Matching transactions, but they are required in Data Integration.

g. **Optional**: Select any applicable location attributes for the integration. For more information, see Selecting Location Attributes.



12. From the **Data Integration** home page, click ⋯ to the right of the integration, and then select **Map Dimensions**.
13. On the **Map Dimensions** page, complete the following:
    a. In **Import Format**, select the name of the import format to use for the integration.

       You can also add a user defined import format name.

    b. In the mappings grid, map the source columns in the source to the dimensions in the target application.

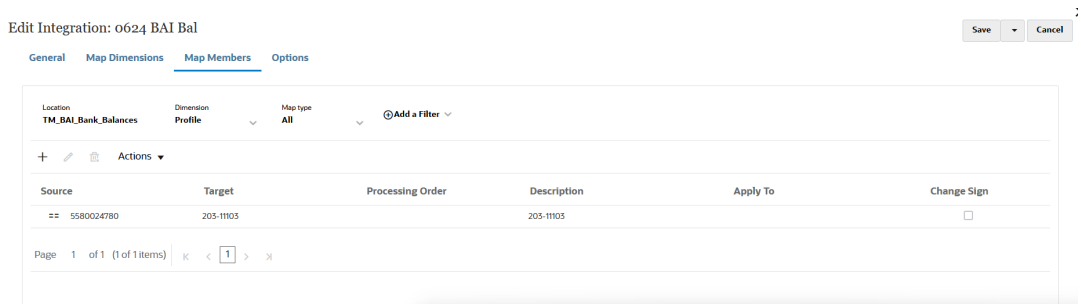
> **Note**
>
> All transaction matching files require the Reconciliation Id dimension to be mapped to the corresponding Transaction Matching Profile.

    The dimensions from the target application are populated automatically.

    If the import format has already been defined for the integration, then the source and target columns are mapped automatically.

14. From the **Data Integration** home page, click ⋯ to the right of the integration, and then select **Map Members**.
15. On the **Map Members** page, define members mapping to map the members from the source to target.

Copyright © 2017, 2026, Oracle and/or its affiliates.



/

> **Note**
>
> BAI codes 100-399 are for bank credits (positive numbers) and 400-699 are for bank debits (negative numbers). For Bank specific BAI codes which are greater than 699, Data Integration treats them as bank credits (positive numbers) by default. If you need any specific code in this range to be treated as bank debit (negative number), then use SQL Mapping (see SQL) to update the Amount as a negative number as in the following example.
>
> ```
> AMOUNTX=
> CASE
>     WHEN UD7 = '868' THEN AMOUNT*-1
>      ELSE AMOUNT
> END
> ```

> **Note**
>
> All transaction matching files require the Reconciliation Id dimension to be mapped to the corresponding Transaction Matching Profile.

16. From the **Data Integration** home page, click ... to the right of the integration, and then select **Workbench**.
17. In the **Workbench**, test and validate the data by executing the integration to ensure that the load is running properly, and your data looks correct. Data Integration transforms the data and stages it for Account Reconciliation to use.

    For more information about using the Workbench, see Running an Integration.

Verifying BAI Bank Statement Balances

Verify correct bank statement using the Bank Statement Verification option.

Account Reconciliation customers can verify that the correct bank statement is being loaded using the Bank Statement Verification feature in Data Integration. When enabled, the system compares the current period expected ending balance to the ending balance provided in the BAI file being loaded.

To calculate the current period expected ending balance, the system adds the net total of the current day transactions to the previous period ending balance for the Account ID. If a

Copyright © 2017, 2026, Oracle and/or its affiliates.

difference is detected or any error is encountered, the transactions remain in the Workbench; however, they are not exported to transaction matching.

Bank Statement Verification enables you to identify:

- duplicate transactions and duplicate data loads
- missing balances or transactions
- statements loaded in the incorrect order (that is, non-chronological order)
- incorrect currency codes

> **Note**
>
> Bank Statement Verification works only when daily periods are configured on the base calendar. Also, when the system retrieves the previous period ending balance for the comparison, it always uses the most recent prior period balance available for the Account ID.

Verifying BAI Bank Statement Balances Considerations

Note the following considerations when using this feature:

- This feature is applicable only for daily periods configured on the base calendar.
- To enable this feature, create a new BAI Format Bank File Transactions application and then create a new integration.
- If the account has just been opened and today is the first day, then disable the Verify Bank Statements setting for the first load and enable it for subsequent loads.
- BAI file loads must be imported and exported at the same time when the Verify Bank Statements is enabled so that a single process ID/Load ID is generated for Import, Validate and Export.
- Import to Workbench and Export to Transaction Matching must be done at the same time.
- When bank statement validation errors occur, the **INFO** log may show the following errors:

1. When the calculated ending balance and the actual ending balance differ, the log shows:

```
2024-04-01 05:50:40,685 INFO  [AIF]:

Profile ID: 38640361-001 - Bank Account Number(s): 38759266, 31461109
Ending Balance Previous Period 26-Jan-2024 (USD) : 47386.00
Total Debits on 29-Jan-2024 (USD)                : (99.00)
Total Credits on 29-Jan-2024 (USD)               : 998.00
Calculated Ending Balance 29-Jan-2024 (USD)      : 48285.00
Actual Ending Balance 29-Jan-2024 (USD)          : 48286.00
Difference (USD)                                 : -1.00

2024-04-01 05:50:40,685 INFO  [AIF]: Bank Statement Balance verification failed. Balance does not match.
2024-04-01 05:50:40,686 ERROR [AIF]: The script has failed to execute
```

2. If a balance has been loaded, but not for the same currency, you get the error message: `"No previous ending balance has been found for Account ID <Account ID and Currency Code>."`

   In this case the log shows:

Copyright © 2017, 2026, Oracle and/or its affiliates.

```
2024-04-01 04:31:14,193 INFO  [AIF]: Previous period balance is not available for Account ID 38640361-001 and Currency Code USD. To verify bank statement balance, a previous balance must be loaded.
2024-04-01 04:31:14,194 ERROR [AIF]: The script has failed to execute
```

3. If no previous ending balance has been loaded for an Account ID, the previous balance is assumed to be zero.

   The log shows the following message:

   `INFO [AIF] No previous ending balance available (INR): 0.00`

4. When the current period balance is not available, the log shows:

```
2024-04-01 05:38:40,899 INFO  [AIF]: Current period balance is not available for Account ID 38640361-001 and Currency Code USD. Please load the current period balance to proceed.
2024-04-01 05:38:40,900 ERROR [AIF]: The script has failed to execute
```

> **Note**
>
> Load the bank balance for the current period prior to loading bank transactions.

5. If you attempt to load a BAI bank statement that has multiple As-of-Dates across one or more "02" records, the system returns the error message: "The file `filename` contains more than one unique As-of-Date. In order to use statement balancing, there must be only one unique As-of-Date in the file. Please adjust the file and reload."

```
2024-04-01 04:31:51,288 INFO  [AIF]: The file " inbox/BSV/BAI_Stmt_Multiple_Days.txt" contains more than one unique As-of-Date.  In order to use statement balancing, there must be only one unique As-of-Date in the file. Please adjust the file and reload.
2024-04-01 04:31:51,288 ERROR [AIF]: The script has failed to execute
```

6. When the As-of-Date does not match with any available period end date, the log shows:

```
2024-04-01 05:41:57,515 INFO  [AIF]: The As-of-date 29-Jan-2024 has previously loaded transactions for Account ID 31461109,38759266,22334512555.  The verify statement balances process supports only one load of transactions per As-of-date per Account ID. This file cannot be loaded with verify statement balances activated.
2024-04-01 05:41:57,516 ERROR [AIF]: The script has failed to execute
```

7. When the default currency of the entered currency bucket was changed but Transaction Matching transactions are shown under a different Entered currency, the log shows:

```
2024-04-01 05:41:57,515 INFO  [AIF]: Appropriate balance not found for Account ID 101-11030 and Currency Code CAD
2024-04-01 05:41:57,516 ERROR [AIF]: The script has failed to execute
```

Enabling the Verify Statement Balance Option

Enabling the Verify Statement Balance Option at the Application-Level

> **Note**
>
> To enable this feature, create a new BAI Format Bank File Transactions application and then create a new integration.

To enable the Verify Statement Balances option at the application-level:

1. From the **Data Integration** home page, and then **Actions**, select **Applications**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

2. From the **Application** page, click to the right of the BAI Format Bank File Transactions application, and then select **Application Detail**.
3. Select the **Options** tab.
4. In the **Verify Statement Balances** field, then **Value**, select **Yes**.
5. Click **Save**.

Enabling the Verify Statement Balance Option at the Integration-Level

To enable the Verify Statement Balances option at the integration-level:

1. From the **Data Integration** home page, and then to the right of the BAI Format Bank File Transactions integration, select **Options**.
2. Select the **Options** tab.
3. In the **Verify Statement Balances** field, then **Value**, select **Yes**.
4. Click **Save**.

Loading Reconciliation Compliance Transactions

Load data from a source system business process to your Reconciliation Compliance Transactions target application.

As an integration mechanism, Data Integration enables Account Reconciliation customers to load Reconciliation Compliance Transactions into a Reconciliation Compliance Transactions target application. You can load the following Reconciliation Compliance transactions:

- Balance Explanations
- Source System Adjustments
- Subsystem Adjustments
- Variance Explanations

Loading Reconciliation Compliance Transactions Process Description

At a high level, this is how you load Reconciliation Compliance Transactions to Account Reconciliation using Data Integration.

> **Note**
>
> As a best practice recommendation while loading transactions through Data Integration , do not replicate your General Ledger or sub-ledgers in Account Reconciliation. Loading activity from your ERP is not a best practice for period end reconciliations. If you need to load more than a 100 transactions; then as an implementer, you need to ask more questions to better understand the customer's requirements. For a reviewer, a large number of transactions for period-end reconciliation would be difficult to review. Use cases with higher volumes of transactions are candidates for Transaction Matching and not Reconciliation Compliance.

To set up the load for Reconciliation Compliance Transactions in Data Integration:

1. Create an Reconciliation Compliance Transactions application.

Copyright © 2017, 2026, Oracle and/or its affiliates.

See Registering a Reconciliation Compliance Transactions Application.

2. Create an integration between the source and Reconciliation Compliance Transactions file and the target Reconciliation Compliance Transactions application.

   When creating the integrating, you also specify location information.

   See Creating a Reconciliation Compliance Transactions Integration.

3. Map the dimensions between the source file to the dimensions in the Reconciliation Compliance Transactions application by building an import format.

   See Mapping Dimensions.

   The "Data" dimension is required when mapping dimensions. You must specify "All Data Type" instead of "Numeric Data," for example "Delimited – All Data Type."

4. Map the "Data" and "Profile" dimension so that the system can create a mapping to pair intersections of data from the source to the target during the load.

   For the others dimensions, member mappings are not required when most attributes are mapped to the ATTR target dimension class. However, they are required when they are mapped to UD.

   See Mapping a Data Member for Reconciliation Compliance Transactions.

5. Run the integration.

   See Running the Reconciliation Compliance Transactions Integration.

Registering a Reconciliation Compliance Transactions Application

You register a Reconciliation Compliance Transaction application by selecting the "Data Export" category and "Reconciliation Compliance Transactions" as the application type, and then you select the application name.

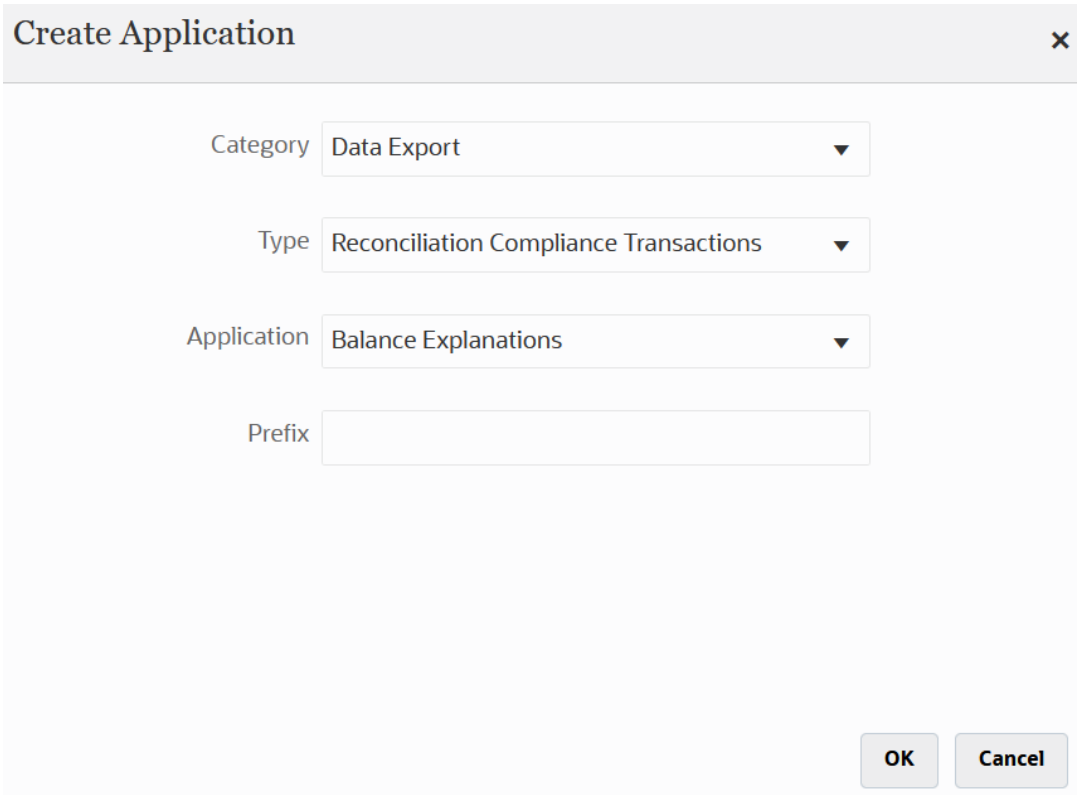
To add a Reconciliation Compliance Transactions target application:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, click (Add icon).
3. On the **Create Application** page, and then **Category**, select **Data Export**.
4. From **Type**, select **Reconciliation Compliance Transactions**.
5. From **Application**, select the Reconciliation Compliance Transactions application transaction type.

   Application Name is a drop-down list with the following transaction types:

   - Balance Explanations
   - Source System Adjustments
   - Subsystem Adjustments
   - Variance Explanations

Copyright © 2017, 2026, Oracle and/or its affiliates.



6. In **Prefix**, optionally specify a prefix to make the application name unique.

   The prefix is concatenated with the application name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.

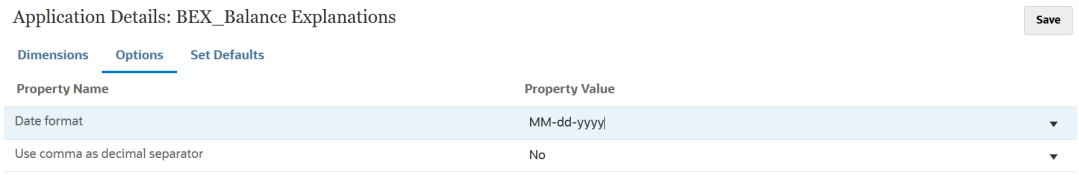
7. Click **OK**.

8. From the **Applications** page, select the Reconciliation Compliance Transactions

   Application that you just registered above, then click ... to the right of the application, and then select **Application Details**.

9. Select the **Options** tab.

   Your selections on the Options tab apply to the target application.



10. From **Data format**

    Available options:

    - MM/dd/yyyy
    - MM-dd-yyyy
    - MMM d, yyyy
    - d-M-yyyy

Copyright © 2017, 2026, Oracle and/or its affiliates.

- dd-MMM-yy
- dd/MM/yyyy

By default, the date format is blank, which enables the system to use all date formats to parse date values.

11. From **Use commas as decimal separator**, select **Yes** to use a comma as the decimal separator.

    Select **No** to use the user-locale for number parsing.

    By default this option is set to **No** (the decimal separator is based on the user-locale and not a comma.)

12. Click **Save**.

Creating a Reconciliation Compliance Transactions Integration

You can load Reconciliation Compliance Transactions from a file into a Reconciliation Compliance Transactions application by creating an integration.

1. From the **Data Integration** home page, click , and then select **Integration**.

   The General page is displayed in Create Integration view.

2. In **Name** and **Description**, enter a name and description for the Reconciliation Compliance Transactions integration.
3. In **Location**, enter a new location name, or pick an existing location to specify where to load data.
4. Click (Select a Source).
5. From the **Select a Source** page, click **File**.
6. From the **File Browser**, select the transactions file from which to load to the Reconciliation Compliance Transactions application and click **OK**.

   You can double-click the **inbox** or **outbox** folders, or any other folders to see a list of files in folders.

   You can also click **Upload** and navigate to a file to upload it.

   Optionally, you can create a file-based load in which you designate the file to be loaded at runtime. However, the column number and column name are designated only when you map the dimensions. See Mapping Dimensions.

   You can select the following types of files.

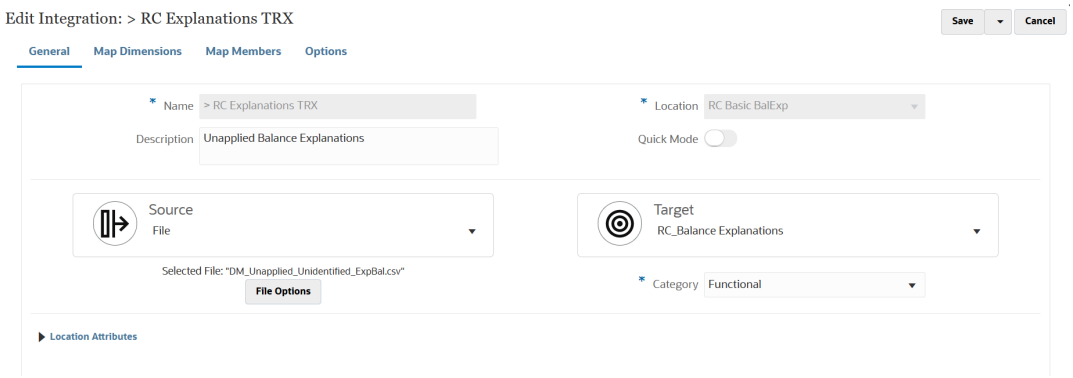
   - comma separated file (CSV)
   - tab separated file (TXT)
   - custom delimiter file

7. Click (Select a Target).
8. From the **Select a Target** page, select the Reconciliation Compliance Transactions target application.
9. From **Category**, select a currency bucket.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> Category Mappings are not applicable while loading transaction data into Reconciliation Compliance transaction Targets. However, as a best practice, choose a currency bucket that aligns with the lowest enabled currency bucket on your profiles.

10. Click **Save**.



Mapping Reconciliation Compliance Transactions Attributes to Dimensions

When you map a Reconciliation Compliance Transactions target application, the dimension details of the application are populated automatically on the Map Dimensions page

Dimension names must match exactly with attribute names in Account Reconciliation. If the dimension is for a standard and custom attributes, its name should be exactly as specified here and should not be changed.

By default "Profile" is mapped to the "Account" (Reconciliation Account ID) target dimension class and "Period" is mapped to the "Period" target dimension class.

The following dimensions are assigned to the Attribute target dimension class and are mapped to the ATTR1 to ATTR4 columns respectively. If mappings rules are needed for these dimensions, change them to Lookup dimension types and map them to UD (user-defined) columns. Attribute dimensions can have no mapping rules.

Table 17-5 Default List of Dimensions

Dimension Name	Target Dimension Class	DB Table Column
Short Description	Attribute	ATTR1
Long Description	Attribute	ATTR2
Transaction Date	Attribute	ATTR3
Close Date	Attribute	ATTR4

For more information about Lookup dimensions, see Adding Lookup Dimensions.

The following are standard dimensions and the names should not be changed. Dimensions for unused currency buckets can be deleted.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 17-6 Standard Dimension Details

Dimension Name	Target Dimension Class	DB Table Column
Amount (Entered)	Attribute	ATTR5
Currency (Entered)	Attribute	ATTR6
Amount (Functional)	Attribute	ATTR7
Currency (Functional)	Attribute	ATTR8
Amount (Reporting)	Attribute	ATTR9
Currency (Reporting)	Attribute	ATTR10

Other standard dimensions are shown below. These can either be Lookup or attribute dimensions. Because Reconciliation Compliance Transactions allow the same custom attributes to be assigned to the transaction itself and its action plan, the system differentiates between custom attributes for the transaction and custom attributes for the action plan. In this case, the system prefixes **Action Plan** at the beginning of dimension names for action plan attributes.

Table 17-7 Other Standard Dimensions

Dimension Name	Target Dimension Class	DB Table Column
Child Account ID—The child account ID for child transactions of a summary reconciliation.	LOOKUP or Attribute	Select any UD columns or ATTR columns.
<sub-segment name>—Sub-segments for transactions of a group reconciliation. The dimension name should match the sub-segment name in Account Reconciliation, for example, Store, Entity.	LOOKUP or Attribute	Select any UD columns or ATTR columns.
Action Plan <custom attribute name>	LOOKUP or Attribute	Select any UD columns or ATTR columns
Amortization	LOOKUP or Attribute	Select any UD columns or ATTR columns.
Amortization Method	LOOKUP or Attribute	Select any UD columns or ATTR columns.
Amortization Half Month Convention	LOOKUP or Attribute	Select any UD columns or ATTR columns.
Amortization Periods	LOOKUP or Attribute	Select any UD columns or ATTR columns.
Amortization Start Period	LOOKUP or Attribute	Select any UD columns or ATTR columns.
Amortization End Date	LOOKUP or Attribute	Select any UD columns or ATTR columns.
Amortization Original Amount<index>	LOOKUP or Attribute	Select any UD columns or ATTR columns.
Action Plan	LOOKUP or Attribute	Select any UD columns or ATTR columns.
Action Plan Closed	LOOKUP or Attribute	Select any UD columns or ATTR columns.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 17-7 (Cont.) Other Standard Dimensions

Dimension Name	Target Dimension Class	DB Table Column
Action Plan Close Date	LOOKUP or Attribute	Select any UD columns or ATTR columns.

To define dimension details for a Reconciliation Compliance Transactions application:

1. From the **Data Integration** home page, click to the right of the Reconciliation Compliance Transactions target integration, and then select **Map Dimensions**.
2. From **Import Format**, select the name of the import format to use for the integration.

   You can also add a user defined import format name.
3. From **Type**, select **Delimited-All Data Type**.

   The Delimited–All Data Type load all data types from a delimited file format.
4. From **Delimiter**, select the character to use for delimiting columns in the output file.

   A delimited file contains one or more records set off from each other by a specified delimiter.

   Available options:

   - Comma (,)
   - Pipe (|)
   - Exclamation (!)
   - Semicolon (;)
   - Colon (:)
   - Tab
   - Tilde (~)
5. In the mappings grid, map the source columns in the source data-load file to the dimensions in the target application.

   The dimensions from the Reconciliation Compliance Transactions target application are populated automatically.

   If the import format has already been defined for the file, then the source and target columns are mapped automatically.
6. Click **Save**.

Mapping a Data Member for Reconciliation Compliance Transactions

As a requirement in Data Integration, you map the "Data" and "Profile" dimension so that the system can create a mapping to pair intersections of data from the source to the target during the load.

For the others dimensions, member mappings are not required when most attributes are mapped to the ATTR target dimension class. However, they are required when they are mapped to UD.

Map the "Data" and "Profile" dimensions so that the system can create a mapping to pair intersections of data from the source to the target during the load.

Copyright © 2017, 2026, Oracle and/or its affiliates.

For the others dimensions, member mappings are not required when most attributes are mapped to the ATTR target dimension class. However, they are required when they are mapped to UD.

A member mapping is required for the "Data" dimension so that the system can create target values.

To create a member mapping for the Data dimension:

1. From the **Data Integration** home page, click ⋯ to the right of the Reconciliation Compliance Transactions integration, and then select **Map Members**.
2. On the **Map Members** page, and then from the **Dimension** drop-down, select **Data**.
3. From the **Map Type** drop-down, select the **Explicit**.
4. Click ➕.
5. From the **Add Member Mapping** page, in **Source Value**, enter ***** (asterisk).

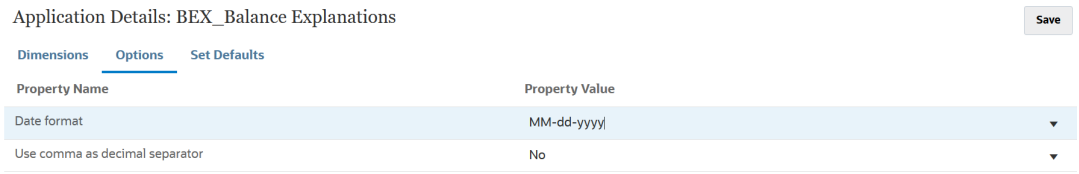
   An asterisk (*) represents the source value.
6. In **Target Value**, enter a reference to the type of transaction type.

   For example, you might enter **BEX** to identify the mapping as "Balance Explanation."
7. Click **Save**.

Defining Application Details for a Reconciliation Compliance Transactions Application

1. From the **Applications** page, select a registered Reconciliation Compliance Transactions application, then click ⋯ to the right of the application, and then select **Application Details**.
2. Select the **Options** tab.

   Your selections on the Options tab apply to the target application.



3. From **Data format**.

   Available options:

   - MM/dd/yyyy
   - MM-dd-yyyy
   - MMM d, yyyy
   - d-M-yyyy
   - dd-MMM-yy
   - dd/MM/yyyy

   By default, the date format is blank, which enables the system to use means all date format to parse date values.

Copyright © 2017, 2026, Oracle and/or its affiliates.

4. From **Use commas as decimal separator**, select **Yes** to use a comma as the decimal separator.

   Select **No** to use the user locale for number parsing.

   By default this option is set to **No** (the decimal separator is based on the user locale and not a comma.)

5. Click **Save**.

Defining a Period for Reconciliation Compliance Transactions

Account Reconciliation transactions must be loaded to one period. To do this, use global mappings to map various periods to one period.

To define a global mapping period for Reconciliation Compliance Transactions:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Period Mapping**.

   Specify the date in *MM/dd/yyyy* format.

2. Select the **Global Mapping** tab.

3. Click

   .

4. In **Period Key**, specify the last day of the month of the current fiscal year to be mapped from the source. This is a date value that is stored in the database during the trial-balance load process and that becomes part of a key that identifies a set of trial-balance records.

   Specify the date in *MM/dd/yyyy* format.

   The Reconciliation Compliance Transactions load uses the Period Key and Prior Period Key defined in Data Integration to determine the source General Ledger periods mapped to each Data Integration period when the integration is run.

5. Select the **Prior Period Key**, specify the prior fiscal period key that is used during export.

   The Prior Period Key is just one month earlier to your current period key.

   The Prior Period Key is just one month earlier to your current period key.

6. Complete the following:

   a. Period Name; for example, July 2018

   b. Target Period Month; for example, August

   c. Target Period Year

   d. Target Period Day

   e. Year Target

7. Click **Save**.

Running the Reconciliation Compliance Transactions Integration

When you execute an integration in Data Integration that has a Reconciliation Compliance Transactions target application, Data Integration imports data from the source and stages it. Then Data Integration exports the data to Account Reconciliation by generating a .CSV file with the same format as the Account Reconciliation pre-mapped transactions import file and

Copyright © 2017, 2026, Oracle and/or its affiliates.

submits a job to Account Reconciliation for import. Transactions with the same Account ID previously loaded from Data Integration are overwritten.

To load data to a Reconciliation Compliance Transactions target application:

1. From the **Data Integration** home page, select an Reconciliation Compliance Transactions integration, and then click ▶.
2. From **Import Mode**, select the method for importing data.

   Available import modes:

   - **Append**—Keep existing rows for the POV but append new rows to the POV. For example, a first-time load has 100 rows and second load has 50 rows. In this case, 50 rows are appended. After this load, the row total for the POV is 150.
   - **Replace**—Clears all data for the POV in the target, and then loads from the source or file. For example, a first-time load has 100 rows, and a second load has 70 rows. In this case, 100 rows are removed, and 70 rows are loaded to TDATASSEG. After this load, the row total is 70.
   - **Merge**—Not applicable.
   - **No Import**—Skip the import of data entirely.
   - **Map and Validate**—Skip importing the data but reprocess the data with updated mappings.
3. From **Export Mode**, select the method for exporting data to the target application.

   Available export mode options:

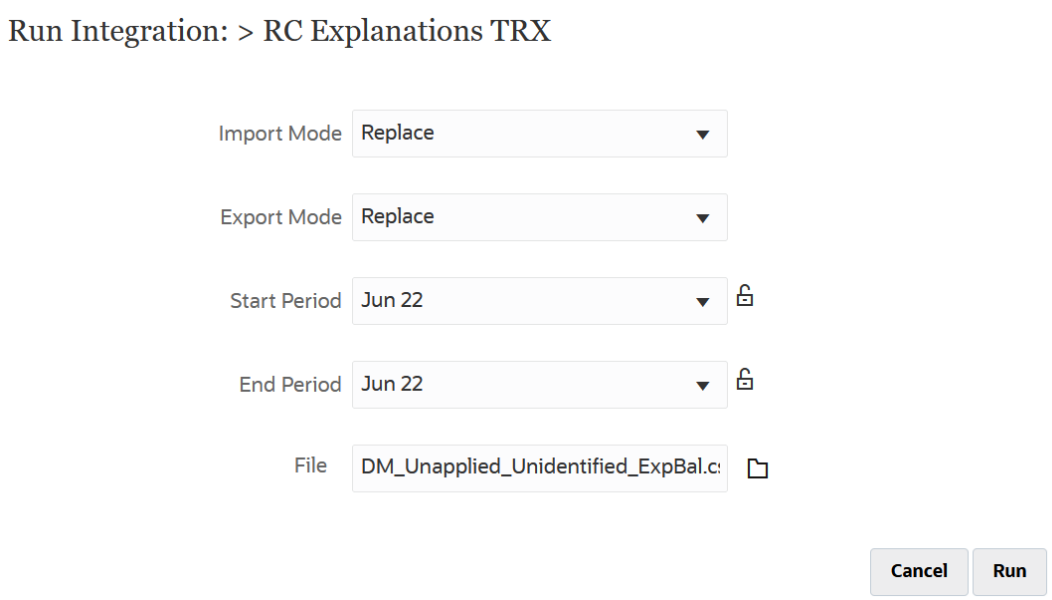
   - **Replace**—Transactions with Account IDs previously imported using Data Integration

     are replaced with the transactions staged in the Workbench. Transactions added through the Reconciliation dialog, Carried Forward from a previous period, imported using Pre-Mapped Transactions, or imported using Smart View are not be affected.
   - **No Export**—Skip the export of data entirely.
4. If the source system is file based, in **File Name**, select the data file name that contains the data you are loading.

   When the file name is provided, then data must be entered for a single period on the Rules Execution window.

   To navigate to a file, click 🗀.
5. From **Start Period** and **End Period**, select the period defined for Reconciliation Compliance Transactions.
6. Click **Run**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



7. In Account Reconciliation, prepare the reconciliation.

Loading Exported Journal Entries

Export adjustments or transactions as dual-sided journal entries that can then be imported into your ERP system or data export file.

In Transaction Matching, you can export adjustments or transactions as dual-sided journal entries that can then be imported into your ERP system or data export file. Using Data Integration, you can create a data source associated with an Accounts Reconciliation journal adjustments source system, and then load the entries to a data export application. When the entries are exported in the integration, all adjustments and transaction that have an open status and match any applied filter criteria are exported.

> **Note**
>
> In Account Reconciliation, updates made to journal columns using Journal Columns under Match Types automatically update the data source Account Reconciliation Journal Adjustments in Data Integration. These updates can be made by Service Administrators and users with the **Match Types - Manage granular** role. Verify the integrations when making updates to journal columns in Transaction Matching.

To load exported journals entries:

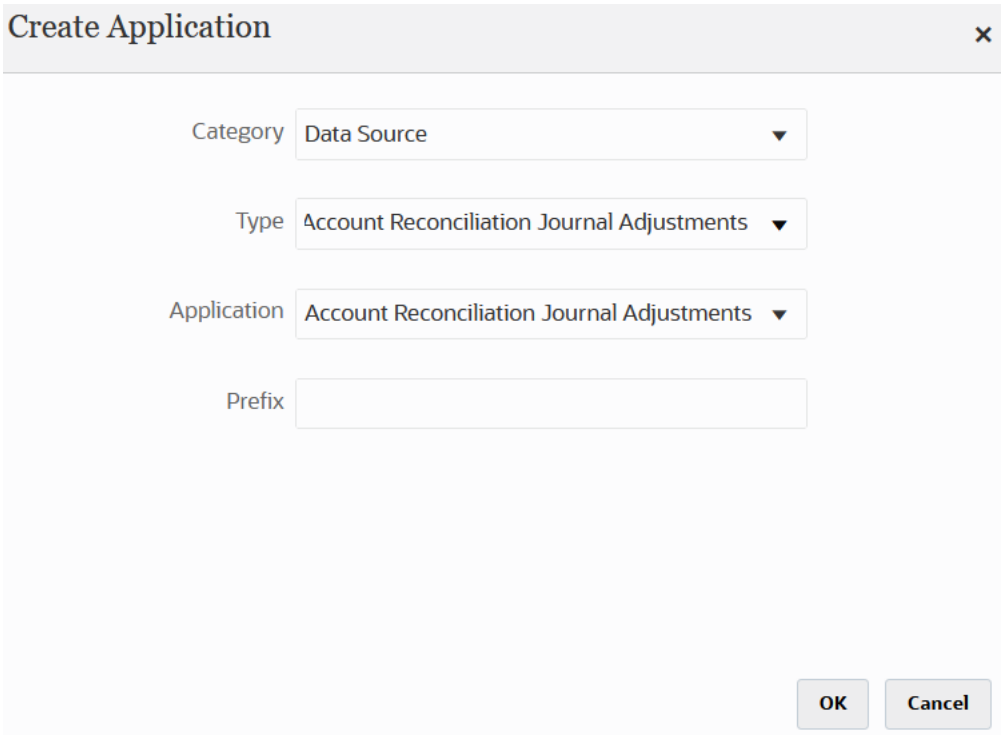
1. Create the source **Account Reconciliation Journal Adjustments** application by completing the following:
   a. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
   b. On the **Applications** page, click (Add icon).
   c. On the **Create Application** page, and from then **Category**, select **Data Source**.
   d. From **Type**, select **Account Reconciliation Journal Adjustments**.
   e. From **Application**, select the Account Reconciliation Journal Adjustments application.

Copyright © 2017, 2026, Oracle and/or its affiliates.

f. In **Prefix**, optionally specify a prefix to make the application name unique.

   The prefix supports up to a maximum of ten characters.

g. Click **OK**.



2. Create the target application.

> **Note**
>
> If you want to write back to the Oracle ERP Cloud instead of a file, see: Writing Back Actuals to the Oracle General Ledger.

a. From the **Data Integration** home page, and then **Actions**, select **Applications**.

b. On the **Applications** page, click (Add icon).

c. On the **Create Application** page, and then **Category**, select **Data Export**.

d. From **Type**, select the **Data Export to File**.

e. From **Application**, select the target application to which to export adjustments.

f. In **Prefix**, optionally specify a prefix to make the application name unique.

   The prefix supports up to a maximum of ten characters.

g. Click **OK**.

3. From the **Application** page, click ... to the right of the application, and then select **Application Details**.
4. Select the **Dimensions** tab.
5. In the **Application Details** section, in the **Name** field, specify the name of the custom application.

Copyright © 2017, 2026, Oracle and/or its affiliates.

6. Select the **Dimension Details** tab.
7. Click **Add**.
8. In **Dimension Name**, enter the name of the Account dimension, and then specify **Account** in the **Target Dimension Class**. and **Data Table Column Name** fields.

   When you specify **Account** in the **Target Dimension Class**. the value **Account** prefills automatically in the **Data Table Column Name** field.

   This step enables you to create dimensions not defined in the target application. You must create an "Account" dimension name and associate it with an "Account" target dimension class, which is a property defined by the dimension type. You also associate it with an "Account" data table column name, which identifies the table column name of the column in the staging table. These definitions are required by Data Integration.

   You do not need to create an "Amount" dimension because it defaults on the Map Dimensions page.

   You can create other dimension names, such as UD1 through UD20 or "Entity" as needed.
9. Create the integration between the source Account Reconciliation Journal Adjustments source application and the data export to file or ERP application.

   Complete the following

   a. From the **Data Integration** home page, click , and then select **Integration**.

      The General page is displayed in Create Integration view.

   b. In **Name** and **Description**, enter a name and description for the Account Reconciliation Journal Adjustments Transactions Integration.

   c. In **Location**, enter a new location name, or pick an existing location to specify where to load data.

   d. Click (Select a Source).

   e. From the **Select a Source** page, click the **Account Reconciliation Journal Adjustments** source application.

   f. Click (Select a Target).

   g. From the **Select a Target** page, select the data export file application or Oracle ERP Cloud application.

   h. Click **Save**.
10. Select **Map Dimensions**.

    When you add an Account Reconciliation Journal Adjustment data source, the dimensions in application are populated automatically on the Dimension Detail tab.
11. From **Import Format**, select the name of the import format to use for the integration.

    You can also add a user defined import format name.
12. Select the **Dimension Detail** tab.
13. Map all dimension names in the **Dimension Names** column with the value **Generic** in the **Target Dimension Class** column and click **Save**.
14. From the **Data Integration** home page, click to the right of the integration, and then select **Map Members**

Copyright © 2017, 2026, Oracle and/or its affiliates.

15. On the **Map Members** page, map members between the Accounts Reconciliation Journal Adjustment source columns and the custom target application columns.

    Do not add mappings for the "Data" dimension.

16. From the **Data Integration** home page, click to the right of the Account Reconciliation Journal Adjustment /Transaction Mapping integration, and then select **Options**.

17. Click the **Filter** tab.

18. Click .

19. On the **Filters** tab, complete any parameters based on the transaction matching type.

    Available parameters:

    - Type—Specify the type of reconciliation.

      Available types:

      – **Transactions**

      – **Adjustments**

    - Match Type—Specify the match type ID such as "Clearing."

      Match Types determine how the transaction matching process works for the accounts using that match type. They determine the structure of the data to be matched, as well as the rules used for matching. Additionally, match types are used to export adjustments back to an ERP system as journal entries in a text file.

    - Data Source—Specify the data source when the transaction matching transaction type is "Transactions."

      Leave this field blank when the transaction matching transaction type is "Adjustments."

      Names for the data sources that appear in Data Integration are actually sourced from the Transaction Matching data sources. The convention used in the drop-down is *Match Type Name: Data Source Name.*

      For example, application choices might include:

      – InterCo3:AR

      – InterCo3:AP1 3

      – Bank BAI:BAI_Bank_File

      – Bank BAI:GL

      – INTERCO2:AR

      – INTERCO2:AP

      – INTERCO:AR 8

      – INTERCO:AP 9

      – CLEARING:CLEARING

    - Filter—If you choose **Type** as the Transaction, specify the filter name for transactions.

      The filters is defined in data source configuration in Account Reconciliation as shown below:

    If you choose Type as **Adjustment**, specify the filter value in JSON format.

    You can select specific transaction types and/or the accounting date while exporting the journal for Adjustments.

Copyright © 2017, 2026, Oracle and/or its affiliates.

To specify the filter for Adjustments, use the **Filter** field to select the following:

- (Adjustment) Type—Specify the adjustment type available for the match type selected in the previous step. You can specify one or more values. If you don't select a value, the default used is **All**.
- Adjustment Date—Specify the operand and date values (using the Date Picker to select the dates). The operands available for filtering are: EQUALS, BEFORE, BETWEEN, and AFTER.

  The date format must be YYYY-MM-DD. If you use EQUALS, BEFORE, and AFTER operands, use the JSON format: `accountingDate` and then specify the accounting date. If you select a BETWEEN operand, use the JSON format:

  – `fromAccountingDate` for the "from" Accounting Date
  – `toAccountingDate` for the "to" Accounting Date

  Here are some sample JSON formats:

```
{"adjustmentTypes" : ["Invoice Dispute","Coding Error"],"operator" :
"BETWEEN", "fromAccountingDate" : "2022-02-01", "toAccountingDate" :
"2022-02-10"}


{"adjustmentTypes" : ["Invoice Dispute","Coding Error"],"operator" :
"EQUALS", "accountingDate" : "2022-02-01"}


{"operator" : "AFTER", "accountingDate" : "2022-02-01"}


{"adjustmentTypes" : ["Invoice Dispute","Coding Error"]}
```

20. From the **Data Integration** home page, select the Account Reconciliation Journal

    Adjustment /Transaction Mapping integration, and then click to the right of the Account Reconciliation Journal Adjustment data source integration, and then select **Workbench**.

21. In the **Workbench**, test and validate the data by executing the integration to ensure that the data is loaded properly, and your data looks correct. Data Integration transforms the data and stages it to post to the Oracle ERP Cloud.

    For information on running the integration using runIntegration in *Working with EPM Automate for Oracle Enterprise Performance Management Cloud* EPMAUTOMATE.

    Additionally, to post journals to Oracle ERP Cloud, see Writing Back Actuals to the Oracle ERP Cloud - Oracle General Ledger in *Administering Data Management for Oracle Enterprise Performance Management Cloud* .

Integrating Metadata

Load metadata from a flat file to an Oracle Fusion Cloud Enterprise Performance Management environment.

Data Integration supports the loading of metadata from a flat file in the order provided in the file. This feature allows customers to build a metadata load file in any format, from any source, and load the metadata to an Cloud EPM environment. Using this approach, users can set property defaults during the load or mapping process.

Copyright © 2017, 2026, Oracle and/or its affiliates.

For example, Workforce customers can load employees, jobs, organizations and other work structure and compensation related items from Oracle Human Capital Management Cloud to the Planning.

Regular (such as Account and Entity), custom (such as Product), and Smart List dimensions are only supported, and for the following services only:

- Planning Modules
- Planning
- Financial Consolidation and Close
- Tax Reporting

> **Note**
>
> Loading metadata is only available for applications that are application type: Planning. If the application type is Oracle Essbase, then use the Planning Outline Load Utility to load metadata.

> **Note**
>
> Profitability and Cost Management does not support the loading of metadata by way of a file using Data Integration.

Metadata Load File Considerations

Consider these points when working with metadata load files:

- For each dimension in the application, you create a load file with fields corresponding to the dimension properties. Each load file can contain members for only one dimension. You can define multiple members for each dimension.
- Required fields are different for each dimension being loaded. For detailed information on the properties available for each Planning member, see the *Dimension Properties* topic in the *Administering Planning* guide.
- The member name and parent name must be included as fields in the load file. If a member property value is not specified, the new member will inherit the property values from its parent.
- When adding new members, unspecified values are inherited from the parent member's property value as appropriate. If the member exists and no value is specified, it is left as is.
- When you load a member that already exists in the Planning application (for example, to change a property) and a parent is not specified in the load file, the member is left under the existing parent. If a new parent is specified, the member is moved under the new parent.
- Only one dimension can be loaded per load file.
- Records are loaded one by one. If a record fails to load, its associated exception is written to the exception file and the load process resumes with the next record.
- Metadata is loaded in the order provided in the file.
- Member names with parenthesis are treated as functions.

Copyright © 2017, 2026, Oracle and/or its affiliates.

- When you load metadata using an integration to a Planning application, the export mode parameter must be set to "Merge".

Loading Metadata Process Description

At a high level, this is how you load metadata from a file to a Planning application:

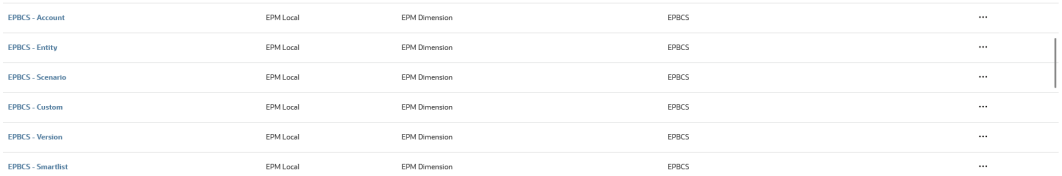
1. Generate a metadata load file for each dimension to be loaded.

   A metadata load is only available to applications built on the Planning platform only.

   For information on how to construct a metadata load file, see Metadata Load File Considerations.

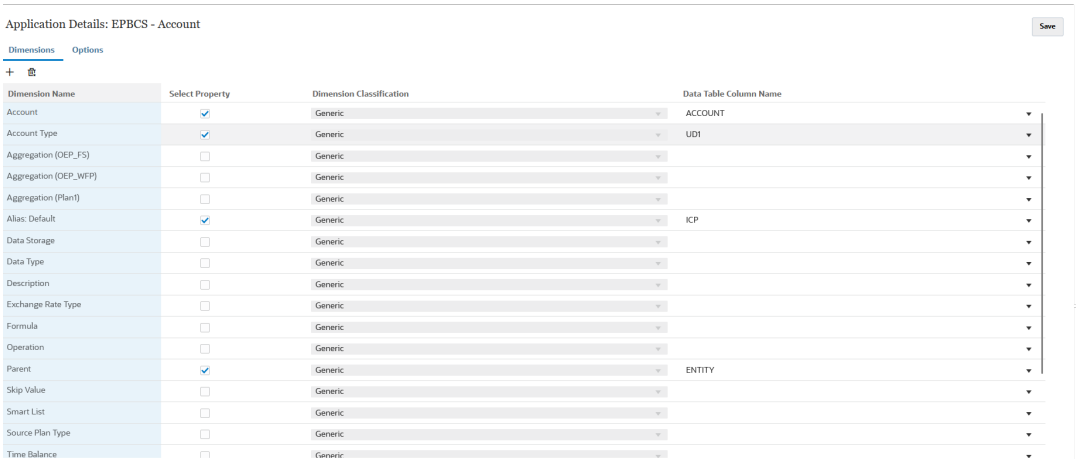
2. In **Application**, register a target application for the class of dimension or dimension type.

   Data Integration creates six-dimensions applications automatically: Account, Entity, Custom, Scenario, Version, and Smartlist.



   For more information on adding a dimension class or type as a target application, see Registering a Target Application for the Class of Dimension or Dimension Type.

3. In **Application Details**, the system creates a record for each dimension property. Only the member name and parent are required fields.

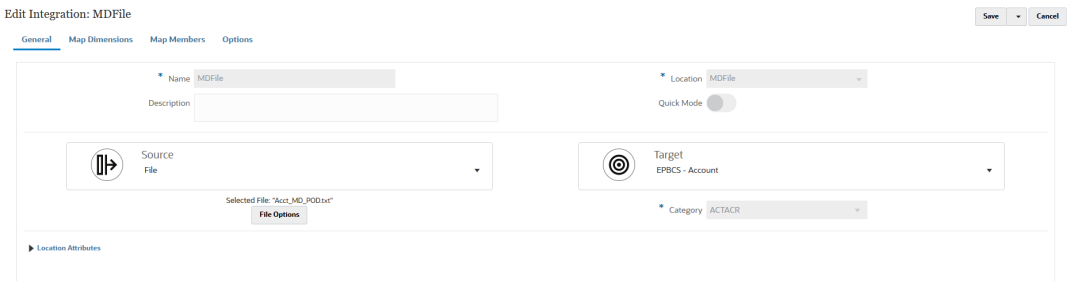


   To enable additional properties, add a row to the dimension metadata application. The name of the row is the property or attribute name used in the Planning application.

4. **Optional**: To add a custom dimension (one designated as Generic in the Planning application), in the target application, select the property name and enable the **Select Property** field, and then map it to a **Data Table Column Name** value. Next create a separate import format for each generic dimension. Then, in the dimension's data rule, specify the dimension name (for example, Product, Movement) in the Dimension name of the data rule's target options.

5. In **Create Integration**, create an integration between the metadata file and the target application.

Copyright © 2017, 2026, Oracle and/or its affiliates.

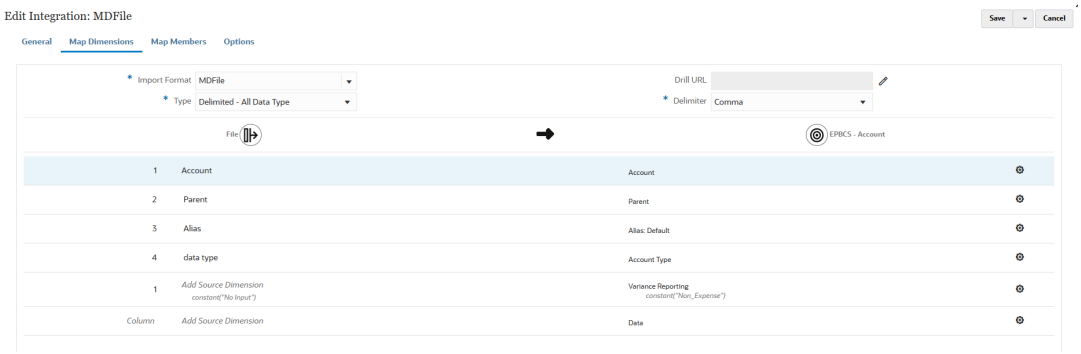


For more information, see Creating File-Based Integrations.

6. In **Map Dimensions**, map the data from the metadata load file to the properties of the dimensions in the Oracle Fusion Cloud Enterprise Performance Management application.

   This allows users to import dimension members from any file format. (The file must be "delimited - all data type" file type.)

   Properties are added as "dimensions" of a dimension application. For example, the Two Pass Calculation property of Entity is added as a dimension and the flat file adds the "yes" or "no" property on the load.



> **Note**
>
> Checked dimensions in the dimension "application" are the ones loaded. If you do not map them, the load will fail. There is no default if a mapping is missing. To avoid loading a "field" such as alias, uncheck the check box in the target application. To supply a single value for all loaded rows, specify the value in the Expression field and map *to* for that dimension.

For more information, see Creating the Dimension Maps.

7. On the **Run Integration** page, from **Export Mode**, select **Merge** and then select all other required parameters.

   You can execute the integration or one or more periods. You then verify that the data was imported and transformed correctly, and then export the data to the target application.

   Running an Integration.

8. You can also to import and export metadata using the Pipeline. For more information, see Using an Import Metadata Job Type and Using an Export Metadata Job Type.

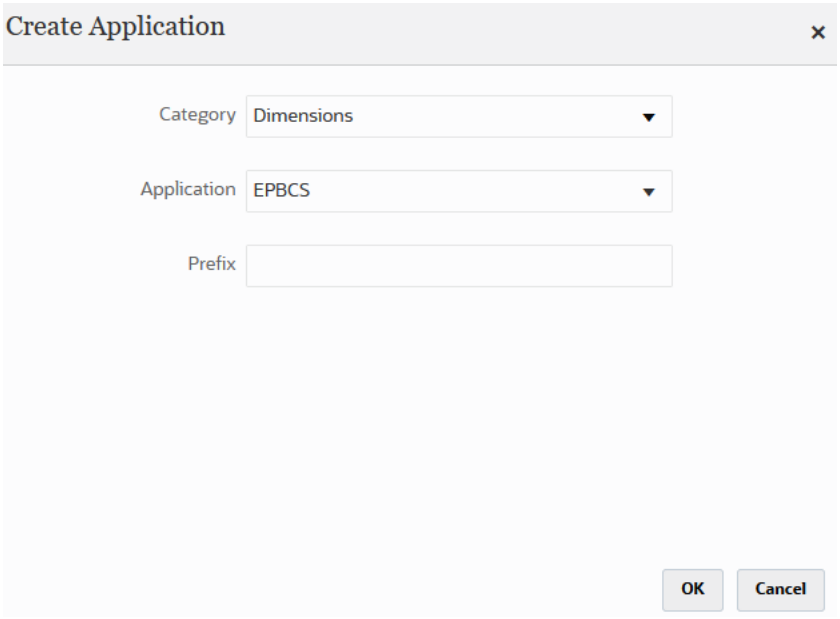
Copyright © 2017, 2026, Oracle and/or its affiliates.

Registering a Target Application for the Class of Dimension or Dimension Type

In the Application option, you add a new target application for the class of dimension or dimension type. This application is used as the target application for loading metadata. When you add a dimension, Data Integration creates six-dimensions applications automatically: Account, Entity, Custom, Scenario, Version, and Smartlist.

To register an application for a class of dimensions or dimension type:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, click (Add icon).
3. On the **Create Application** page, and then **Category**, select **Dimensions**.
4. From **Application**, select the target application.
5. To specify a prefix name to use with each target application, in **Prefix**, specify a unique name.



6. Click **OK**.

   Data Integration creates the dimension application name using this format: "Application Name -" + Dimension Type.

   You can add dimensions multiple times. Data Integration checks if a dimension application exists. If a dimension exists, it is skipped, and if not, it is created. This enables you to delete applications as needed and recreate it.

Loading Data Using an Incremental File Adapter

The Incremental File Adapter feature enables you to compare a source data file with a prior version of the source data file and identify new or changed records and then load only that

Copyright © 2017, 2026, Oracle and/or its affiliates.

data set. You can sort the initial source data file before making the comparison or provide a pre-sorted file for better performance.

To use this feature, you register an initial source data file as an incremental file adapter. The initial source data file is used as the template. Actual data loads are run from the file designated in the integration where a file comparison is run between the initial source data file and a subsequent file. You can load once, twice or many times thereafter. The last run file becomes the basis against which the subsequent load is evaluated. The adapter loads only the differences, which leads to a faster load during the file import. The remaining data import processes stay the same as in a standard data load for a file.

Considerations:

- The source data file must be a delimited data file.
- Data files used must contain a one-line header, which describes the delimited columns.
- Both numeric and non-numeric data can be integrated.
- Any deleted records between the two files are ignored. In this case, you must handle the deleted records manually.
- If the file is missing (or you change the last ID to a non-existent run), the load completes with an error.
- Sort options determine the level of performance using this feature. Sorting increases the processing time. Pre-sorting the file makes the process faster.
- Only single period data loads are supported for an incremental load. Multi-period loads are not supported.
- Drill down is not supported for incremental loads since incremental files are loaded in Replace mode and only the last version of the file comparison is present in the staging table.

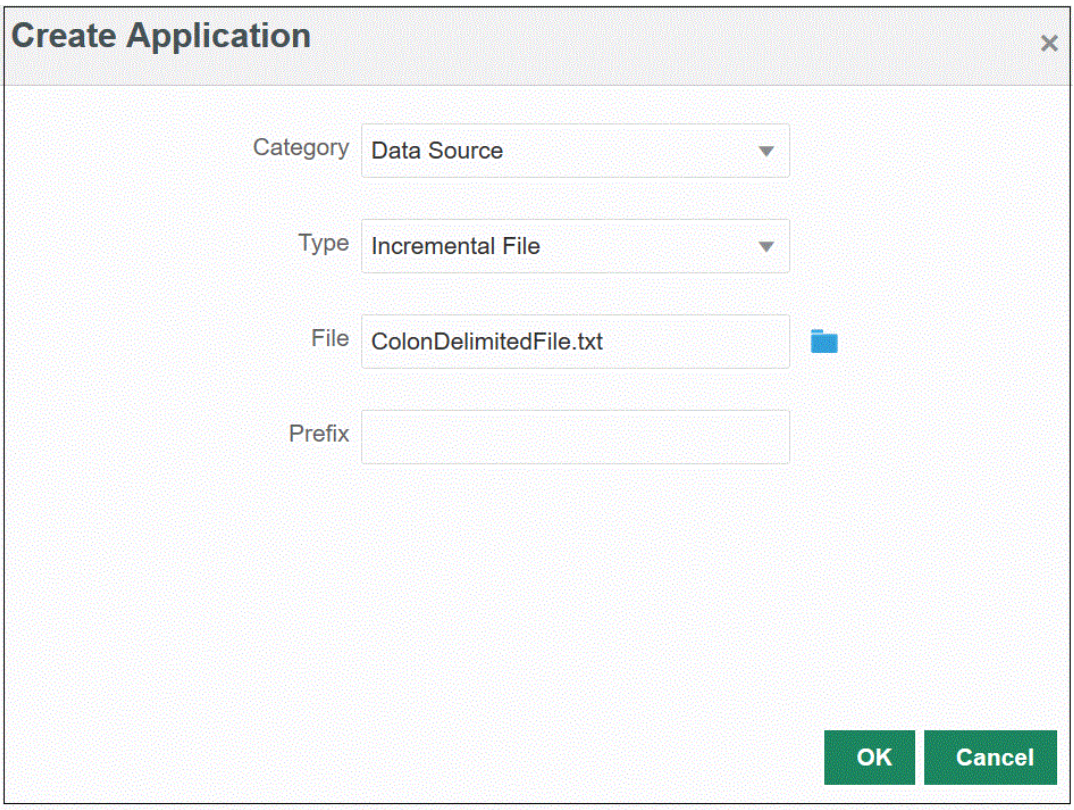
  As a workaround, you can load the same data file to another location using the full data load method. In this case, you should import data only and not export it to the target application.
- Copies of the source data file are archived for future comparison. Only the last 5 versions are retained. Files are retained for a maximum of 60 days. If no incremental load is performed for more than 60 days, then set the Last Process ID to 0 and perform the load.

Setting up the Incremental File Adapter

To register an incremental file data source:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, click + (Add icon).
3. On the **Create Application** page, and then **Category**, select **Data Source**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



4. From **Type**, select **Incremental File**.
5. From **File**, specify the initial source data file to use as the template.

   Click to search for a file from the **File Browser** page.

   When selecting a file, note the following:

   - The source data file must be a delimited data file.
   - Data files used must contain a one-line header, which describes the delimited columns.
   - Both numeric and non-numeric data can be loaded.
6. **Optional**: In **Prefix**, specify a prefix to make the application name unique.

   The prefix is concatenated with the application name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.
7. Click **OK**.
8. From the Data Integration home page, click (Create), and then select **Integration**.

   The General page is displayed in Create Integration view.
9. Create the integration between the source and target application and then click **Save and Continue**.

   a. In **Name** and **Description**, enter a name and description for the new integration.

   b. In **Location**, enter a new location name, or pick an existing location to specify where to load data.

Copyright © 2017, 2026, Oracle and/or its affiliates.

c. From the **Source** ( ) drop-down, select **File**.

d. From the **File Browser**, navigate to and select the incremental file, and then click **OK**.

When you select the incremental file, click **File Options** to preview and manage the format of the contents on the **File Import page.**

e. From the **Target** ( ), select the target application to which to load the incremental data.

f. From **Cube**, select the plan type of the target system.

g. From **Category**, select the category mappings for categorizing and mapping source system data to a target Scenario dimension member.

The categories listed are those that you created in the setup, such as "Actual." For more information, see Managing Category Mappings.

h. **Optional**: Select any applicable location attributes for the integration. For more information, see Selecting Location Attributes.

i. Click **Save and Continue**.

10. On the **Map Dimensions** page, map dimensions (build an import format) between the source and target.

See Mapping Dimensions.

a. From **Type**, select the delimited file type.

For example, you might select **Delimited - Numeric Data** as the format of the file.

b. From the **File Delimiter** drop-down, select a type of delimiter.

Available delimiters:

- comma (,)
- exclamation (!)
- semicolon (;)
- colon (:)
- pipe (|)
- tab
- tilde (~)

c. In the mappings grid, map the source columns in the source data-load file to the dimensions in the target application.

The dimensions from the target application are populated automatically.

If the import format has already been defined for the file, then the source and target columns are mapped automatically.

If you are adding a new import format or editing an existing import format, complete the following:

- In **Column**, specify the field number from the file to import.
- In **Select Source Dimension**, specify the name of the source dimension to assign to the target application.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Multiple source columns of the same dimension can be mapped to target dimensions. For example, you can map four "Account" source columns.

- Add a source or target expression: assign an expression that operates on values directly from the source or target.

  See Using Source Expressions and Using Target Expressions.

d. **Optional**: For a comma delimited file, select an additional row to map in the import format by clicking to the right of a row and selecting the row to add from the drop-down.

   Available rows:

   - Source Period
     - Year
     - Period
     - Period Number
   - Currency
   - Attribute
   - Description
   - Dimension Row
     - Account
     - Version
     - Entity
     - View

   You can also skip a row.

   For more information, see Creating the Dimension Maps.

e. Click **Save and Continue**.

11. From **Map Members** page, map any members from the source to target.

    For more information, see Mapping Members.

12. From the **Options** page, click the **Filters** tab.

> **Note**
>
> You can also select the Source File, Incremental Processing Options, and Last Process ID options on the Filters tab on the Run Integration page.

13. From **Source File**, select the data file name that contains the data you are loading. It may be the same one from which you created the data source application, or another file that has data as well as the appropriate header.

    Select the file that contains your data, as before. It may have the same name as the original file, or it may have a new name. The differences in the file (i.e., the incremental load file) is created automatically between the two files loaded. So, if file A.txt has 100 rows and file B.txt has 300 rows where the first 100 are identical, your first load should select file A.txt when the ID is 0. The second load will be against file B.txt and the ID automatically points to the load ID that was assigned to A.

Copyright © 2017, 2026, Oracle and/or its affiliates.

14. From **Incremental Processing Options**, select the method for sorting data in the source file.

    Available options:

    - Do not sort source file—Source file is compared as provided. This option assumes that the source file is generated in the same order each time. In this case, the system performs a file comparison, and then extracts the new and changed records. This option makes the incremental file load perform faster.
    - Sort source file—Source file is sorted before performing the file comparison for changes. In this option the source file is first sorted. The sorted file is then compared to the prior sorted version of this file. Sorting a large file consumes a lot system resources and performs slower.

    > **Note**
    >
    > If you have an integration that uses the **Do Not Sort** option and then switch to the **Sort source file** option, then the first load has invalid results since the files are in different order. Subsequent runs load data correctly.

15. The **Last Process ID** shows the last run ID for the original source data file.

    When the load is first run for the original data file, the **Last Process ID** shows the value of **0**.

    When the load is run again, the **Last Process ID** shows the run number of the last load.

    If the newly created file comparison version and the original data file shows no differences, or the file is not found, the value of the **Last Process ID** is assigned to the last load ID that ran successfully.

    To reload all data, set the **Last Process ID** back to **0**, and select a new source file to reset the baseline.

16. Click **Save**.

17. From the **Data Integration** home page, select the integration, and then click .

18. From the **Run Integration** page, select the **Options** tab

19. Complete the following options and click **Run**.

    - Import Mode
    - Export Mode
    - Start Period
    - End Period

    For more information on running an integration, see Running an Integration.

Bulk Loading Workforce New Hire Data

You can integrate new hire bulk data to Workforce. You can load bulk new hires data or change new hire requisition data for multiple new hires at once.

Bulk new hire data is loaded using an Incremental File Adapter. This type of integration compares a source data file with a prior version of the source data file and identifies new or changed records and then loads only that data set.

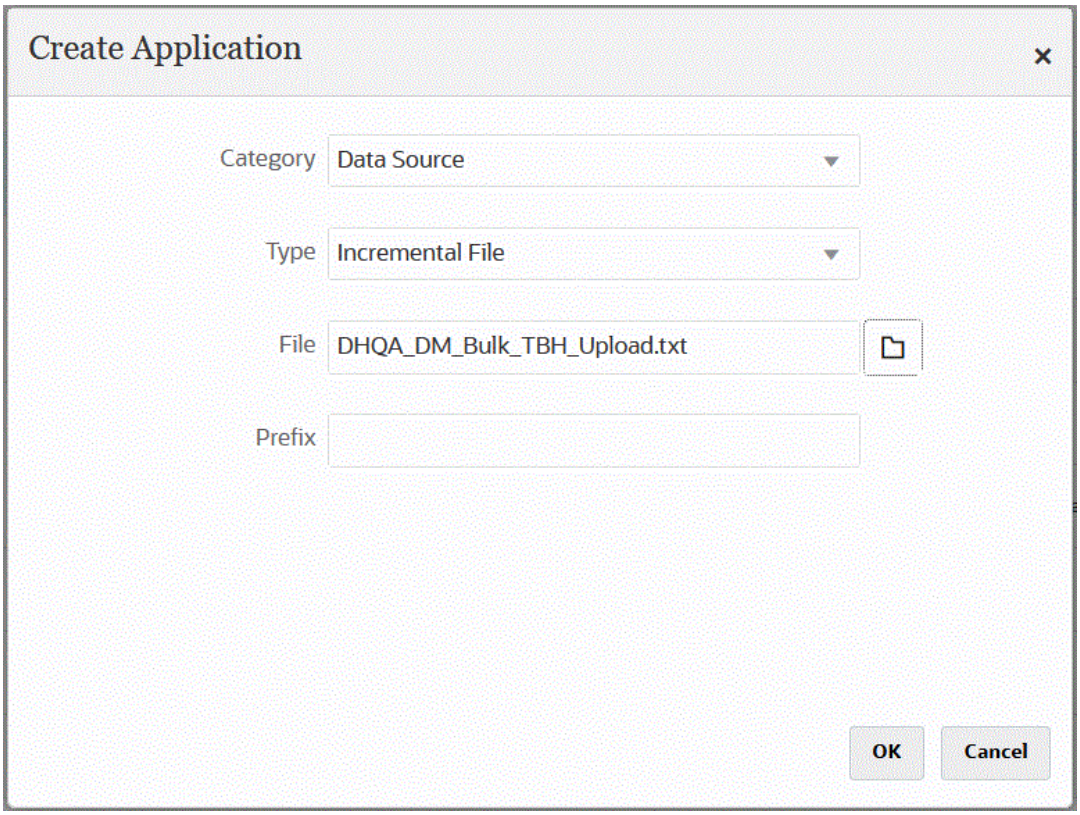
Copyright © 2017, 2026, Oracle and/or its affiliates.

Customers can bulk load new hires without having to manually use **Add Hiring Requisition** to add requisitions one at a time. For more information, see Adding Hiring Requisitions in *Working with Planning Modules* .

Bulk Loading Process Description

To bulk load new hire data to Workforce:

1. Before loading new hire data in Data Integration, check for empty requisitions available in the system for the **Process New Hires** mass update form in Workforce. If you are loading data for a requisition which is already in the system, then the existing requisition data is replaced with a new record loaded through Data Integration.
2. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
3. On the **Applications** page, click .
4. On the **Create Application** page, and then from the **Category** drop-down, select **Data Source**.



5. From **Type**, select **Incremental File**.
6. From **File**, specify the bulk new hire source data file to use as the template.

   Click to search for a file from the **File Browser** page.

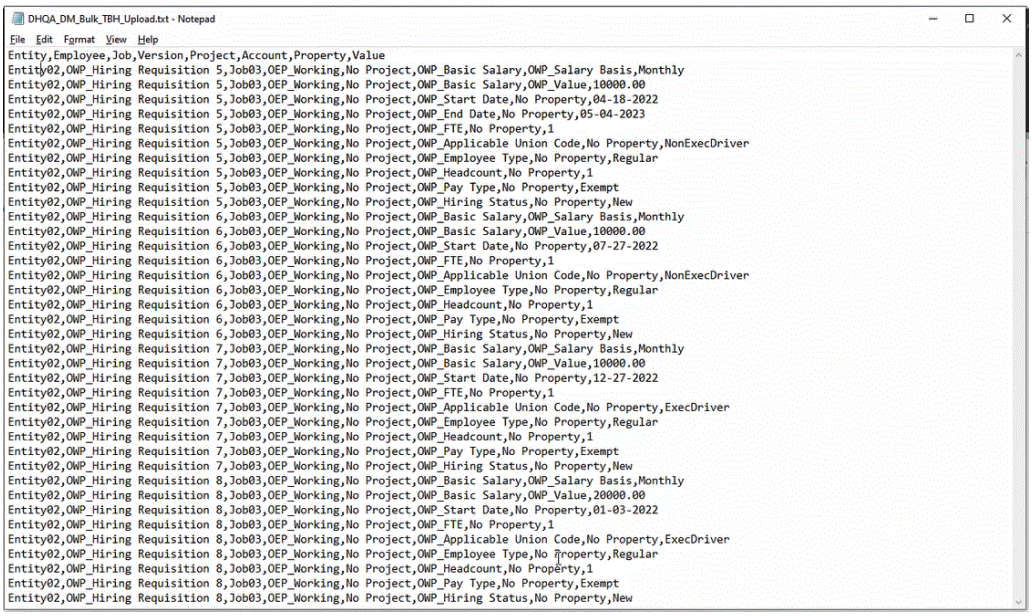
   When selecting a file, note the following:

   - The source data file must be a delimited data file.
   - Data files used must contain a one-line header, which describes the delimited columns.

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Both numeric and non-numeric data can be loaded.

  The following shows a bulk new hire file:



```
DHQA_DM_Bulk_TBH_Upload.txt - Notepad
File  Edit  Format  View  Help
Entity,Employee,Job,Version,Project,Account,Property,Value
Entity02,OWP_Hiring Requisition 5,Job03,OEP_Working,No Project,OWP_Basic Salary,OWP_Salary Basis,Monthly
Entity02,OWP_Hiring Requisition 5,Job03,OEP_Working,No Project,OWP_Basic Salary,OWP_Value,10000.00
Entity02,OWP_Hiring Requisition 5,Job03,OEP_Working,No Project,OWP_Start Date,No Property,04-18-2022
Entity02,OWP_Hiring Requisition 5,Job03,OEP_Working,No Project,OWP_End Date,No Property,05-04-2023
Entity02,OWP_Hiring Requisition 5,Job03,OEP_Working,No Project,OWP_FTE,No Property,1
Entity02,OWP_Hiring Requisition 5,Job03,OEP_Working,No Project,OWP_Applicable Union Code,No Property,NonExecDriver
Entity02,OWP_Hiring Requisition 5,Job03,OEP_Working,No Project,OWP_Employee Type,No Property,Regular
Entity02,OWP_Hiring Requisition 5,Job03,OEP_Working,No Project,OWP_Headcount,No Property,1
Entity02,OWP_Hiring Requisition 5,Job03,OEP_Working,No Project,OWP_Pay Type,No Property,Exempt
Entity02,OWP_Hiring Requisition 5,Job03,OEP_Working,No Project,OWP_Hiring Status,No Property,New
Entity02,OWP_Hiring Requisition 6,Job03,OEP_Working,No Project,OWP_Basic Salary,OWP_Salary Basis,Monthly
Entity02,OWP_Hiring Requisition 6,Job03,OEP_Working,No Project,OWP_Basic Salary,OWP_Value,10000.00
Entity02,OWP_Hiring Requisition 6,Job03,OEP_Working,No Project,OWP_Start Date,No Property,07-27-2022
Entity02,OWP_Hiring Requisition 6,Job03,OEP_Working,No Project,OWP_FTE,No Property,1
Entity02,OWP_Hiring Requisition 6,Job03,OEP_Working,No Project,OWP_Applicable Union Code,No Property,NonExecDriver
Entity02,OWP_Hiring Requisition 6,Job03,OEP_Working,No Project,OWP_Employee Type,No Property,Regular
Entity02,OWP_Hiring Requisition 6,Job03,OEP_Working,No Project,OWP_Headcount,No Property,1
Entity02,OWP_Hiring Requisition 6,Job03,OEP_Working,No Project,OWP_Pay Type,No Property,Exempt
Entity02,OWP_Hiring Requisition 6,Job03,OEP_Working,No Project,OWP_Hiring Status,No Property,New
Entity02,OWP_Hiring Requisition 7,Job03,OEP_Working,No Project,OWP_Basic Salary,OWP_Salary Basis,Monthly
Entity02,OWP_Hiring Requisition 7,Job03,OEP_Working,No Project,OWP_Basic Salary,OWP_Value,10000.00
Entity02,OWP_Hiring Requisition 7,Job03,OEP_Working,No Project,OWP_Start Date,No Property,12-27-2022
Entity02,OWP_Hiring Requisition 7,Job03,OEP_Working,No Project,OWP_FTE,No Property,1
Entity02,OWP_Hiring Requisition 7,Job03,OEP_Working,No Project,OWP_Applicable Union Code,No Property,ExecDriver
Entity02,OWP_Hiring Requisition 7,Job03,OEP_Working,No Project,OWP_Employee Type,No Property,Regular
Entity02,OWP_Hiring Requisition 7,Job03,OEP_Working,No Project,OWP_Headcount,No Property,1
Entity02,OWP_Hiring Requisition 7,Job03,OEP_Working,No Project,OWP_Pay Type,No Property,Exempt
Entity02,OWP_Hiring Requisition 7,Job03,OEP_Working,No Project,OWP_Hiring Status,No Property,New
Entity02,OWP_Hiring Requisition 8,Job03,OEP_Working,No Project,OWP_Basic Salary,OWP_Salary Basis,Monthly
Entity02,OWP_Hiring Requisition 8,Job03,OEP_Working,No Project,OWP_Basic Salary,OWP_Value,20000.00
Entity02,OWP_Hiring Requisition 8,Job03,OEP_Working,No Project,OWP_Start Date,No Property,01-03-2022
Entity02,OWP_Hiring Requisition 8,Job03,OEP_Working,No Project,OWP_FTE,No Property,1
Entity02,OWP_Hiring Requisition 8,Job03,OEP_Working,No Project,OWP_Applicable Union Code,No Property,ExecDriver
Entity02,OWP_Hiring Requisition 8,Job03,OEP_Working,No Project,OWP_Employee Type,No Property,Regular
Entity02,OWP_Hiring Requisition 8,Job03,OEP_Working,No Project,OWP_Headcount,No Property,1
Entity02,OWP_Hiring Requisition 8,Job03,OEP_Working,No Project,OWP_Pay Type,No Property,Exempt
Entity02,OWP_Hiring Requisition 8,Job03,OEP_Working,No Project,OWP_Hiring Status,No Property,New
```

7. **Optional**: In **Prefix**, specify a prefix to make the application name unique.

   The prefix is concatenated with the application name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.

8. Click **OK**.

9. From the Data Integration home page, click (Create), and then select **Integration**.

   The General page is displayed in Create Integration view.

10. Create the integration between the source and target application target and then click **Save and Continue**.

    a. In **Name** and **Description**, enter a name and description for the new integration.

    b. In **Location**, enter a new location name, or pick an existing location to specify where to load data.

    c. From the **Source** ( ) drop-down, select **File**.

    d. From the **File Browser**, navigate to and select the bulk new hire file, and then click **OK**.

       When you select the bulk new hire file, click **File Options** to preview and manage the format of the contents on the **File Import page.**

    e. From the **Target** ( ), select the Workforce application to which to load the bulk new hire data.

    f. From **Cube**, select the plan type of the target system.

    g. From **Category**, select the category mappings for categorizing and mapping source system data to a target Scenario dimension member.

Copyright © 2017, 2026, Oracle and/or its affiliates.

The categories listed are those that you created in the setup, such as "Actual." For more information, see Managing Category Mappings.

h. **Optional**: Select any applicable location attributes for the integration. For more information, see Selecting Location Attributes.

i. Click **Save and Continue**.

11. On the **Map Dimensions** page, map dimensions (build an import format) between the source and target.

a. From **Type**, select **Delimited - All Data Type**.

b. From the **File Delimiter** drop-down, select a type of delimiter.

Available delimiters:

- comma (,)
- exclamation (!)
- semicolon (;)
- colon (:)
- pipe (|)
- tab
- tilde (~)

c. In the mappings grid, map the source columns in the source data-load file to the dimensions in the target application.

The dimensions from the target application are populated automatically.

If the import format has already been defined for the file, then the source and target columns are mapped automatically.

If you are adding a new import format or editing an existing import format, complete the following:

- In **Column**, specify the field number from the file to import.
- In **Select Source Dimension**, specify the name of the source dimension to assign to the target application.

  Multiple source columns of the same dimension can be mapped to target dimensions. For example, you can map four "Account" source columns.
- Add a source or target expression: assign an expression that operates on values directly from the source or target.

  See Using Source Expressions and Using Target Expressions.

d. **Optional**: For a comma delimited file, select an additional row to map in the import format by clicking to the right of a row and selecting the row to add from the drop-down.

Available rows:

- Source Period
  - Year
  - Period
  - Period Number
- Currency

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Attribute
- Description
- Dimension Row
  - Account
  - Version
  - Entity
  - View

You can also skip a row.

For more information, see Creating the Dimension Maps.

e. Click **Save and Continue**.

12. From **Map Members** page, map any members from the source to target.

    To map all members to Workforce "as is" without any modification, select **All** for the **Mapping Type**, click **Add**, and on the **Add Map Member** page, in **Source**, enter: *****, and in **Target**, enter: *****.

    For more information, see Mapping Members.

13. From the **Options** page, click the **Filters** tab.

    a. From **Source File**, select the data file name that contains the data you are loading. It may be the same one from which you created the data source application, or another file that has data as well as the appropriate header.

    Select the file that contains your data, as before. It may have the same name as the original file, or it may have a new name. The differences in the file (i.e., the incremental load file) are created automatically between the two files loaded. So, if file A.txt has 100 rows and file B.txt has 300 rows where the first 100 are identical, your first load should select file A.txt when the ID is 0. The second load will be against file B.txt and the ID automatically points to the load ID that was assigned to A.

    b. From **Incremental Processing Options**, select the method for sorting data in the source file.

    Available options:

    - Do not sort source file—Source file is compared as provided. This option assumes that the source file is generated in the same order each time. In this case, the system performs a file comparison, and then extracts the new and changed records. This option makes the incremental file load perform faster.
    - Sort source file—Source file is sorted before performing the file comparison for changes. In this option the source file is first sorted. The sorted file is then compared to the prior sorted version of this file. Sorting a large file consumes a lot system resources and performs slower.

    > **Note**
    >
    > If you have an integration that uses the **Do Not Sort** option and then switch to the **Sort source file** option, then the first load will have invalid results since the files are in different order. Subsequent runs load data correctly.

    c. The **Last Process ID** shows the last run ID for the original source data file.

Copyright © 2017, 2026, Oracle and/or its affiliates.

When the load is first run for the original data file, the **Last Process ID** shows the value of **0**.

When the load is run again, the **Last Process ID** shows the run number of the last load.

If the newly created file comparison version and the original data file shows no differences, or the file is not found, the value of the **Last Process ID** is assigned to the last load ID that ran successfully.

To reload all data, set the **Last Process ID** back to **0**, and select a new source file to reset the baseline.

> **Note**
>
> You can also select the Source File, Incremental Processing Options, and Last Process ID options on the Filters tab on the Run Integration page.

14. Click **Save**.
15. Select the **Options** tab, and complete the following if necessary:
    a. From **Category**, select the category mappings for categorizing and mapping source system data to a target Scenario dimension member.

       You may need to change the category if you selected another source file for an incremental data load.

    b. From **Plan Type**, select the plan type.
    c. From **Load Method**, select **All data types with security**.
16. Register the **OWP_Incremental Process Data with Synchronize Defaults** or **OWP_Incremental Process Data With Strategic Workforce** business rule in embedded mode.

    The Embedded mode is used to register business rules that are executed after the data load process. This mode is embedded as part of the data load process and provides access to the data intersections loaded. This mode is available only when using the All Data Type load method. Embedded rules are not triggered by an Event and do not support runtime parameters.

    > **Note**
    >
    > Because incremental rules are only executed for updated records, it is necessary to add incremental Groovy rules in the embedded mode. This is because the context of the loaded data are passed to the rule in the embedded mode.

    To register the **OWP_Incremental Process Data with Synchronize Defaults** or **OWP_Incremental Process Data With Strategic Workforce** business rule:

    a. To register a business rule for a given integration, from the **General** page, click to the right of the integration, then select **Options**, and then from the **Edit Integration** page, select the **Business Rules** tab.
    b. From the **Business Rules** page, click the **Embedded** mode.
    c. Click .

Copyright © 2017, 2026, Oracle and/or its affiliates.

d. From **Name**, select the **OWP_Incremental Process Data with Synchronize Defaults** or **OWP_Incremental Process Data With Strategic Workforce** business rule.

   From **Name**, specify the business rule to run.

   The **OWP_Incremental Process Data with Synchronize Defaults** business rule enables you to:

   - Apply entity defaults for additional earnings, benefits and taxes and calculate compensation based on the component definition and rates that are set in the Benefits and Taxes Wizard.
   - Copy changed data (employee/job properties or any change in salary) to future periods in the Year Range.

   The **OWP_Incremental Process Data with Synchronize Defaults** business rule is prebuilt in Calculation Manager and then made available to Data Integration when loading data to your Planning target applications.

   If a business rule name is changed in Planning, the new business rule name is not automatically renamed in Data Integration, which does not validate business rules.

   The **OWP_Incremental Process Data With Strategic Workforce** business rule enables you to improve processing time when loading data.

   The **OWP_Incremental Process Data With Strategic Workforce** business rules available in Data Integration are defined in Strategic Workforce Planning. If you want a business rule to be executed for specific integration, then register the business rule on the Integration Options page. If you want the business rule to be executed for all integration loading data to the application, then register the business rule on the Application page. If a given Event has a business rule registered at both the integration and application levels, only the business rule registered on the Integration Options page is executed.

e. Click **Save**.

17. From the **Data Integration** home page, select the integration, and then click ▶.
18. From the **Run Integration** page, select the **Options** tab.
19. Complete the following options and click **Run**.
    - Import Mode
    - Export Mode
    - Start Period
    - End Period

> **Note**
>
> When you enter a Start Date as a prior year date, then the system resets the Start Date date to the plan Start Year, Start Month, and Start Date for the new hires.

For more information, see Running an Integration.

Defining Incremental File Adapter Options

You specify incremental file adapter options to define or change the incremental file adapter parameters used to compare a source data file with a prior version of the source data file and identify new or changed records and then load only that data set.

Copyright © 2017, 2026, Oracle and/or its affiliates.

To define incremental file adapter options:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.

   From the **Application** page, click to the right of the Oracle ERP Cloud data source application, and then select **Application Details**.

   From the **Application Details** page, click the **Options** tab.

2. **Source File** shows the data file name that contains the data you are loading. It is read only.

3. From **Incremental Processing Options**, select the method for sorting data in the source file. You can sort the initial source data file before making the comparison or provide a pre-sorted file for better performance.

   Available options:

   - Do not sort source file—Source file is compared as provided. This option assumes that the source file is generated in the same order each time. In this case, the system performs a file comparison, and then extracts the new and changed records. This option makes the incremental file load perform faster.
   - Sort source file—Source file is sorted before performing the file comparison for changes. In this option the source file is first sorted. The sorted file is then compared to the prior sorted version of this file. Sorting a large file consumes a lot system resources and performs slower.

   > **Note**
   >
   > If you have an integration that uses the **Do Not Sort** option and then switch to the **Sort source file** option, then the first load will have invalid results since the files are in different order. Subsequent runs load data correctly.

4. The **Last Process ID** shows the last run ID for the original source data file.

   When the load is first run for the original data file, the **Last Process ID** shows the value of **0**.

   When the load is run again, the **Last Process ID** shows the run number of the last load.

   If the newly created file comparison version and the original data file shows no differences, or the file is not found, the value of the **Last Process ID** is assigned to the last load ID that ran successfully.

   To reload all data, set the **Last Process ID** back to **0**, and select a new source file to reset the baseline.

5. Click **Save**.

> **Note**
>
> You can also select the Source File, Incremental Processing Options, and Last Process ID options on the Filters tab on the Run Integration page.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Integrating With Oracle Autonomous Database Data

You can integrate data directly from the Oracle Autonomous Database with Cloud EPM without using an agent. This type of integration enables you to extract source data from staging or other applications running on the Oracle Autonomous Database.

Additionally, you can also use the external table function in the Oracle Autonomous Database to define a table where the source data is from a file in Object Storage and then extract data from it.

See:

- Process Description for Integrating Oracle Autonomous Database
- Connecting to the Oracle Cloud Infrastructure
- Configuring an Oracle Autonomous Database Source Connection
- Creating Database Tables
- Creating Data Source Application
- Setting Up an Integration
- Running Integration to Load Data from Oracle Autonomous Database to Cloud EPM Without an Agent

Process Description for Integrating Oracle Autonomous Database

At a high level, here are the steps for integrating data between the Oracle Fusion Cloud Enterprise Performance Management and the Oracle Autonomous Database without an agent:

1. Connect to the Oracle Cloud Infrastructure Autonomous Database.
2. Download the wallet from the Oracle Cloud Infrastructure.
3. Configure the Oracle Autonomous Database source connection
4. Create database tables.
5. Create data source application.
6. Set up the integration.
7. Run the integration.

See:

- Connecting to the Oracle Cloud Infrastructure
- Configuring an Oracle Autonomous Database Source Connection
- Creating Database Tables
- Creating Data Source Application
- Setting Up an Integration
- Running Integration to Load Data from Oracle Autonomous Database to Cloud EPM Without an Agent

Copyright © 2017, 2026, Oracle and/or its affiliates.

Connecting to the Oracle Cloud Infrastructure

To connect to Oracle Cloud Infrastructure, perform the steps provided in Connecting to the Oracle Cloud Infrastructure.

Configuring an Oracle Autonomous Database Source Connection

You must define a source connection to connect to an Oracle Autonomous Database data source in any of the following ways:

- Defining an Oracle Autonomous Database Source Connection Without Using Object Storage
- Defining an Oracle Autonomous Database Source Connection Using Object Storage

Defining an Oracle Autonomous Database Source Connection Without Using Object Storage

To configure an Oracle Autonomous Database Connection without object storage:

1. On the **Data Integration** home page, from the **Actions** menu, click **Setup**, and then click **Applications**.
2. On the **Applications** page, click (Configure Connections icon).
3. On the **Connections** page, click (Add drop-down icon), and select **Oracle Autonomous DB**.
4. In **Name**, specify the name of the connection to the Oracle Autonomous DB.

   The name should not contain space or hyphens.

   The connection name will be used to store the wallet.
5. In **Description**, specify a description of the connection to the Oracle Autonomous DB.
6. In **Wallet**, click to upload the wallet file for the autonomous database from Oracle Cloud Infrastructure. This is the wallet that you downloaded when connecting to Oracle Cloud Infrastructure.

   A wallet is a password-protected container that stores authentication and signing credentials, including private keys, certificates, and trusted certificates, all of which are used by SSL for strong authentication.

   This step assumes that you are connecting to an autonomous database instance that has already been created. For more information, see Connecting to the Oracle Cloud Infrastructure.
7. In the **DB Username**, enter the Oracle Autonomous Database username.
8. In **DB Password**, enter the Oracle Autonomous Database password. You must update this password anytime you change your Oracle Autonomous Database password.
9. In **DB Service Name**, specify the database service/database instance on that server.
10. In **Object Storage Credentials**, select **None**.
11. Click **Test Connection**.
    After the connection is successful, the "Connection to [source system name] successful" message is displayed.

Copyright © 2017, 2026, Oracle and/or its affiliates.

12. Click **OK**.

Defining an Oracle Autonomous Database Source Connection Using Object Storage

If you plan to import data from a file using the External Table feature in Oracle Autonomous Database, you must configure the Object Storage connection along with the database connection. You can configure the Object Storage connection as a Credential in the Oracle Database or defined in the EPM Connections. For details, see Managing Connections in *Administering Planning* .

If you want to browse the files in Object Storage using the file browser, you must specify the namespace and bucket in the System Settings and/or User Settings. For details, see System Settings and Setting User Level Profiles. If you specify the user settings, then the user settings takes preference over system settings.

To define an Oracle Autonomous Database source connection using object storage:

1. On the **Data Integration** home page, from the **Actions** menu, click **Setup** and then click **Applications**.
2. On the **Applications** page, click (Configure Connections icon).
3. On the **Connections** page, click (Add drop-down icon), select **Oracle Autonomous DB**.
4. In **Name**, specify the name of the connection to the Oracle Autonomous DB.

   The name should not contain space or hyphens.

   The connection name will be used to store the wallet.
5. In **Description**, specify a description of the connection to the Oracle Autonomous DB.
6. In **Wallet**, click to upload the wallet file for the autonomous database from Oracle Cloud Infrastructure. This is the wallet that you downloaded when connecting to Oracle Cloud Infrastructure.

   A wallet is a password-protected container that stores authentication and signing credentials, including private keys, certificates, and trusted certificates, all of which are used by SSL for strong authentication.

   This step assumes that you are connecting to an autonomous database instance that has already been created. For more information, see Connecting to the Oracle Cloud Infrastructure.
7. In the **DB Username**, enter the Oracle Autonomous Database username.
8. In **DB Password**, enter the Oracle Autonomous Database password. You must update this password anytime you change your Oracle Autonomous Database password.
9. In **DB Service Name**, specify the database service/database instance on that server.
10. In **Object Storage Credentials**, select one of the following:
    - **Credential Name**: Specify the **Object Storage URL** and the **Object Storage Credential Name** defined in the Oracle Autonomous Database.
    - **Connection**: Select the required **Object Storage Connection** defined in the Cloud EPM Connection and click **Create Credential**. The credentials are automatically created in the database.
11. Click **Test Connection**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

After the connection is successful, the "Connection to [source system name] successful" message is displayed.

12. Click **OK**.
13. After the object storage connection is established, on the Data Integration page, click **Actions** > **File Browser**. Select the object storage connection on the top-right drop down list. The data from the object storage bucket is displayed. Upload file directly into the required folder.

Creating Database Tables

You can create both External and Standard tables through a centralized UI. An External Table provide access to data stored outside the database engine without physically ingesting the data into the database. A Standard table stores data physically within the database itself and operates independently without relying on any external data source or linked storage location.

- Creating External Tables
- Creating Standard Tables

Creating External Tables

To create an external table in the Oracle Autonomous Database:

1. On the **Data Integration** home page, from the **Actions** menu, click **Setup** and then click **Applications**.
2. On the **Applications** page, click **Actions** and then click **Create Table**.
3. Select **Table Type** as **External Table**.
4. Specify the following:
   - **Table Name**: Specify a name for the table.
   - **Connection**: Select the Oracle Autonomous Database source connection.
   - **Upload File**: Upload the file that has the data. The file will be used to derive the table structure and the table data. If the uploaded file has header, then it uses the header as column names in the table.
   - **Delimiter**: Select the delimiter used in the file.

   > **Note**
   >
   > The **Namespace** and **Bucket Name** fields are automatically populated from the user settings or system settings. If both are available, the user settings take precedence. You can also change these fields to any desired namespace or bucket, including any subdirectories within the bucket.
   > The file uploaded in the UI must also be manually copied to the selected namespace and bucket location, including any applicable subdirectories.

5. Click **Next** and adjust the **Column Name**, **Data Type**, **Column Length**, **Date Format**, and **Is Nullable** details, if required.
6. Click **Finish**.
   A message is displayed indicating the table is created successfully.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Creating Standard Tables

To create a standard table in the Oracle Autonomous Database:

1. On the **Data Integration** home page, from the **Actions** menu, click **Setup** and then click **Applications**.
2. On the **Applications** page, click **Actions** and then click **Create Table**.
3. Select **Table Type** as **Standard Table**.
4. Specify the following:
   - **Table Name**: Specify a name for the table.
   - **Connection**: Select the Oracle Autonomous Database source connection.
   - **Upload File**: Upload the file that will be used to derive the table structure. The data must be loaded to the table separately. If the uploaded file has header, then it uses the header as column names in the table.
   - **Delimiter**: Select the delimiter used in the file.

   > **Note**
   >
   > The **Namespace** and **Bucket Name** fields are disabled.

5. Click **Next** and adjust **Data Type** and **Data Length**, if required.
6. Click **Finish**.
   A message is displayed indicating the table is created successfully.

Creating Data Source Application

A query-based data source application uses a custom SQL query instead of a direct table.It allows the user to specify a query and connection, and requires a template file to define the query columns. A table-based datasource application directly uses a database table as its primary data source. It allows the user to select an existing table and then provide a filter that is applied at run time.

- Creating Query-based Data Source Application
- Creating Table-based Data Source Application

Creating Query-based Data Source Application

Before creating the query-based data source application, you must create a query. For details, see Creating an SQL Query. The parameters you specify in the query must match the column names.

To create a query-based data source application:

1. On the **Data Integration** home page, from the **Actions** menu, click **Setup** and then click **Applications**.
2. Click (Add icon).
3. Specify the following:
   - **Category**: Select **Data Source**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

- **Type**: Select **Oracle Autonomous Database – Direct (Query)**.
- **File**: Upload the header file.
- **Prefix**: Specify a prefix to make the application name unique. The prefix is concatenated with the application name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.

4. Click **Save**.
5. Edit the application details and the integration-level filters. For details, see:
   - Editing Application Details
   - Editing Integration-Level Filters Associated With Oracle Autonomous DB Data Source

Creating Table-based Data Source Application

To create a table-based data source application:

1. On the **Data Integration** home page, from the **Actions** menu, click **Setup** and then click **Applications**.
2. Click (Add icon).
3. Specify the following:
   - **Category**: Select **Data Source**.
   - **Type**: Select **Oracle Autonomous Database – Direct (Table)**.
   - **Connection**: Select Oracle Autonomous Database source connection.
   - **Application**: Provides the list of tables from the database. Select the required table.
   - **Prefix**: Specify a prefix to make the application name unique. The prefix is concatenated with the application name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.
4. Click **Save**.
5. Edit the application details and the integration-level filters. For details, see:
   - Editing Application Details
   - Editing Integration-Level Filters Associated With Oracle Autonomous DB Data Source

Editing Application Details

After you create a query-based or table-based data source application, you can edit the details such as dimensions and options, and set defaults.

To edit application details:

1. On the Applications page, click the application name with **Type** as **Oracle Autonomous Database – Direct (Query)** or **Oracle Autonomous Database – Direct (Table)**. You can also click the (**…**) **Actions** icon and then click **Application Details**.
   The Application Details window displays all the fields from the file as dimensions.
2. On the **Dimensions** tab:
   a. Select the **Dimension Classification** to reclassify the dimensions.

Copyright © 2017, 2026, Oracle and/or its affiliates.

b. Click (Add icon) to add a dimension if it isn't defined in the application. A row is added at the end of the table. Specify a dimension name and select the **Dimension Classification**.

c. Select the dimension and click (Delete icon) to delete a dimension.

3. On the **Options** tab:
   - **Data Extract Query**: Specify SQL query name to run against the file.

     > **Note**
     >
     > This property is displayed only for query-based applications.

   - **Delimiter**: Select the delimiter type used in the file. Available delimiters are:
     - Comma (,)
     - Exclamation (!)
     - Semicolon (;)
     - Colon (:)
     - Vertical bar (|)
     - NA
   - **Fetch Size**: Specify the number of rows fetched (that the driver processes) with each database round trip for a query.
   - **Connection name**: Select the Oracle Autonomous Database source connection.
4. Click **Save**.
   The application is ready for use with Data Integration.

Editing Integration-Level Filters Associated With Oracle Autonomous DB Data Source

You can specify additional application options to return only records that meet selected conditions when using Oracle Autonomous database data sources. You can specify a single filter condition or multiple conditions, and specify the exact values to return.

To apply additional application options for an Oracle Autonomous Database data source:

1. On the **Data Integration** home page, from the **Actions** menu, click **Setup** and then click **Applications**.
2. On the Application page, click next to Oracle Autonomous Database data source application and click **Edit Options**.
3. On the Edit Options page, click **Add**. Blank entry fields display.
4. In **Name**, enter the name of the filter.
   For example, enter **Location** or **Period**.

   The name you enter is a bind variable or a placeholder for actual values in the SQL statement. In the SQL query, bind variables must be enclosed in ~~ characters. For example, to use "Period" as a bind variable, specify: ~PERIOD~. The name must exactly match the name specified in the SQL query.
5. In **Display Prompt**, specify the name of the display prompt for the filter on the Edit Integration page in Data Integration.

Copyright © 2017, 2026, Oracle and/or its affiliates.

6. In **Display Order**, specify the display order of the filter on the Edit Integration page.
   If this field is blank, the custom filter cannot be displayed, and the default value is used as the filter value.

   For example, enter 99 to show the filter in the 99th position sequence or position in a list of filters. Display orders are listed from lowest to highest.
7. In **Property Level**, select the type of process at which the filter is displayed.
   Available options:
   - Application
   - Application and Integration
   - Integration
8. In **Validation Type**, to provide a drop-down of values listed using the lookup type, in Validation Type, select one of the following:
   - None
   - Number
   - Lookup Validate
   - Lookup No Validate
   - Yes/No
   - Date
   - Query
   - Choice list— You can enter a list of values in the Validation Object field. Each value is entered on a new line. The list of values (LOV) on the Option page shows the values as a list.
9. In **Validation Object**, enter a list of lookup objects to be validated by type. To enter a list of values if you chose the **Choice list** validation type above, click the pencil icon, then enter the list of values on the Validation Object page, and then click **OK**.
10. Click **OK**.

Setting Up an Integration

To set up an integration:

1. On the **Data Integration** home page, click (Add drop-down icon) and then click **Integration**.
2. On the Create Integration window, specify the **Name**, **Location**, **Description**, **Quick Mode**, **Source**, **Target**, and **Category**. For details, see Creating Direct Integrations.
3. Click **Save and Continue**.
4. In **Map Dimensions**, map the columns in the data source to dimensions in the target application. For details, see Mapping Dimensions .
5. Click **Save and Continue**.
6. In **Map members**, map the dimensions to identify how source field values translate to target dimension members. For details, see Mapping Members.
7. Click **Save and Continue**.
8. In **Options**, define the options for importing and exporting data. Also, define any source filters. For details, see Setting Data Integration Options.

Copyright © 2017, 2026, Oracle and/or its affiliates.

9. Click **Save**.

To run the integration, see Running Integration to Load Data from Oracle Autonomous Database to Cloud EPM Without an Agent.

Running Integration to Load Data from Oracle Autonomous Database to Cloud EPM Without an Agent

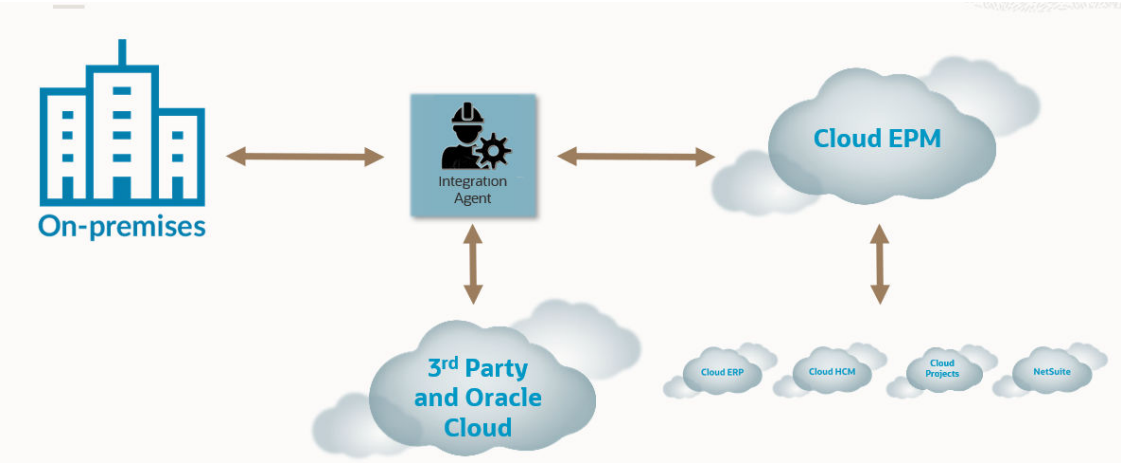
To load data from the Oracle Autonomous Database to Cloud EPM without an agent, run the integration. See Running an Integration.

Copyright © 2017, 2026, Oracle and/or its affiliates.

18

EPM Integration Agent

You can extract data and metadata from your on-premises data source and then load the data directly to the Oracle Fusion Cloud Enterprise Performance Management using the EPM Integration Agent. The EPM Integration Agent executes a query against an on-premises relational database and then loads the data or metadata to the Cloud EPM. The EPM Integration Agent is defined as a data source for an integration. The EPM Integration Agent may also be extended by way of scripting to access other data sources including third-party REST APIs, non-relational sources, or any system or systems that can be accessed using a Jython, Groovy, or Java program.



The EPM Integration Agent can be deployed in two modes: synchronous and asynchronous. The mode determines how communication is opened between your on-premises data and the Cloud EPM. When an integration is executed in synchronous mode, the Cloud EPM initiates a direct call from the Cloud EPM to the appropriate agent based on the cluster assignment. When an integration is executed in asynchronous mode, the Cloud EPM queues for the integration job to execute. The agent polls at intervals and executes the jobs queued.

You can allocate the integration jobs to different clusters by defining cluster assignments. A cluster enables you to manage the distribution of integration jobs. When an integration job is initiated, the systems checks where a job has been assigned, determines the cluster, and assigns the job to that cluster. The order of precedence for executing a job is determined by the cluster.

This chapter is divided into two sections. The Installing and Configuring the EPM Integration Agent section describes the steps required to configure and deploy the EPM Integration Agent in your enterprise. The Cloud EPM to On-premises Connectivity Using the EPM Integration Agent section describes how to connect to and pull data from your on-premises data sources using the EPM Integration Agent paired with an on-premises data adapter. It also explains how to distribute integration jobs using clusters and assignments in synchronous and asynchronous modes.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Your Goal	Watch This Video
Learn how to integrate your source data using the EPM Integration Agent	Leveraging Source System Data With the EPM Integration Agent
Learn how to manage data using the EPM Integration Agent	**Managing Data with the EPM Integration Agent**

Installing and Configuring the EPM Integration Agent

Describes the setup and configuration tasks that you need to complete to deploy the EPM Integration Agent in the Oracle Fusion Cloud Enterprise Performance Management

This section describes the setup and configuration tasks that you need to complete to deploy the EPM Integration Agent in the Cloud EPM.

Quick Reference:

- Configuring and Starting the EPM Integration Agent Process Description
- Downloading the EPMAgent ZIP
- Creating Application Folders
- Encrypting the Cloud EPM User Password
- Configuring the EPM Integration Agent Startup Parameters
- Running the EPM Integration Agent as a Windows Service
- Starting the EPM Integration Agent
- Writing Back with the EPM Integration Agent
- Configuring Synchronous Mode
- Cloud EPM to On-premises Connectivity Using the EPM Integration Agent

Configuring and Starting the EPM Integration Agent Process Description

Describes the setup and configuration tasks that you need to complete to deploy the EPM Integration Agent in the Oracle Fusion Cloud Enterprise Performance Management

The steps below are a high-level description on configuring the EPM Integration Agent. It includes configuration steps and then steps for starting the EPM Integration Agent.

To configure and the EPM Integration Agent:

1. Set up the `JAVA_HOME` environment variable.

   For more information, see Setting JAVA_HOME.

> **Note**
>
> The EPM Integration Agent runs on Windows and Linux with Java 8, 11, and higher. The current version is Java 19. For more information about Java 8, see https://www.oracle.com/java/technologies/java8.html.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> The EPM Integration Agent is only certified with Oracle Java. The EPM Integration Agent is **not** certified with Open JDK.

2. From the Data Integration home page, download the **EPMAgent ZIP** file.

   For more information, see: Downloading the EPMAgent ZIP.

3. Extract `EPMAgent.zip`.

4. Create the cluster that will be used to direct how jobs are extracted when placed in a queue for execution.

   For more information, see Understanding Agent Clusters and Adding a Cluster.

   The default cluster name is EPMCLUSTER.

5. By default, the EPM Integration Agent includes the JDBC drivers for Oracle.

   If you require the Microsoft JDBC driver for SQL Server, see Configuring the Microsoft JDBC Driver for SQL Server in the EPM Integration Agent.

   If you require JDBC drivers for other sources, then download and install the drivers.

6. Create the Application folders.

   For more information, see Creating Application Folders.

7. Use the **EPM Agent Utility** to encrypt the Cloud EPM user password.

   Copy and save the encrypted password. (You add it as a parameter in the next step.)

   For more information, see Encrypting the Cloud EPM User Password.

8. Configure the EPM Integration Agent startup parameters.

   For more information, see Configuring the EPM Integration Agent Startup Parameters.

9. Run the EPM Integration Agent as a Windows service.

   For more information, see Running the EPM Integration Agent as a Windows Service.

   If you are not using a Windows service to start the agent, then instead use the bat file described in the next step.

10. Start the EPM Integration Agent.

    For more information, see Starting the EPM Integration Agent.

11. Configure synchronous mode.

    For more information, see Configuring Synchronous Mode.

12. Select the integration flow.

    The integration flow can either be in synchronous mode or asynchronous mode. The mode defines how the agent interacts with the Cloud EPM: either by direct calls from the Cloud EPM to the EPM Integration Agent, or polling by the EPM Integration Agent to determine when a Cloud EPM request for data has been submitted.

Upgrading the EPM Integration Agent

Describes how to upgrade the EPM Integration Agent in the Oracle Fusion Cloud Enterprise Performance Management

Copyright © 2017, 2026, Oracle and/or its affiliates.

When performing the initial installation of the agent, it is recommended to create the application folders in a location that is not under the integration agent home location. In addition, it is recommended to make a copy of the existing agent installation folders prior to updating to the latest version of the agent.

To upgrade the EPM Integration Agent:

1. From the Data Integration home page, click **Application** ( ).
2. Click **Data Exchange** ( ), and then select the **Data Integration** tab.
3. Optionally, you can launch Data Integration by clicking **Navigator** ( ), and then from **Application**, selecting **Data Exchange** ( ).
4. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Configure** actions drop-down, select **Download Agent**.
5. From the **File Manager** page, extract the **EPMAgent.zip** file to the existing agent location or a different location.

   If you extract to a different location, you will need to update any *ini* files that were previously used.

Setting JAVA_HOME

Describes how to set `JAVA_HOME` on Windows or UNIX systems.

The `JAVA_HOME` environment variable points to the directory where the Java runtime environment (JRE) is installed on your computer.

The following tasks provide the information you need to set `JAVA_HOME` on Windows or UNIX systems.

To set `JAVA_HOME` on a Windows system:

1. Open **File Explorer**, right click **This PC**, and then select **Properties**.
2. On the **System > About** page, click **Advanced system setting**.
3. On the **System Properties** page, click **Environment Variables**.
4. On the **Environment Variables**, under **System Variables** select **JAVA_HOME** and click **Edit**.
5. On the **Edit System Variable** page, then **Variable Value**, specify the location of the Java Runtime Environment (JRE).

   For example, you might specify: `C:\Program Files\Java\jdk1.8\jre`

   JRE is part of the Java Development Kit (JDK) but can be downloaded separately.
6. Click **OK**.

To set `JAVA_HOME` on a UNIX system:

For Korn and bash shells, specify:

```
export JAVA_HOME=jdk-install-dir
export PATH=$JAVA_HOME/bin:$PATH
```


Copyright © 2017, 2026, Oracle and/or its affiliates.

For a Bourne shell, specify:

```
JAVA_HOME=jdk-install-dir
export JAVA_HOME
PATH=$JAVA_HOME/bin:$PATH
export PATH
```

For a C shell, specify:

```
setenv JAVA_HOME jdk-install-dir
export JAVA_HOME
PATH=$JAVA_HOME/bin:$PATH
export PATH
setenv PATH $JAVA_HOME/bin:$PATH
export PATH=$JAVA_HOME/bin:$PATH
```

Java Runtime Environment and EPM Automate

Installing EPM Automate on Windows installs the required Java Runtime Environment (JRE). However, a JRE is not included in the Linux, Unix, and macOS X installers. You must have access to a JRE installation (version 8 through version 11) to use EPM Automate.

You are entitled to use Oracle Java Standard Edition (SE) with EPM Automate without the need to separately purchase a Java SE subscription. For details about Oracle JDK licensing with EPM Automate, refer to Oracle Support Document 1557737.1: "Support Entitlement for Java SE When Used As Part of Another Oracle Product" .

Downloading the EPMAgent ZIP

Describes how to download the EMPAgent ZIP file.

To download the EMPAgent ZIP file:

1. From the Home page, click **Application** ( ).
2. Click **Data Exchange** ( ), and then select the **Data Integration** tab.
3. Optionally, you can launch Data Integration by clicking **Navigator** ( ), and then from **Application**, selecting **Data Exchange** (Data Exchange).
4. From the Data Integration home page, from the **Actions** menu, then from the **Configure** actions drop-down, select **Download Agent**.
5. From the **File Manager** page, extract the **EPMAgent ZIP** to your destination folder.

   The destination folder is *AGENT_HOME*.

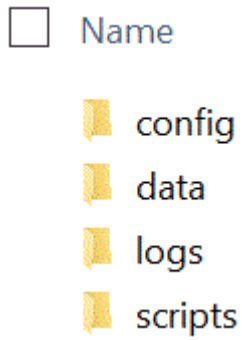
Creating Application Folders

Describes how to download and extract the `EPMAgent.zip` to your *Agent Home* directory.

After downloading and extracting the `EPMAgent.zip` to your *Agent Home* directory, use the `createAppFolder.bat` to create the Application folder structure to store data, logs, and configuration files. The configuration file contains all initialization parameters.

Copyright © 2017, 2026, Oracle and/or its affiliates.

The following image shows the Application folder that contains folders named: config, data, logs, and scripts.



To create the Application folder structure:

1. At a command prompt, change the working directory by typing: `cd AGENT HOME/bin directory`.

   AGENT HOME is the directory where you extracted the `EPMAgent.zip`.

2. At the command prompt, type: `createAppFolder.bat C:\EPMAgentData`.

   When the `createAppFolder.bat` has finished running, the following message displays: "Create application folders script completed."

   When you run the `createappfolder.bat`, the script creates an INI file that contains the populated `EPM_AGENT_HOME` and `EPM_APP_DATA_HOME` parameters.

Secure Sockets Layer (SSL) Certificate

> **Note**
>
> Starting in the release 22.07 update, when you install the EPM Integration Agent, the `cert` folder containing the Secure Sockets (SSL) certificate is no longer downloaded. Certificates are now included automatically in the `JAVA_HOME` identified in the environment variables. An empty cert sub folder is still installed when the agent is downloaded.
> If you plan to use an older version of the agent (prior to the 22.07 update), then retain any existing certificates in the `cert` folder. This also applies if you use any custom certificates.

Configuring the Microsoft JDBC Driver for SQL Server in the EPM Integration Agent

This section describes how to download and configure the Microsoft JDBC Driver for SQL Server driver.

If you plan to use the Microsoft Java Database Connectivity (JDBC) driver to connect to the SQL Server in the EPM Integration Agent, this section describes how to download and configure the driver.

To configure the Microsoft JDBC Driver for SQL Server:

Copyright © 2017, 2026, Oracle and/or its affiliates.

1. Download the SQL server jar from the following website and unzip it: Download Microsoft JDBC Driver for SQL Server

   The driver must be a JRE8 compatible driver.

2. Copy the jar file `mssql-jdbc-6.4.0.jre8.jar` to the `EPM_AGENT_HOME/lib` directory.

   For example, copy the jar file to the `C:\EPMAgent\lib` directory.

3. Specify the location of the `mssql-jdbc-6.4.0.jre8.jar` in the EPM Integration Agent by setting the **CUSTOM_CLASS_PATH** parameter in the `agentparams.ini` file.

   For example, specify: `CUSTOM_CLASS_PATH=C:\EPMAgent\lib\mssql-jdbc-6.4.0.jre8.jar`

   For more information about setting parameters in the `agentparams.ini` file, see Configuring the EPM Integration Agent Startup Parameters.

4. Specify the **JDBC_URL** in the Data Source application filter.

   THE JDBC_URL must be in the format:

   ```
   jdbc:sqlserver://server:port;DatabaseName=dbname
   ```

5. Specify the **JDBC_URL** in the application filter.

   THE JDBC_URL must be in the format:

   ```
   jdbc:sqlserver://server:port;DatabaseName=dbname
   ```

Configuring a SSL Connection to an Oracle Database

Describes how to set the JDBC URL in applications.

The EPM Integration Agent supports an SSL (Secure Sockets Layer) connection to an Oracle database. The SSL protocol provides network-level authentication, data encryption, and data integrity.

To use a SSL connection to an Oracle database, you need to set the JDBC URL in Application options and define custom JAVA properties in the EPM Agent INI file.

> **Note**
>
> A SSL connection is supported only for an Oracle Database.

To set the JDBC URL in applications:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.

2. On the **Application** page, click next to the Oracle database data source application and then select **Application Details**.

3. On the **Application Details** page, click the **Options** tab.

4. In **JDBC URL**, specify the JDBC driver URL connection string format for the SSL connection to the Oracle database.

   For an Oracle Thin JDBC driver, the JDBC driver URL for a service is connected to an Oracle database using this format:

   ```
   jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS=(PROTOCOL=TCPS)(HOST=<hostname>)
   (PORT=<portnumber>))(CONNECT_DATA =(SERVICE_NAME=<servicename>)))
   ```


Copyright © 2017, 2026, Oracle and/or its affiliates.

The JDBC driver URL for a SID (security identifier) is connected to an Oracle database using this format:

```
jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS=(PROTOCOL=TCPS)(HOST=<hostname>)
(PORT=<portnumber>))(CONNECT_DATA =(SID=<sid>)))
```

5. Click **Save**.

To set the custom Java option:

1. Go to the `C:\EPMAgentData\config` directory.
2. Double click the `agentparams.ini` file and open it using a text editor such as Notepad.
3. In the **CUSTOM_JAVA_OPTIONS=** field, specify the following:

```
Djdbc.keyStore=<keystore jks file location> -Djdbc.keyStorePassword=<encrypted
key store password> -Djdbc.keyStoreType=JKS -Djdbc.trustStore=<trust store jks
file location> -Djdbc.trustStorePassword=<encrypted trust store password> -
Djdbc.trustStoreType=JKS
```

> **Note**
>
> Passwords must be encrypted using the encryptpassword utility in the agent. For more information, see Encrypting the Cloud EPM User Password.

4. Save the `agentparams.ini` file.

Configuring a Custom JDBC Driver for SQL Data Sources in EPM Integration Agent

Describes how to configure a Type 3 or Type 4 compliant JDBC driver for the SQL Server.

When using the EPM Integration Agent, JDBC drivers that are Type 3 and Type 4 compliant can now be used to establish a connection to the data source, execute a query, and then extract the results. The system provides only a framework for using the JDBC driver but does not certify individual drivers. If a driver has special requirements, the customer still needs to write custom scripts.

To configure a Type 3 or Type 4 compliant JDBC driver for the SQL Server:

1. Download your Type 3 or Type 4 JDBC driver.
2. Copy the jar file to the `EPM_AGENT_HOME/lib` directory.

   For example, you might copy and paste: `mysql-connector-java-8.0.22.jar` to `C:\EPMAgent\lib` directory.
3. Specify the location of the **jar** file in the EPM Integration Agent by setting the **CUSTOM_CLASS_PATH** parameter in the `agentparams.ini` file.

   For example, specify: `CUSTOM_CLASS_PATH=../lib/mysql-connector- java-8.0.22.jar`

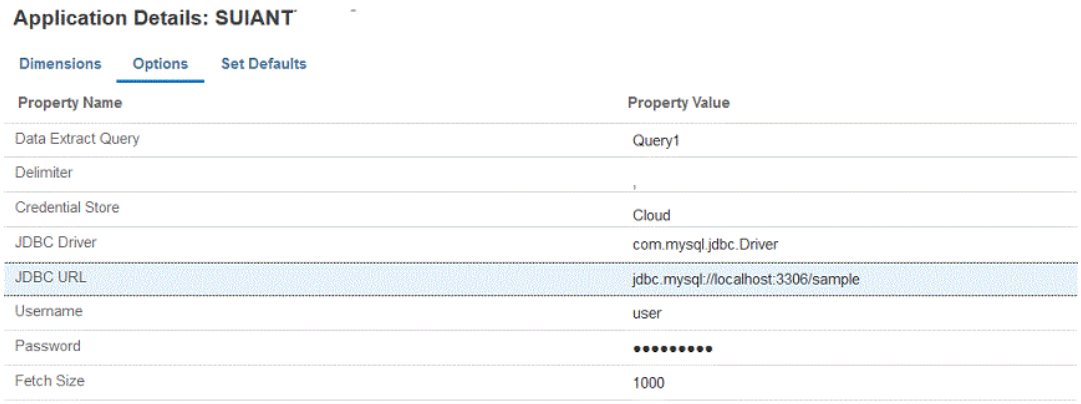
   For more information about setting parameters in the `agentparams.ini` file, see Configuring the EPM Integration Agent Startup Parameters.
4. Specify the **JDBC_URL** in Application Details.

   THE JDBC_URL must be in the format:

   ```
   jdbc:sqlserver://server:port;DatabaseName=dbname
   ```


Copyright © 2017, 2026, Oracle and/or its affiliates.

The following example shows how you might populate the JDBC Driver and JDBC URL:



Configuring the EPM Integration Agent Startup Parameters

Describes how to configure the startup parameters used by the EPM Integration Agent.

Startup parameters for your Oracle Fusion Cloud Enterprise Performance Management URL and Cloud EPM domain for EPM Integration Agent, are defined in the `agentparams.ini` file.

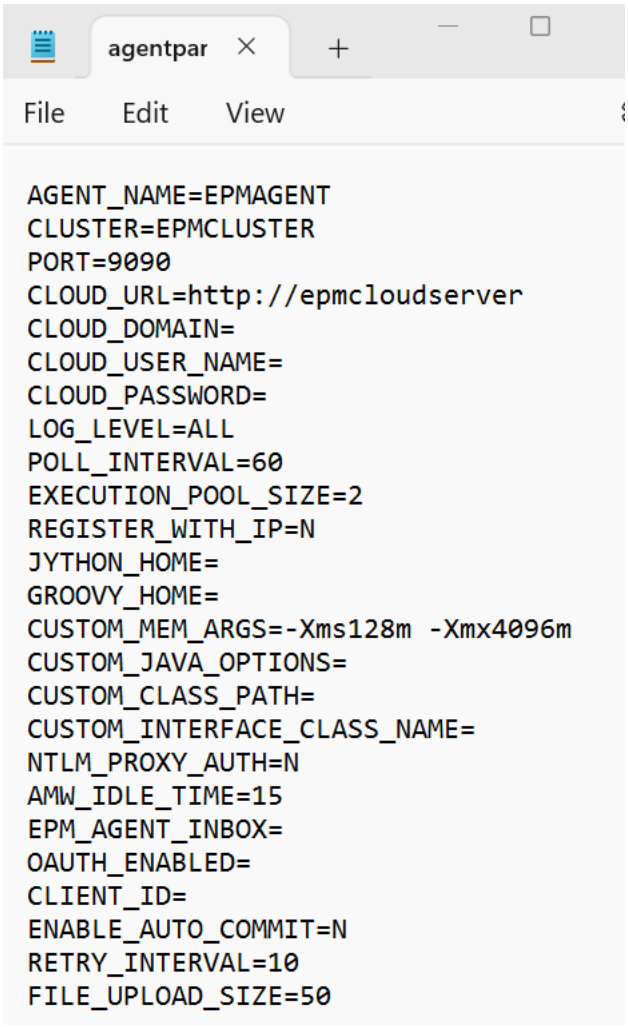
> **Note**
>
> Before defining the EPM Integration Agent startup parameters, run the `createAppFolder.bat`, which creates an INI file that contains the populated `EPM_AGENT_HOME` and `EPM_APP_DATA_HOME` parameters. For more information, see Creating Application Folders.
> To encrypt the password used by the Cloud EPM user, you must run `ecryptpassword.bat`. For more information, see Encrypting the Cloud EPM User Password.

To configure the startup parameters used by the EPM Integration Agent:

1. Go to the `C:\EPMAgentData\config` directory.
2. Double click the `agentparams.ini` file to open it using a text editor such as Notepad.

   `agentparams.ini` file opens in a text file:

Copyright © 2017, 2026, Oracle and/or its affiliates.



3. `EPM_AGENT_HOME` is the name of the home directory of the agent. This folder contains the bin and lib folders.

   When you run `createappfolder.bat`, the script creates the populated `EPM_AGENT_HOME` parameter automatically.

   If you change the name of the folder, the agent runs from the new location and files are created in the new `EPM_APP_DATA_HOME` folder.

4. `EPM_APP_DATA_HOME` is the name of the Application data folder, which contain the config, log, and script folders. The config folder also contains an empty certificate folder, which is not used at this time.

   When you run the `createappfolder.bat` the script creates an INI file that contains the populated `EPM_APP_DATA_HOME` parameter.

5. In **AGENT_NAME**, specify a unique name for the EPM Integration Agent.

   Only characters and numbers can be used in name. The *`AGENT_NAME`* cannot contain any symbols or spaces.

6. In **CLUSTER**, specify the name of the cluster to which the agent belongs.

   Only alphanumeric characters can be used in the name. Do not use special characters such as sign (@) or ampersand (&). The name cannot be modified once the cluster has been created.

   The default cluster name is **EPMCLUSTER**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> The name of the cluster must already be defined in Data Integration before it can be referenced here. For more information, see Adding a Cluster.

This parameter is required.

7. In **PORT**, specify the port number where the EPM Integration Agent runs.

   This parameter is required.

8. In **CLOUD_URL**, specify the Cloud EPM URL used to authenticate and start the agent.

   This is the URL of the EPM instance associated with the agent, for example:

   `https://example-pbcs.us1.oraclecloud.com`

   > **Note**
   >
   > Be sure not to include "epmcloud" or "HyperionPlanning" at the end of the Cloud_URL.

   This parameter is required.

9. In **CLOUD_DOMAIN**, specify the unique name for the Cloud EPM URL.

   An identity domain controls the accounts of users who need access to service instances. It also controls the features that authorized users can access. A service instance belongs to an identity domain.

   Administrators can update the domain name that is presented to the user, but Data Management requires the original domain name that was provided when the customer signed up for the service. Alias domain names cannot be used when setting up Cloud EPM connections from Data Management.

   > **Note**
   >
   > The CLOUD_DOMAIN parameter is required except for any of the Gen 2 data centers. When using the EPM Integration Agent with Gen 2 data centers, you can resolve any errors by removing the domain in the `CLOUD_DOMAIN` field.

10. In **CLOUD_USER_NAME**, specify the administrator user name present in the Cloud EPM environment.

    The user name must be a native user and not an SSO user name.

    This parameter is required.

11. In **CLOUD_PASSWORD**, specify the encrypted password string for the administrator user.

    You can encrypt the password for the Cloud EPM user and get the encrypted password string by running the `encryptpassword.bat` (for Windows) or `ecryptpassword.sh` (for Linux). For more information, see Encrypting the Cloud EPM User Password.

    This parameter is required.

12. In **LOG_LEVEL**, specify the local EPM Integration Agent log level.

    The valid log levels include the following:

Copyright © 2017, 2026, Oracle and/or its affiliates.

- ALL—Shows the most detail level and prints all log statements.
- INFO—Prints selected logs which are important.
- ERROR—Performs minimal logging, only fatal errors are printed.

All logs are written to a file within the *APPDATA_HOME*\logs folder.

This parameter is optional. **All** is the default log level.

13. In **POLL_INTERVAL**, specify the time interval to wait and poll in asynchronous mode.

    The time interval is specified in seconds. The default time interval is 120 seconds.

14. In **REGISTER_WITH_IP**, specify **N** to register the agent with the host name. Specify **Y** to register the agent with the IP address.

    This parameter is optional.

15. In **EXECUTION_POOL_SIZE**, specify how many jobs can be run in parallel.

    The default execution pool size is 2.

    This parameter is optional.

16. In **JYTHON_HOME**, specify the top-level directory where Jython is installed.

    This is required only if you are using Jython as a scripting language.

17. In **GROOVY_HOME**, specify the top-level directory where Groovy is installed.

    This is required only if you are using Groovy as a scripting language.

18. In **CUSTOM_MEM_ARGS**, specify the variable to override the standard memory arguments passed to Java with a custom memory argument.

    For example, in the parameter `CUSTOM_MEM_ARGS=-Xms128m -Xmx4096m`, 128 MB is the minimum initial memory size and 4096 MB is the maximum memory size.

    This parameter is optional.

19. In **CUSTOM_JAVA_OPTIONS**, specify any additional Java runtime parameters.

    The CUSTOM_JAVA_OPTIONS setting can also be configured to have the EPM Integration Agent use a proxy authentication method.

    The system supports the following authentication methods:

    - simple
    - basic
    - digest
    - NTLM

    If you use a proxy authentication method, specify the proxy host name, proxy port, proxy user name, and encrypted password in this field based on the proxy authentication method.

    If the proxy uses basic authentication, then set **CUSTOM_JAVA_OPTIONS** to:

    ```
    -Djdk.http.auth.tunneling.disabledSchemes=""
    ```


Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> The Basic authentication scheme has been deactivated by default in Oracle Java Runtime when you add Basic to the **jdk.http.auth.tunneling.disabledSchemes** networking property. As a result, proxies requiring Basic authentication when setting up a tunnel for HTTPS no longer succeed by default. If required, this authentication scheme can be reactivated by removing it from the jdk.http.auth.tunneling.disabledSchemes networking property.

Simple Proxy Authentication Method:

To enable simple proxy authentication, set **NTLM_PROXY_AUTH** field above to **N** and include the following parameters in **CUSTOM_JAVA_OPTIONS**:

For HTTP, specify: `-Dhttp.proxyHost=proxy.example.com -Dhttp.proxyPort=80`

For HTTPS, specify: `-Dhttps.proxyHost=proxy.example.com -Dhttps.proxyPort=443`

Basic or Digest Proxy Authentication Method:

To enable basic or digest proxy authentication, set **NTLM_PROXY_AUTH** field above to **N** and include the following parameters in **CUSTOM_JAVA_OPTIONS**:

`-DproxyHost=proxy.example.com -DproxyPort=8080 -DproxyUser=username -DproxyPassword=encryptedpassword`

NTLM Proxy Authentication Method:

To enable NTML proxy authentication, set **NTLM_PROXY_AUTH** to **Y** and include the following in the parameters in **CUSTOM_JAVA_OPTIONS**:

```
-DproxyHost=proxy.example.com -DproxyPort=8080 -DproxyUser=username -DproxyPassword=encryptedpassword -DproxyDomain=domain
```

20. In **CUSTOM_INTERFACE_CLASS_NAME**, specify the fully qualified class name of the class in `custom.jar` that implements the `EpmAgentInterface`.

    For example, specify: `com.mycompany.agent.implementation.MyImplementation`.

    This parameter is optional.

21. In **CUSTOM_CLASS_PATH**, specify the full path and name of the jar (for example, `C:\AgentDeployment\agenthome\myJarFolder\custom.jar`

    This parameter is optional.

    If you use a Microsoft Java Database Connectivity (JDBC) driver connect to the SQL Server in the EPM Integration Agent, see Configuring the Microsoft JDBC Driver for SQL Server in the EPM Integration Agent.

22. In **NTLM_PROXY_AUTH**, specify **Y** to use an NTLM proxy authentication method.

    If you use NTLM proxy authentication, you must additionally specify proxy parameters in **CUSTOM_JAVA_OPTIONS**.

    Specify **N** to use basic or digest proxy authentication, or if you are not using proxy authentication.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> Set JYTHON_HOME or GROOVY_HOME if you want to use Jython or Groovy for scripting.
>
> Set CUSTOM_JAVA_OPTIONS, CUSTOM_INTERFACE_CLASS if you want to use custom Java for your extensions.

23. In **AMW_IDLE_TIME**, specify the idle time value for agents running in asynchronous mode during the Automated Maintenance Window (AMW) downtime or an unscheduled maintenance.

    The AMW_IDLE_TIME parameter value is set in minutes and the default setting is 15 minutes. Customers can define a longer idle time but should not specify an idle time value less than 15 minutes. The AMW downtime is a contiguous time interval during which automated maintenance tasks are run for a business process.

24. In **Client_ID**, enter the client ID generated when the Identity Domain Administrator configures the mobile application for OAuth. It is visible on the Configuration tab of the application, under **General Information**. For more information, see Enabling the OAUTH Option in the EPM Integration Agent.

    The Client ID is generated when the Identity Domain Administrator configures the mobile application for OAuth. It is visible on the Configuration tab of the application, under **General Information**.

    Oracle supports one CLIENT_ID for each EPM Integration Agent.

25. In **ENABLE_AUTO_COMMIT**, specify the auto-commit setting.

    The **ENABLE_AUTO_COMMIT** parameters determines whether every database operation is a transaction that is committed when performed. This setting addresses errors that may occur during agent processing in some versions of the JDBC Driver (for example, IBM DB2), which perform the auto-commit by default.

    To enable auto-commit, specify **Y**.

    To disable auto-commit, specify **N**.

    When using a JDBC driver that does not support auto-commit behavior, specify **I** to ignore the auto-commit setting.

    The default setting is **N**

26. In **RETRY_INTERVAL**, specify the retry logic period after a specified number of seconds.

    The default is **10** (seconds).

27. In **FILE_UPLOAD_SIZE**, specify the size of the file upload in iteration by a selected megabyte (MB) value.

    By default the file upload size is **50** (MB).

28. Save any changes to the `agentparamas.ini` file.

    When you have defined the startup parameters, you can run the EPM Integration Agent in diagnostic mode, which provides a way for you to identify any issue with the EPM Integration Agent startup parameters or network connectivity when the agent can't be started. For more information, see Running the EPM Integration Agent in Diagnostic Mode.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Encrypting the Cloud EPM User Password

Describes how to configure the startup parameters used by the EPM Integration Agent.

You can encrypt the Oracle Fusion Cloud Enterprise Performance Management user password referenced in the INI file by using the `encryptpassword.bat`.

To encrypt the Cloud EPM user password:

1. At a command prompt, type:

   Agent Home\EPMAgent\bin\encryptpassword.bat

   > **Note**
   >
   > Linux users, type:
   >
   > *Agent Home*\EPMAgent\bin\encryptpassword.sh

   When the EPM Password Utility is launched, in **Enter cloud password**, type the password and then press **[Enter]**.

   The password encoder in the agent uses an obscuring algorithm based on character substitution. The enconder supports only the following characters:

   - A
   - B
   - C
   - D
   - E
   - F
   - G
   - H
   - I
   - J
   - K
   - L
   - M
   - N
   - O
   - P
   - Q
   - R
   - S
   - T
   - U

Copyright © 2017, 2026, Oracle and/or its affiliates.

- V
- X
- Y
- Z
- a
- b
- c
- d
- e
- f
- g
- h
- i
- j
- k
- l
- m
- n
- o
- p
- q
- r
- s
- t
- u
- v
- w
- x
- y
- z
- 0
- 1
- 2
- 3
- 4
- 5
- 6
- 7

Copyright © 2017, 2026, Oracle and/or its affiliates.

- 8
- 9
- ~
- !
- @
- #
- $
- ?
- *
- (
-)
- _
- +
- ,
- /
- <

2. Copy the encrypted password shown after **Encrypted Password is:**.

You can paste the encrypted password in the `CLOUD_PASSWORD` parameter in the INI used to start the EPM Integration Agent.

Using OAuth 2.0 Authorization Protocol with OCI (Gen2) Environments

Describes how to set up OAuth 2.0 authorization for the EPM Integration Agent.

The EPM Integration Agent can use the OAuth 2.0 authentication protocol to access OCI (Gen 2) Oracle Fusion Cloud Enterprise Performance Management environments. OAuth is an open-standard authorization protocol that enables a service to use another service without requiring user security information (username, password, etc.).

Process Description for Using OAuth 2.0 Authorization

At a high level the following steps show you how to set up OAuth 2.0 authorization for the EPM Integration Agent.

To set up the OAuth 2.0:

1. **Register the application as a public client**—An Identity Domain Administrator registers an application as a public client in Oracle Cloud Identity Services. The domain administrator provides this authorization by registering a client and providing the appropriate registration information to the clients' users. OAuth2 is enforced for the application; not across your subscription.

   For more information, see Setting Up Authentication with OAuth 2.

Copyright © 2017, 2026, Oracle and/or its affiliates.

2. **Configure the EPM Integration Agent `agentparams.ini`**—Enable OAuth authentication for the EPM Integration Agent, specify the client identification generated when the Identity Domain Administrator configures the mobile application for OAuth, and specify the Cloud URL and EPM_APP_DATA_HOME.

   For more information, see Enabling the OAUTH Option in the EPM Integration Agent.

3. **Regenerate Tokens**— Run the `createoauthtoken.bat` file (for Windows) or `createoauthtoken.sh` (for Linux) to have the system regenerate the access token and refresh token automatically.

   For more information, see Running the createoauthtoken.bat or createoauthtoken.sh File to Generate Tokens.

Setting Up Authentication with OAuth 2

Oracle Fusion Cloud Enterprise Performance Management environments on an Oracle Cloud Infrastructure (OCI) / Gen 2 architecture enable you to use an OAuth 2 access token to issue REST APIs on Cloud EPM to satisfy the requirement of avoiding the use of passwords in your environment.

Setting Up Authentication with OAuth 2

In order for the EPM Integration Agent to access OCI (Gen 2) Oracle Fusion Cloud EPM environments, a Cloud EPM Service Administrator has to request the Domain Administrator to set up an OAuth 2 client and provide the Identity Domain Cloud Service (IDCS) URL, application scope, and Client ID.

Register the Client

The first step is to update the service provider configuration to authorize requests from the client application. As a security measure, any client application that accesses Oracle Cloud resources must be authorized to do so. A domain administrator provides this authorization by registering a client and providing the appropriate registration information to the client's users.

Clients can be public (outside a data center) or confidential. Public clients are assigned a client_id. Confidential clients also have a client secret in addition to a client_id. Clients are also authorized to access particular scopes. The type of application you select determines the allowed grant types available to request access tokens.

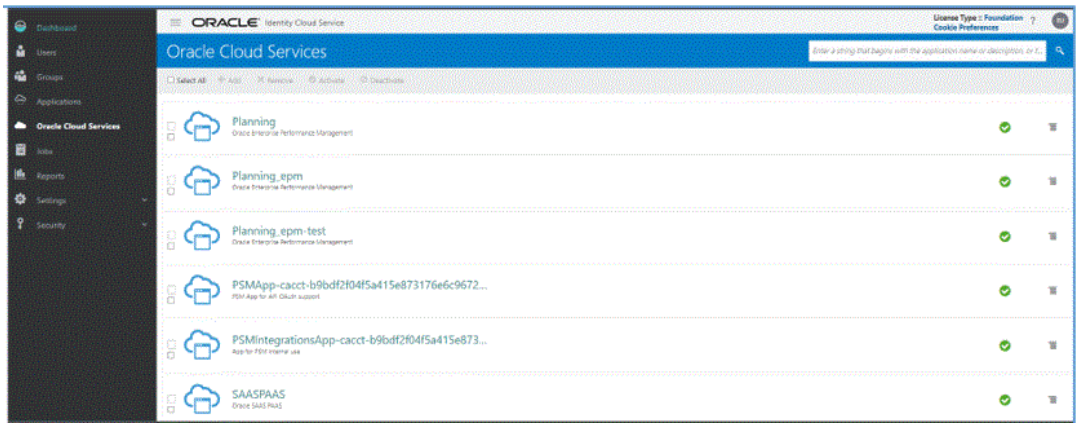
Client applications require an access token in order to access server resources. To obtain an access token, the client implements one of the IDCS supported access grant types, see Refresh Token Grant Type.

The Identity Cloud Service domain administrator follows the steps in this topic to create a public client in the Identity Cloud Service Administrator console for the requested client. The domain administrator then shares the Identity Cloud Service application URL, and client identification with the Cloud EPM Service Administrator.

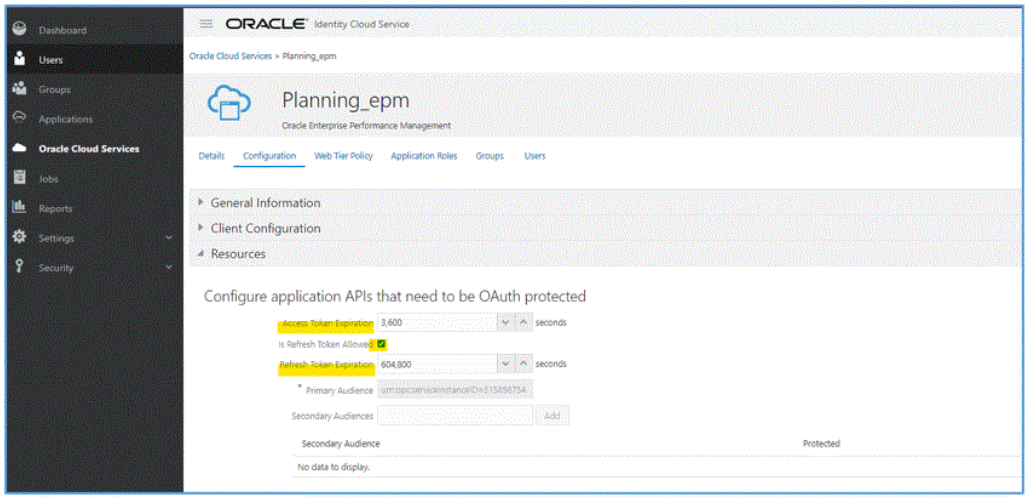
Identity Cloud Service Administrator tasks to register a client:

1. Log in to the Identity Cloud Service Administrator console as a Domain Administrator.

Copyright © 2017, 2026, Oracle and/or its affiliates.

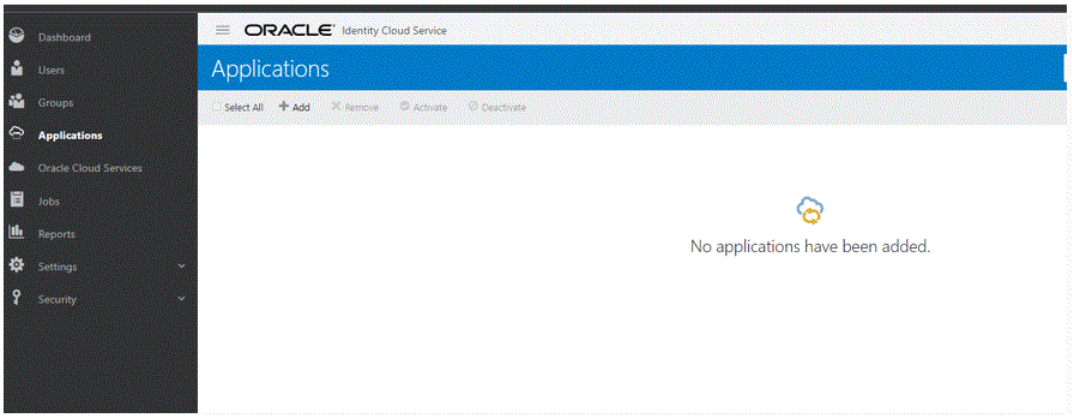


2. Configure the token properties for protected OAuth 2 APIs:
   a. From the **Dashboard** drawer, click **Oracle Cloud Services**, and then select the Cloud Service.
   b. On the **Configuration** tab, under **Resources**, select the **Is Refresh Token Allowed** check box.

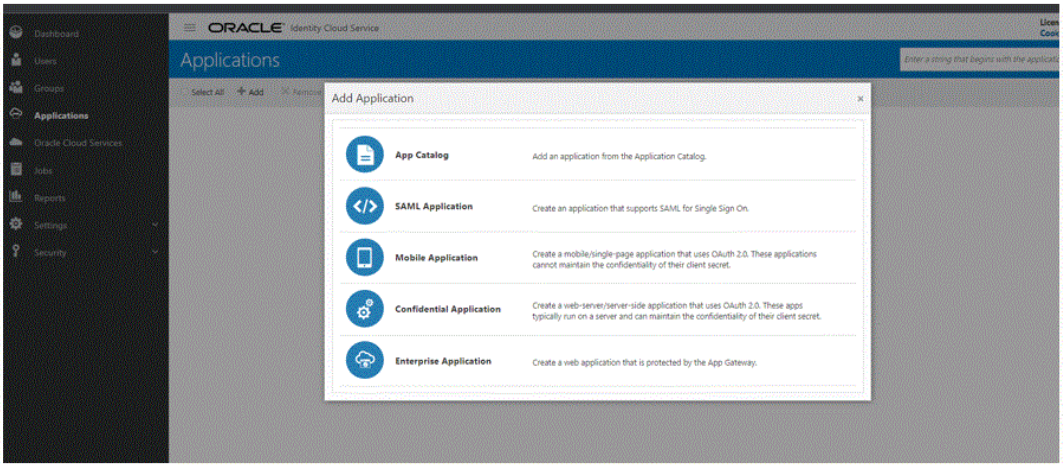


   c. **Optional**: Change the **Access Token Expiration** and **Refresh Token Expiration**. Oracle recommends 3600 (1 hour) as the value of Access Token Expiration and 604,800 (7 days) as the value of Refresh Token Expiration. The Refresh Token Expiration cannot be increased beyond 7 days.
   d. Click **Save**.
3. From the **Applications** drawer, click **Add** .

Copyright © 2017, 2026, Oracle and/or its affiliates.

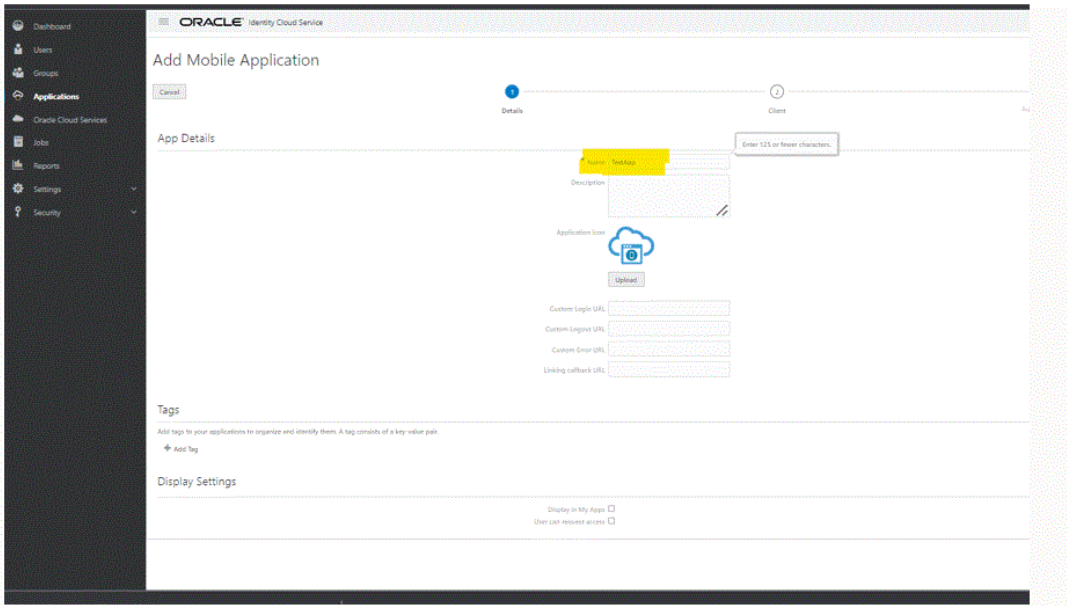


4. From the **Add Application** page, select **Mobile Application**.



For detailed information about adding a mobile application, see Add a Mobile Application.

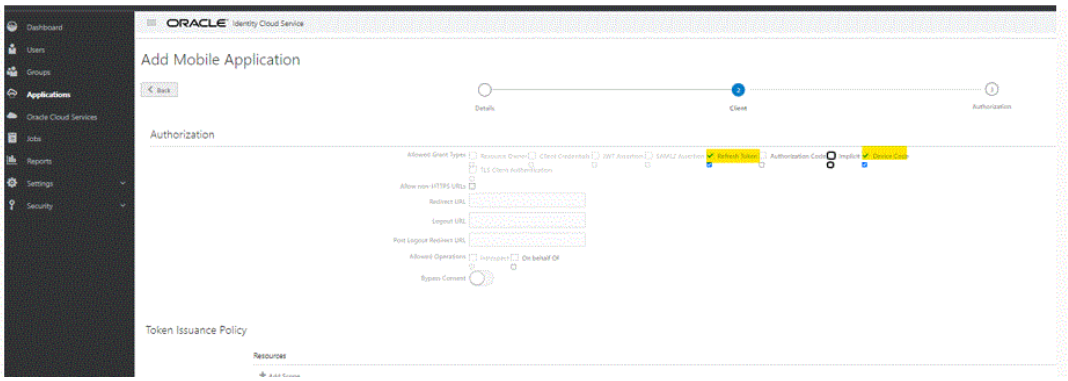
5. In the **App Details** section, then **Name**, enter the name for the REST client.



6. **Optional:** Add other details as needed.

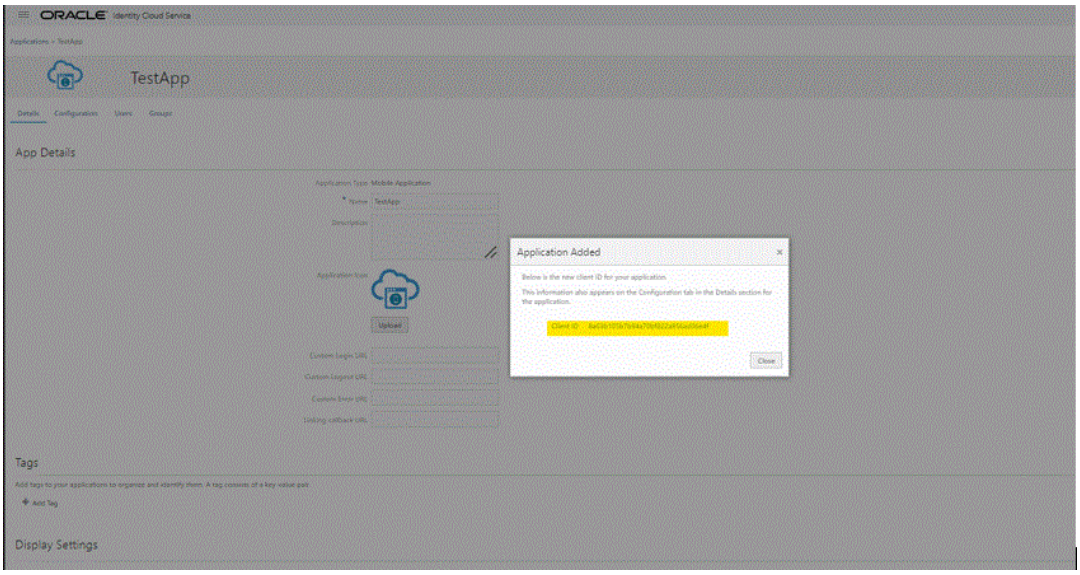
Copyright © 2017, 2026, Oracle and/or its affiliates.

7. Click **Next**.
8. In the **Authorization** section, under **Allowed Grant Types**, select the **Refresh Token** and **Device Code** check boxes.



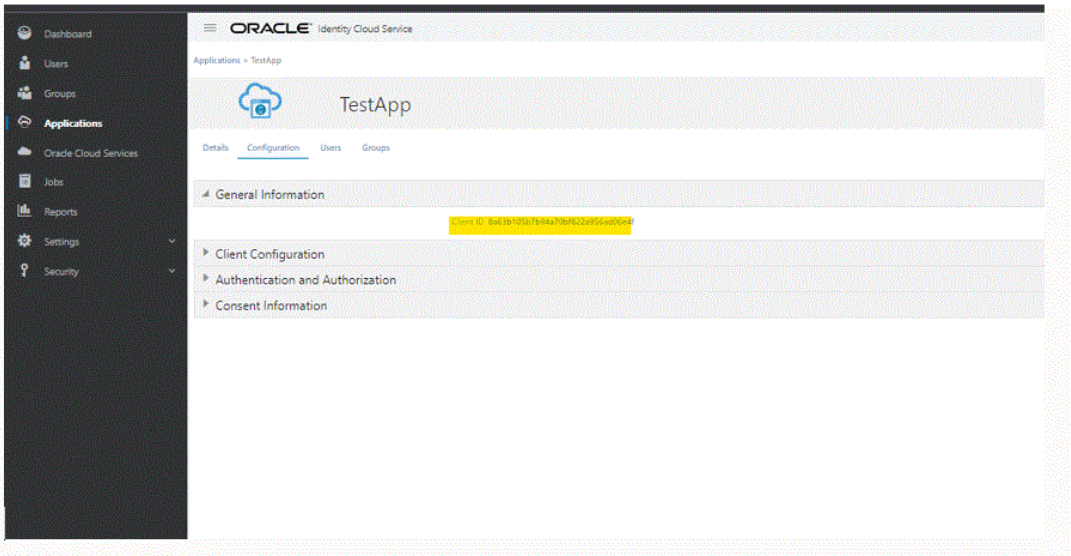
9. Click **Next** and then click **Finish**.

   The **Application Added** page shows the generated Client ID.

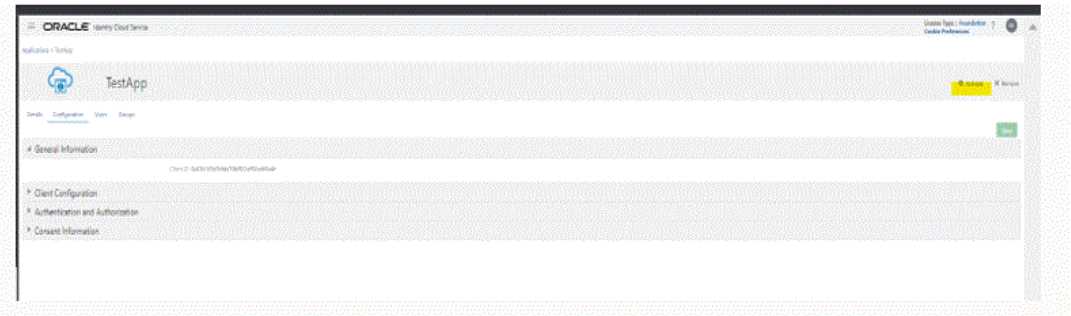


   This Client ID also appears on the **Configuration** tab under the **Details** section for the application.

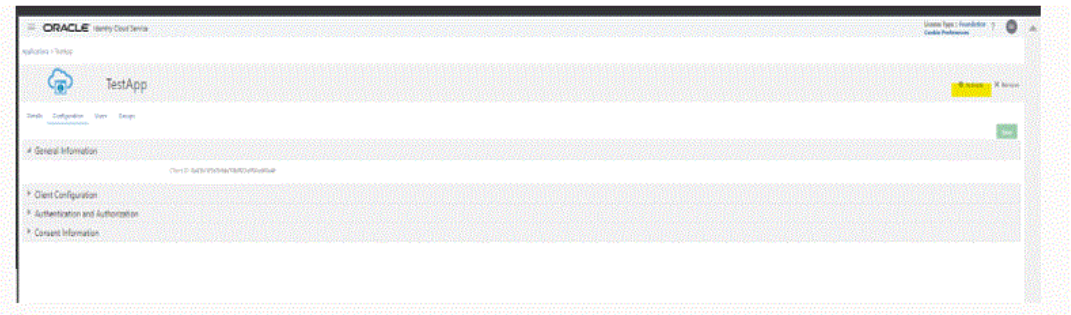
Copyright © 2017, 2026, Oracle and/or its affiliates.



10. Copy the ID to the Client_ID field in the `agentparams.ini` for the agent. See Enabling the OAUTH Option in the EPM Integration Agent.
11. On the **Application Added** page, click **Close**.
12. Click **Activate** to activate the client, and then click **OK**.



The message: "*application name* has been activated" is displayed.



Enabling the OAUTH Option in the EPM Integration Agent

Shows how to enable the Oauth option in the EPM Integration Agen:.

Service Administrators who want to use OAuth 2.0 for the EPM Integration Agent to access environments require these details in the `agentparamas.ini` file.

To enable the Oauth option in the EPM Integration Agent:

Copyright © 2017, 2026, Oracle and/or its affiliates.

1. Navigate to the `C:\EPMAgentData\config` directory.
2. Open the `agentparams.ini` file in a text editor.

3. In **Cloud_URL**, specify the Oracle Fusion Cloud Enterprise Performance Management URL used to authenticate and start the agent.

   This is the URL of the Cloud EPM instance associated with the agent, for example: *http://epmcloudserver*.

   Oracle supports one CLIENT_ID for each EPM Integration Agent
4. In **OAUTH_ENABLED** parameter, enter **Y**.

   If you enter **N** in this field, then password authentication is used instead of OAuth authentication.
5. In **Client_ID**, enter the client ID generated when the Identity Domain Administrator configures the mobile application for OAuth. It is visible on the Configuration tab of the

Copyright © 2017, 2026, Oracle and/or its affiliates.

application, under **General Information**. For more information, see Enabling the OAUTH Option in the EPM Integration Agent.

The Client ID is generated when the Identity Domain Administrator configures the mobile application for OAuth. It is visible on the Configuration tab of the application, under **General Information**.

6. Save the `agentparams.ini` file.

Running the createoauthtoken.bat or createoauthtoken.sh File to Generate Tokens

Shows how to run the `createoauthtoken.bat` or `createoauthtoken.sh` file to generate tokens.

After the Domain Administrator has registered the client and provided the IDCS URL and client ID, the Oracle Fusion Cloud Enterprise Performance Management Administrator can execute the `createoauthtoken.bat` (for Windows) or `createoauthtoken.sh` (for Linux) to regenerate the access token and refresh token automatically.

The script generates an access token and refresh token and stores the information securely in the `oauth.properties` file.

The system uses the access token to access the Cloud EPM information and when the access token expires, it uses the refresh token to generate a new access token.

> **Note**
>
> If you stop the agent and you do not start for more than refresh token expiry period, then you must rerun the `createauthtoken.bat` script to generate a new set of tokens. The expiry time of the refresh token is configurable in Identity Cloud Service by the Domain Administrator on a per Cloud EPM environment basis. The default expiry period is 604800 seconds, which is 7 days."

Running the `createoauthtoken.bat` (Windows) or `createoauthtoken.sh` (Linux)

To run `createoauthtoken.bat` or `createoauthtoken.sh`:

1. At a command prompt, execute the `createoauthtoken.bat` or `createoauthtoken.sh` as shown below and indicate the path to the *agent_parameters.ini* file.

   For example, specify:

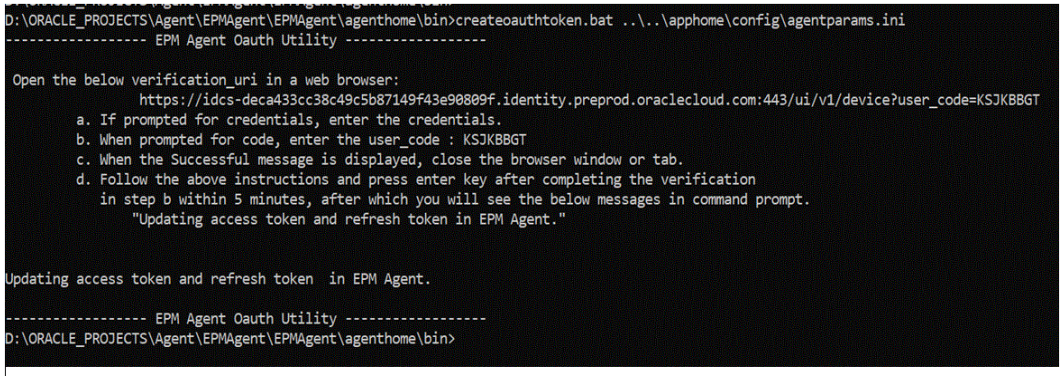
   ```
   createoauthtoken.bat path to the agent_parameters.ini file.
   ```

   or

   ```
   createoauthtoken.bat "C:\my apphome\config\agentparams.ini"
   ```

2. When you execute the `createauthtoken.bat C:\EPMAgent\apphome\config\agentparams.ini`, the following message is displayed in a command prompt window.

Copyright © 2017, 2026, Oracle and/or its affiliates.



Open the **verification_url** in a web browser, and then follow the steps as shown on the above page to verify the **user_code**. When the user_code is verified in the browser, then press **Enter** at the command prompt to complete the process.

a. If prompted for credentials, enter the credentials of the Cloud EPM user.

b. When prompted for the code, enter the user_code.

c. When the Successful message is displayed, close the browser window or tab.

d. Follow the above instructions and press enter key after completing the verification in step b within 5 minutes, after which you will see the below message in command prompt: "Updating access token and refresh token in EPM Agent."

Running the EPM Integration Agent as a Windows Service

Shows how to run the EPM Integration Agent in its own Windows session.

The EPM Integration Agent can be installed as a Windows service, enabling you to run the agent in its own Windows session. This service can be started automatically when the computer boots, can be paused and restarted, and does not show any user interface. On Linux computers, the EPM Integration Agent is started as a background process.

Note the following:

- When installed, you can start and stop the service as a Windows service from the Windows services console.
- Multiple agent services with a different service name, agent name and port can be created and run simultaneously.
- During execution the service logs are present in the agents `EPM_APP_DATA_HOME\logs` folder with the name `<serviceName>_<agent_name>_Service_<date>.log`. This log contains all the console output of the agent.
- To display help, use the option `EPMAgentService.exe -help` or double click `EPMAgentService.exe` in Windows Explorer.
- Always check the log file after starting the service.
- The agent name has to be specified in the `params.ini` file. It cannot be passed as a parameter in the Install command.

To install the agent as a Windows service, run the `EPMAgentService.exe` installer present in the `bin` folder by issuing the following command from a command prompt:

```
EPMAgentService.exe -install service_name path_to_agent_startup_parameter_file
```


Copyright © 2017, 2026, Oracle and/or its affiliates.

- *service_name*: Specify the unique name of the Window service for the agent. Letters and numbers are permitted. Spaces and special characters are not permitted.
- *path_to_agent_startup_parameter_file*: Specify the full path of the directory where the agentparams.ini file used during startup is located.

For example, you might specify: `C:\EPMAgentData\config\agentparams.ini`

To uninstall the agent as a Windows service, run the following command from a command prompt:

```
EPMAgentService.exe -uninstall service_name
```

Starting the EPM Integration Agent

Shows how to start the EPM Integration Agent from a command line.

You can start the EPM Integration Agent from a command line by specifying the name of EPM Integration Agent and the file name with the full path of the INI file to be used during the startup.

On startup, the agent makes an authorization call to the selected Oracle Fusion Cloud Enterprise Performance Management URL to authenticate the user's credentials. Upon a successful authentication, the EPM Integration Agent is registered in the Cloud EPM environment by storing the agent's host and port in the Cloud EPM environment's database.

> **Note**
>
> If you use a Windows service to start the agent, use the Windows service console. For more information, see Running the EPM Integration Agent as a Windows Service. If you are not using the Windows service to start the agent, then instead use the bat file steps described below.

To start the EPM Integration Agent:

1. Start the **Command Prompt** from the **Start Menu** (in Windows 10 and Windows 7) or the **Start Screen** (in Windows 8.1).

   To run the command prompt as an Administrator, right-click or tap and hold on its shortcut, then open the sub-menu and click or tap **Run as administrator**.

2. Type the following command with two parameters:

   `epmagent.bat Agent Name ini Parameter file name`

   - *Agent Name*: Specify the unique name of the agent. Letters and numbers are permitted. Spaces and special characters are not permitted.

     You can ignore *Agent Name* parameter if it is specified in the agentparams.ini file.

   - *ini Parameter file name*: File name with full path of the agentparams.ini file to be used during startup.

   For example, if the agent name has been specified in the agentparams.ini, type:

   ```
   epmagent.bat C:\EPMAgentData\config\agentparams.ini
   ```

   If the agent name has **not** been specified in the agentparams.ini, type:

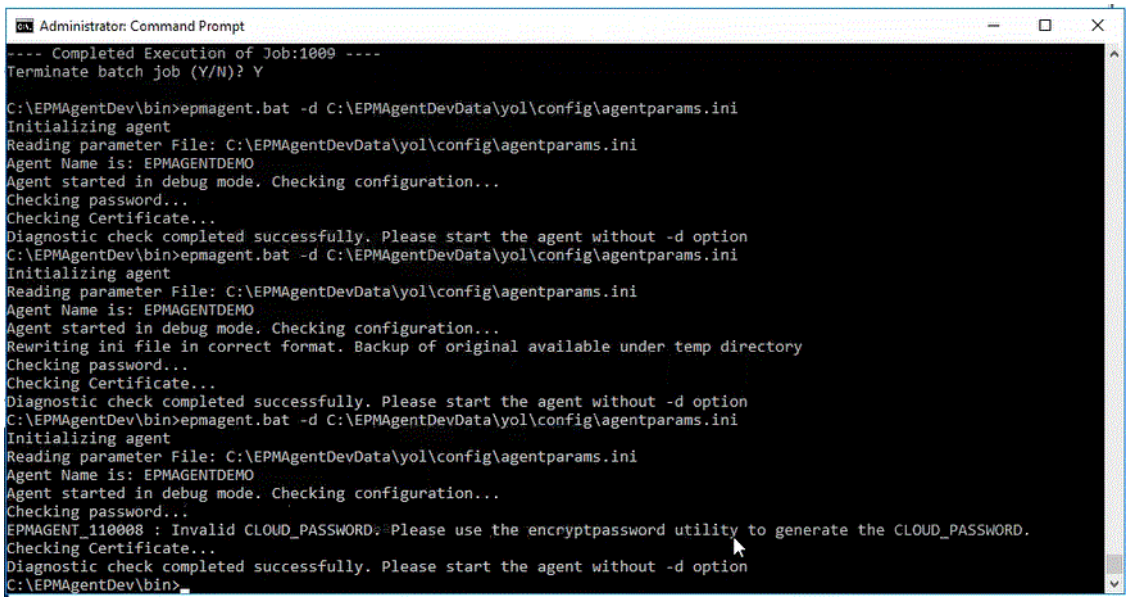
   ```
   epmagent.bat myagentname C:\EPMAgentData\config\agentparams.ini
   ```


Copyright © 2017, 2026, Oracle and/or its affiliates.

Running the EPM Integration Agent in Diagnostic Mode

Running the EPM Integration Agent in diagnostic mode provides a way for you to identify any issue with the EPM Integration Agent startup parameters or network connectivity when the agent can't be started. Diagnostic mode enables the system to perform a set of validations against the parameters in the agent startup file. If an error or invalid parameter is detected, the system shows the error associated with the parameter.

For example, when the Cloud password is unencrypted, the message: "Invalid CLOUD_PASSWORD. Please use the encryptpassword utility to generate the CLOUD_PASSWORD" as shown below:



> **Note**
>
> In diagnostic mode, the agent is not registered to the Oracle Fusion Cloud Enterprise Performance Management, but the original startup file is backed up to a temporary directory.

Validations include checks on:

- Agent connectivity to the internet. Includes making a test call to the oracle.com from the agent.
- Agent connectivity to the Cloud EPM URL within the agent.
- Proxy type (basic, digest, or NTML) when proxy is enabled in the customers environment
- Double quotes are used in Linux/Mac OS startup parameter file entries. Double quotes enable the shell to interpret dollar sign ($), backtick(`), backslash(\) and exclamation mark(!)
- Password is encrypted in the agent parameter file.
- Any whitespaces before or after parameters. If whitespaces are encountered, the system trims them and updates the parameter file.

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Any comments specified in the file. If a Comment line is encountered, it is removed after the diagnostic mode is performed.

To run the EPM Integration Agent in diagnostic mode:

1. Start the **Run command dialog** prompt from the **Start Menu** (in Windows 10 and Windows 7) or the **Start Screen** (in Windows 8.1).
2. At the **Run command prompt**, start the **EPM Integration Agent**.

   For example, type: **C:\EPMagent\bin\epmagent.bat**.
3. At the **Run command prompt**, run the agent in diagnostic mode by typing: `-d <path to startup parameters>`

   For example, type: **-d C:\EPMAgentData\yol\config\agentparams.ini**.

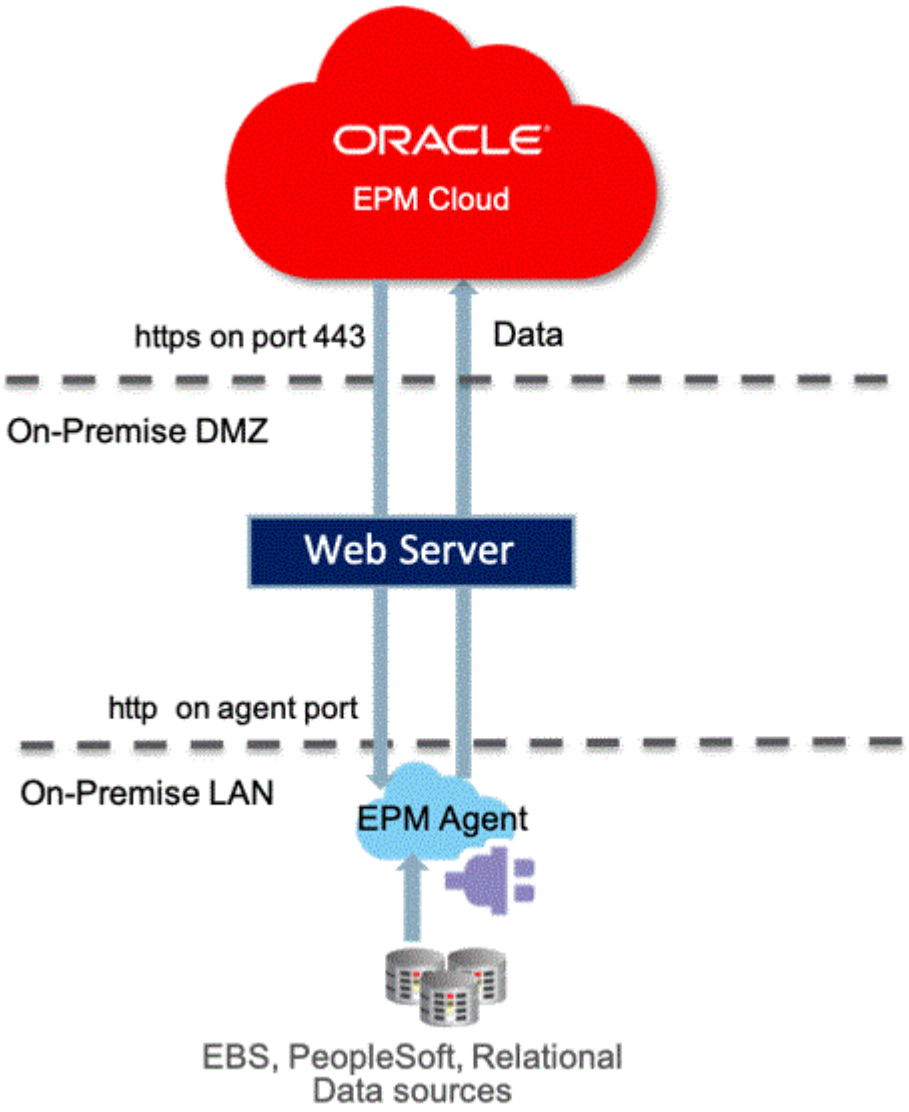
Configuring Synchronous Mode

Shows how to configure a web server as a gateway to enable inbound communication from the Oracle Fusion Cloud Enterprise Performance Management.

In synchronous mode you configure a web server as a gateway to enable inbound communication from the Cloud EPM.

The typical deployment model is shown below:

Copyright © 2017, 2026, Oracle and/or its affiliates.



> **Note**
>
> For more information about the synchronous mode, see Understanding EPM Integration Agents and Understanding Agent Clusters.

To configure a web server as a gateway for synchronous mode:

1. Configure an HTTPS web server that uses a trusted certificate from CA.
2. Define the Reverse Proxy in your Web Server configuration to route the request for `context/epmagent` to the Agent URL.

   For example, if you are using an Apache server, add the following entry to the httpd.conf file:

```
VirtualHost *:443
      ProxyPreserveHost On
      ProxyPass /epmagent http://Agent Server:9090/epmagent
```


Copyright © 2017, 2026, Oracle and/or its affiliates.

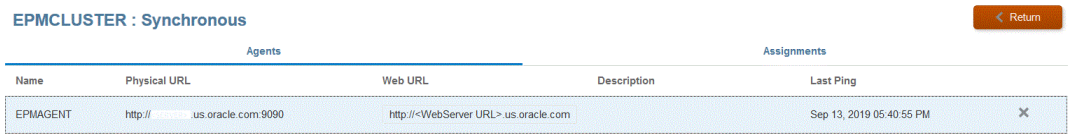
```
    ProxyPassReverse /epmagent http://Agent Server:9090/epmagent
/VirtualHost
```

> **Note**
>
> Use 443 as the SSL port and other ports. Other ports are blocked for security reason.

3. From the Data Integration home page, click **Actions**, and then select **Agent**.
4. From the **Agent Cluster** screen, select the name of the agent.
5. From the **Agent** tab, and then in **Web URL**, define the Web URL as:

   `https://WebServer URL`

   The following image shows the Web URL field on the Agents tab.



6. If you have multiple agents, define the proxy settings as follows:

   ```
   agent1/epmagent  http://Agent Server:9090/epmagent
   agent2/epmagent  http://Agent Server2:9090/epmagent
   ```

   On the Agent tab, define the Web URL as follows:

   ```
   https://WebServer URL/agent1
   https://WebServer URL/agent2
   ```

> **Note**
>
> Oracle recommends that you work with your network and web server administration to configure the web server and proxy configurations.

Cloud EPM to On-premises Connectivity Using the EPM Integration Agent

This section explains how to use the EPM Integration Agent to extend connectivity between the Oracle Fusion Cloud Enterprise Performance Management applications and your on-premises data sources.

Quick Reference:

- Extracting Data from On-premises Data Sources
- Using Prepackaged On-premises Adapters
- Drill through to Source using the EPM Integration Agent

Copyright © 2017, 2026, Oracle and/or its affiliates.

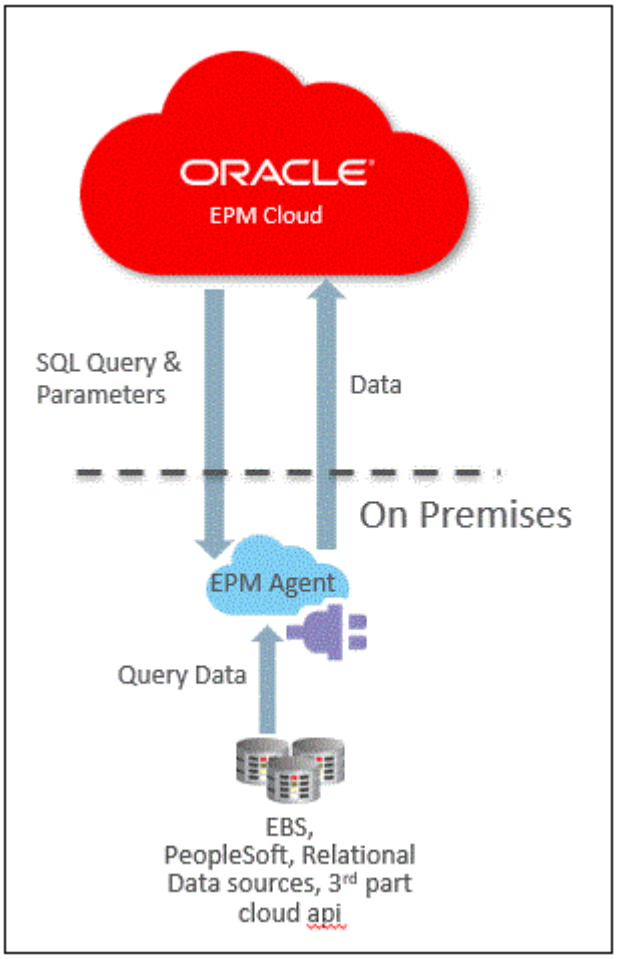
- Understanding EPM Integration Agents
- Understanding Agent Clusters
- Writing Back with the EPM Integration Agent
- EPM Integration Agent Scripting

> **Note**
>
> For information on installing and configuring the EPM Integration Agent, see Installing and Configuring the EPM Integration Agent.

Using the EPM Integration Agent

The EPM Integration Agent is a fully unified solution for extracting and transforming data and metadata from your on-premises or any system such as a third-party cloud, and delivering it to the Oracle Fusion Cloud Enterprise Performance Management.

You can connect to and load data from on-premises data sources using custom SQL queries or prepackaged queries to import data from EBS and the PeopleSoft General Ledger.



Copyright © 2017, 2026, Oracle and/or its affiliates.

Understanding EPM Integration Agents

Explains the synchronous and asynchronous modes when using the EPM Integration Agent.

The EPM Integration Agent can be deployed in two modes: synchronous and asynchronous.

In synchronous mode, when an integration is executed, the Oracle Fusion Cloud Enterprise Performance Management initiates a direct call from the cloud to the appropriate EPM Integration Agent based on the cluster assignment. The EPM Integration Agent executes the data extraction process and transfers data to the cloud. When the data is transferred to the Cloud EPM, the data is imported, mapped, and then loaded to the target application. In this mode, because the Cloud EPM communicates to the EPM Integration Agent directly, configure your on-premises environment to accept inbound communications from the Cloud EPM.

In asynchronous mode, when an integration is executed, the Cloud EPM queues the integration jobs to execute in the Cloud EPM. The EPM Integration Agent constantly polls the job queue at specified time intervals. If it finds a job, it executes the data extraction step and transfers data to the Cloud EPM. When the data is transferred to the Cloud EPM, the data is imported, mapped and loaded to the target application. In this mode, all communication between the Cloud EPM and the EPM Integration Agent is always initiated from the EPM Integration Agent and no inbound communication configuration is required. However, there may be a delay in the execution based on the polling frequency.

EPM Integration Agent Process Description

The steps are a high-level description on how to import a subset of data or metadata from your on-premises data sources and then load it directly to the Oracle Fusion Cloud Enterprise Performance Management using the EPM Integration Agent.

1. In Data Integration, write a **SQL query** to extract the data from the source system.

   For more information, see Creating an SQL Query.

2. Using SQL, generate a sample data file with a header record.

   The column name in the header record must exactly match the column name or column alias in the SQL. The names are case-sensitive.

3. In **Applications**, create an "On Premise Database" data source application and then select the sample data file.

4. Map the source columns to identify how source dimensionality translates to target dimensionality.

5. Assign integrations to different **clusters**.

   A cluster is a group of independent EPM Integration Agents. A cluster distributes incoming job traffic across a group of integrations.

6. Run the integration.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Extracting Data from On-premises Data Sources

Explains how to create an on-premises database adapter so that you can extract data from your on-premises data sources and then load the data directly to the Oracle Fusion Cloud Enterprise Performance Management.

You can extract data from your on-premises data sources and then load the data directly to the Cloud EPM using the EPM Integration Agent. The EPM Integration Agent executes a query against an on-premises relational database and then loads the data to the Cloud EPM.

The EPM Integration agent is defined as a data source for an integration, and when executed provides data that is subsequently mapped and loaded to the selected target application.

The EPM Integration Agent may also be extended by way of scripting to access other data sources including third-party cloud applications, non-relational sources, or any system that provides access by way of scripting, a report, or an API.

In order to use this feature and register the application created for the extraction, you must create a file that contains only sample data with one header record. The file does not contain the application name or SQL query, etc. The name of the file must be the name of the application. This type of data source is an application of the type "On Premise Database" and uses an adapter-based framework.

To create an on-premises database adapter:

1. In Data Integration, create the query definition and save the SQL query.

   To do this, see Creating an SQL Query.

   If you already have a SQL query registered in Data Integration, skip this step.

2. Create a file that contains one header row from the on-premises database.

   The header row must exactly match the dimension name in the target dimension. You can use an alias for a column name in the SQL query to the dimension name.

	A	B	C	D	E
1	Entity	Account	Product	Amount	
2	580	DPO	P_200	1233.98	
3	580	7300	P_200	45100.56	
4	580	7850	P_200	4000.33	
5	580	Capital Eq	P_200	-11900.7	
6	511	NI	P_200	17744.81	
7	580	NI	P_200	100	
8	580	NI	P_200	700	

3. Save the file as a CSV format file with a header row.

   Upload the file using the file browser when registering the target application.

4. **Create the on-premises database data source**:

Copyright © 2017, 2026, Oracle and/or its affiliates.

a. From the **Data Integration** home page, and then **Actions**, select **Applications**.

b. On the **Applications** page, click .

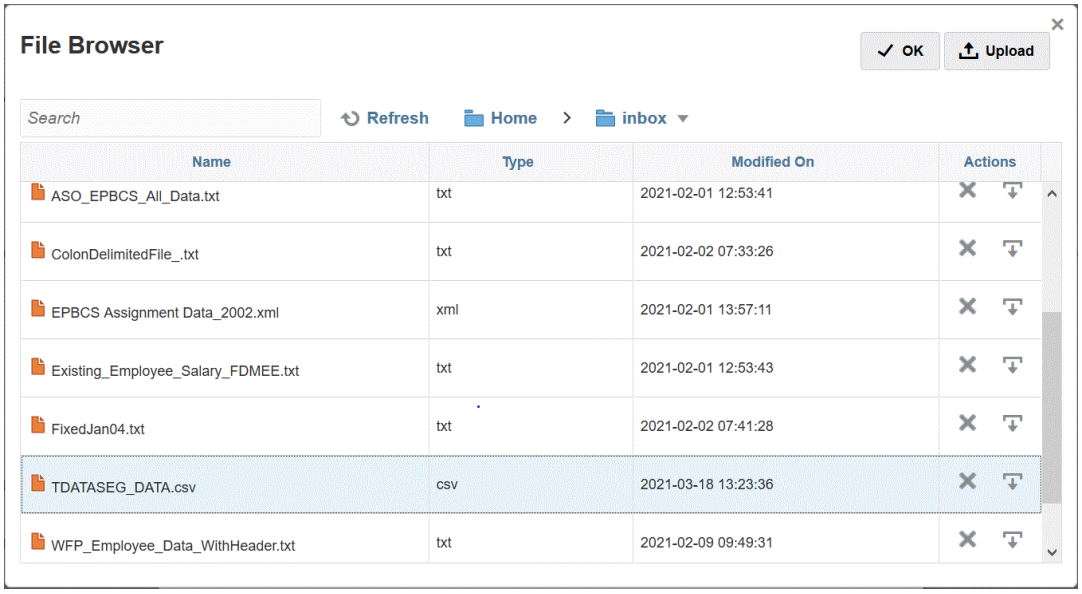
c. From **Create Application**, then **Category**, select **Data Source**.

d. From **Type**, select **On Premise Database**.

e. From **File**, select the file that you created in step 1.

   Click to browse for the file on the File Browser page.

5. When prompted, select the file that you created in step 1.

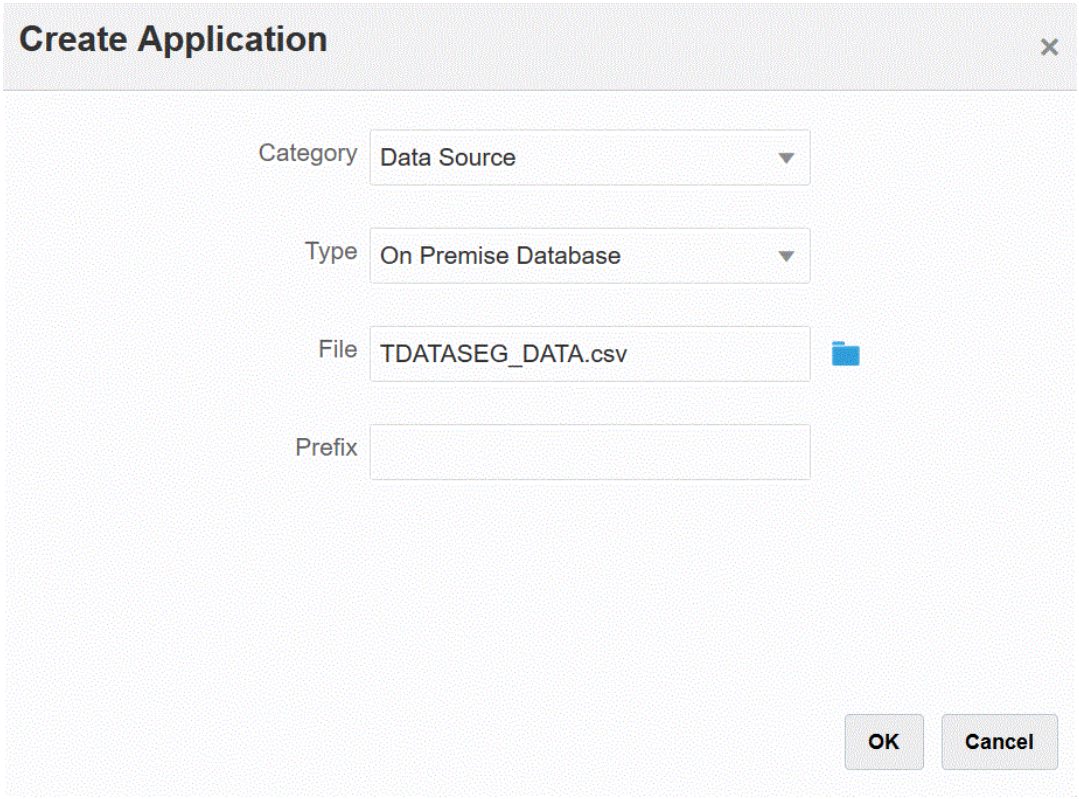


6. In **Prefix**, specify a prefix to make the application name unique.

   The prefix is concatenated with the application name to form a unique application name. For example, if you want to name an application with the same name as an existing one,

Copyright © 2017, 2026, Oracle and/or its affiliates.

you can assign your initials as the prefix.



7. Click **OK** and then click **Save**.
8. On the **Application**, click ... next to on-premises data source application created in steps 3-7 and then select **Application Details**.
9. On the **Application Details** page, click the **Options** tab.
10. In **Data Extract Query**, specify the name of the SQL query to run against the file.
11. In **Delimiter**, select the type of delimiter used in the file.

    Available delimiter symbols include the following:

    - Comma (,)
    - Exclamation (!)
    - Semicolon (;)
    - Colon (:)
    - Vertical bar (|)
12. In **Credential Store**, specify the type of credential store used by the EPM Integration Agent.

    Available types of credential stores include the following:

    - Cloud
    - File

    For the **Cloud** credential store type, store the *username/password/connect* string in the application.

Copyright © 2017, 2026, Oracle and/or its affiliates.

For the **File** credential store type, create a file in the on-premises environment to store the JDBC URL, user name, and password. The file name must be named *appname*`.cred` and stored in the `config` directory.

The file must contain the following lines:

```
driver=oracle.jdbc.driver.OracleDriver
jdbcurl=jdbc:oracle:thin:@host:port/service
username=apps
password=w+Sz+WjKpL8[
```

> **Note**
>
> The password used for both credential store types must be encrypted.
> When the type is a "Cloud" store, type the password in the user interface in the usual way. The system encrypts and stores the password.
>
> When the type is a "File" store, you must encrypt the password using the `encryptpassword` utility and store the password in the file. For more information about encrypting your password, see Encrypting the Cloud EPM User Password.

13. In **JDBC Driver**, select the type of JDBC driver to use when connecting to the Database Management System (DBMS).

    Available types of JDBC drivers include the following:

    - Microsoft SQL Server
    - Oracle

14. In **JDBC URL**, specify the JDBC driver URL connection string.

    The JDBC Driver URL connection string enables you to connect to a relational database using Java.

    For an Oracle Thin JDBC driver, the JDBC driver URL includes the following:

    `jdbc:oracle:thin:@`*`host`*`:`*`port`*`:`*`sid`*

    `jdbc:oracle:thin:@`*`host`*`:`*`port/service`*

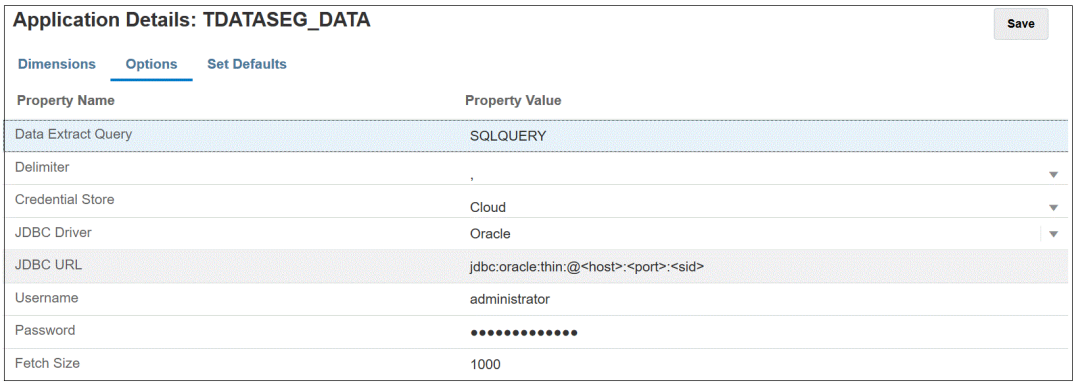
    For a MS SQL Server, the JDBC driver URL includes:

    `jdbc:sqlserver://server:port;DatabaseName=dbname`

15. In **Username**, specify the on-premises database user name.
16. In **Password**, specify the on-premises database password.
17. In **Fetch Size**, specify the number of rows fetched (that the driver processes) with each database round trip for a query.
18. Click **Save**.

    The following image shows application filters:

Copyright © 2017, 2026, Oracle and/or its affiliates.



Editing Application Options

You can specify additional application to return only records that meet selected conditions when using on-premises database sources. You can specify a single filter condition or multiple conditions, and specify the exact values to return.

To apply additional application options for an on-premises data source:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Application** page, click ... next to on-premise data source application.
3. Select the on-premises data source and click **Edit Options**.
4. On the **Edit Options** page, click **Add**.

   Blank entry fields display.
5. In **Name**, enter the name of the filter.

   For example, enter **Location** or **Period**.

   The name you enter is a bind variable or a placeholder for actual values in the SQL statement. In the SQL query, bind variables must be enclosed in ~~ characters. For example, to use "Period" as a bind variable, specify: `~PERIOD~`. The name must exactly match the name specified in the SQL query.
6. In **Display Prompt**, specify the name of the display prompt for the filter on the **Edit Integration** page in Data Integration.
7. In **Display Order**, specify the display order of the filter on the **Edit Integration** page.

   If this field is blank, the custom filter cannot be displayed, and the default value is used as the filter value.

   For example, enter **99** to show the filter in the 99th position sequence or position in a list of filters. Display orders are listed from lowest to highest.
8. In **Property Level**, select the type of process at which the filter is displayed.

   Available options:

   - Application
   - Application and Integration
   - Integration
9. In **Validation Type**, to provide a drop-down of values listed using the lookup type, in **Validation Type**, select one of the following:

Copyright © 2017, 2026, Oracle and/or its affiliates.

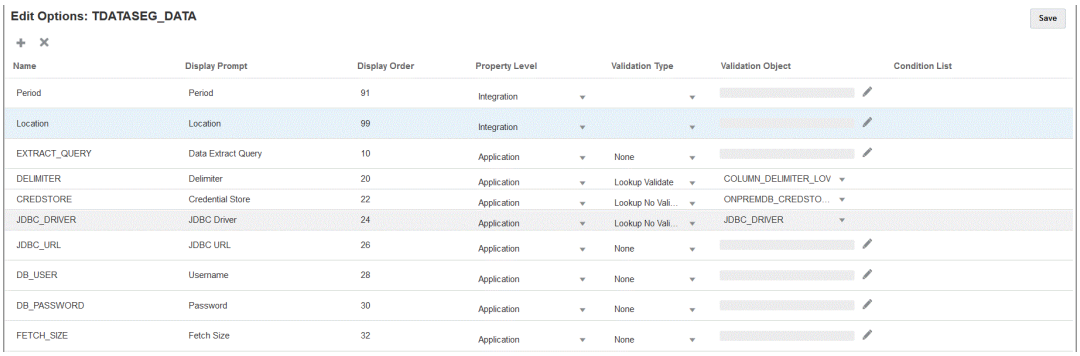
- None
- Number
- Lookup Validate
- Lookup No Validate
- Yes/No
- Date
- Query
- Choice list— You can enter a list of values in the Validation Object field. Each value is entered on a new line. The list of values (LOV) on the Option page shows the values as a list.

10. In **Validation Object**, enter a list of lookup objects to be validated by type.

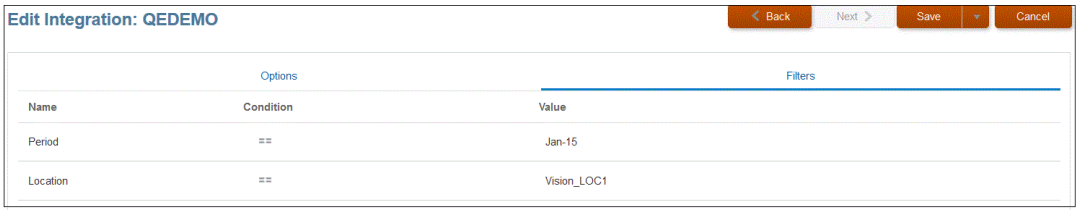
    To enter a list of values if you chose the **Choice list** validation type above, click , then enter the list of values on the Validation Object page, and then click **OK**.

11. Click **OK**.

    The following graphic shows how you might set up Period and Location application filters Data Integration:



    The following page shows how Period and Location filters are rendered on the **Edit Integration** page in Data Integration.



Creating an SQL Query

The Query option enables you to save the SQL query definition when creating an SQL data source. This type of data source is associated with an on-premises database adapter, which enables you to extract data from an on-premises database and load it to the Oracle Fusion Cloud Enterprise Performance Management using the EPM Integration Agent.

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Query**.

2. On the **Query** page, click **Add** ( ).

Copyright © 2017, 2026, Oracle and/or its affiliates.

3. From the **Create Query** page, and then in **Query Name**, specify the SQL query name.

   The query name is also used on the Application Filter page in Data Integration to identify the data extract query when registering the SQL data source in the target application.

4. In **Query String**, specify the SQL statement used to extract the data based on the header row in the file.

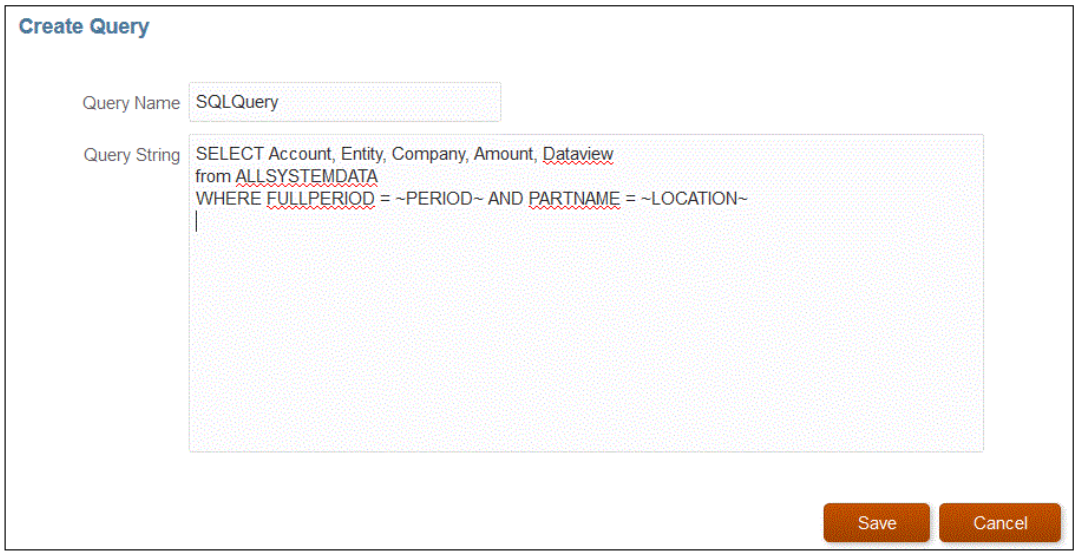
   You can use an alias for a column name in the SQL query to the dimension name.

   > **Note**
   >
   > Do not use the following keywords: DELETE, UPDATE, DROP, TRUNCATE , and ALTER in the Query definition.

   When specifying the string for a header row value and the string is mixed case, enclose the string in double quotation marks. When a string with mixed case characters is not enclosed within double quotation marks, it is resolved in uppercase.

   For example, if your header row uses the alias names Acct,Prod,Ent,Amt,Loc,Dat, then you might enter values as follows:

   ```
   SELECT ENTITYX AS "Ent" ,ACCOUNTX AS "Acct",UD3X AS "Prod",AMOUNTX AS "Amt",
   L.PARTNAME "Loc", P.PERIODDESC AS "Dat"FROM TDATASEG T, TPOVPARTITION L,
   TPOVPERIOD P WHERE T.PARTITIONKEY = L.PARTITIONKEY AND T.PERIODKEY =
   P.PERIODKEY AND L.PARTNAME = ~LOCATION~ AND P.PERIODDESC = ~PERIOD~
   ```

5. Click **Save**.

Using Conditions in the SQL Query

Following the WHERE clause, you can specify a search condition for the rows returned by the SELECT statement. The search condition returns a value when a condition evaluates to true about a specific row.

Valid conditions include:

- Equal
- Between

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Like
- Not In
- In

Note the following:

- The parameters returned by the conditions selected in the SQL Query are specified on the Target Application Filters page in Data Management.
- The = (equal) symbol in a complex query must precede the $ parameters.
- Target application parameters can have values separated by a comma such as: IN,BETWEEN,NOT IN.
- The system assigns the data type of the bind parameter. Number and Date are processed As Integer and Date. All other values are treated as Strings.
- Conditionalized filters are available on the Filters tab of the Run Integration page in Data Integration.

The following table describes the conditions that you can use in the SQL Query.

Condition	Description

EQUAL	Check whether two expressions are equal or not. If expressions are equal, then the condition is true and matched records are returned.
	When the following SQL statement is run for the equal condition and return records where the customer id equals Smith:
	`SELECT * FROM Customers`
	`WHERE CustomerID=1`

BETWEEN	Checks values between a certain range and returns matched values.
	The BETWEEN condition is inclusive. Beginning and ending values are included.
	BETWEEN syntax:
	`SELECT column_name(s)`
	`FROM table_name`
	`WHERE column_name BETWEEN value1 AND value2;`

LIKE	Performs and returns pattern matching using wildcards in the WHERE clause of a SELECT statement.
	Two wildcards are used in conjunction with the LIKE operator:
	• %: The percent sign represents zero, one, or multiple characters
	• _ : The underscore represents a single character
	LIKE syntax:
	`SELECT column1, column2, ...`
	`FROM table_name`
	`WHERE column LIKE pattern`

Copyright © 2017, 2026, Oracle and/or its affiliates.

Condition	Description

IN	Equal to any value in a list of values.
	IN syntax:
	`SELECT column_name1, column_name2, etc`
	`FROM table_name`
	`WHERE column_name1 IN (value1, value2, etc);`

NOT IN	Check whether two expressions equal or not. If expressions are not equal, then condition is true and returns not matched records.
	NOT IN syntax:
	`SELECT column_name1, column_name2, etc`
	`FROM table_name`
	`WHERE column_name1 NOT IN (value1, value2, etc);`

Displaying a Numeric Type Filter Using In/Between Conditions in an SQL Query

The following steps describe how to display a numeric type filter using the In and Between Conditions in the SQL Query executed by the EPM Integration Agent.

To display a numeric type filter using In/Between condition:

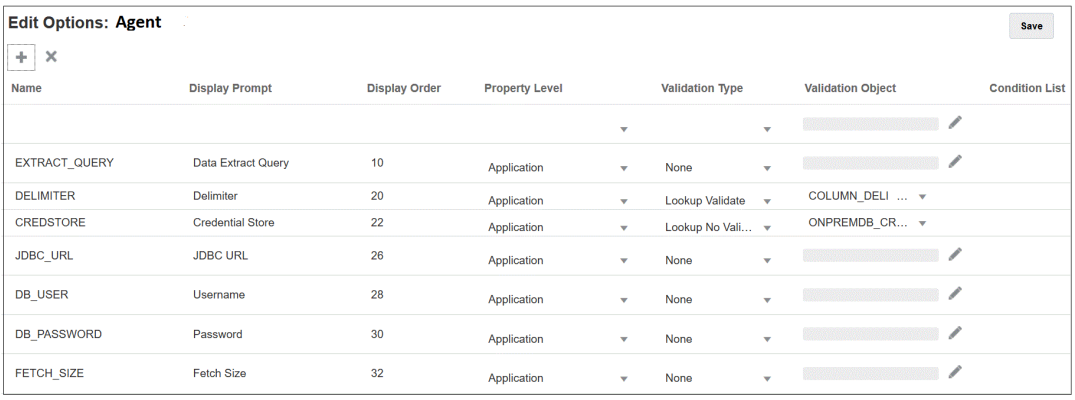
1. In Data Integration, create the SQL query associated with the SQL data source.

   The query name is used on the Application Details page to identify the data extract query when registering the SQL data source in the target application.

   For more information, see Creating an SQL Query.

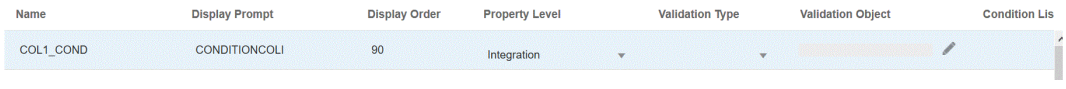
2. From the **Data Integration** home page, and then **Actions**, select **Applications**.

3. From the **Application** page, click ... to the right of the data target application, and then select **Edit Options**.



4. Click **(Add)**.

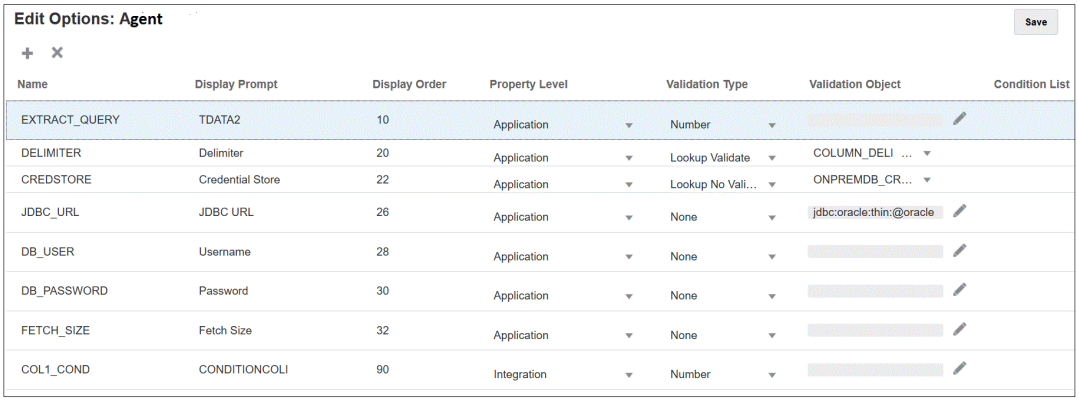
   In this example, the name of the filter is **COL1_COND** and the name of the Display Prompt is **CONDITIONCOLI**. This name is shown on the Options page as a target option in Data Integration.



Copyright © 2017, 2026, Oracle and/or its affiliates.

5. Click **Add** and add a filter for **EXTRACT_QUERY**, specify the value of the EXTRACT_QUERY parameter and click **OK**.

   The results of adding the filters are shown below:



   The value of the **EXTRACT_QUERY** is **TDATA2** and the **CONDITIONALCOLI** uses a "Between" condition with a range of values between **1 - 1999**.

6. In Data Integration, select the SQL query from step 1.

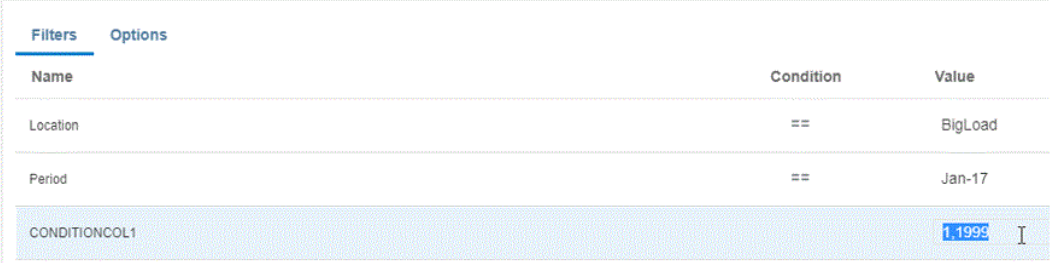
7. Specify the SQL statement used to extract the data based on the header row in the file and any filters.

   The following SQL includes the syntax for the filter, which allows users to select from a flexible account values in a range using "Between". Note that there is no equal sign. This is provided by the **Condition List** in the application filter.

8. Data Integration, update the integration definition to add the filter.

   Optionally, you can update the Filters tab with your selection on the Run integration page at runtime.

Copyright © 2017, 2026, Oracle and/or its affiliates.



Using Prepackaged On-premises Adapters

Explains how to load data using a prepackaged on-premises adapter.

Direct connectivity from the Oracle Fusion Cloud Enterprise Performance Management to on-premises data sources is available using prepackaged on-premises adapters. These adapters enable you to import data from:

- EBS GL Balances
- Peoplesoft GL Balances

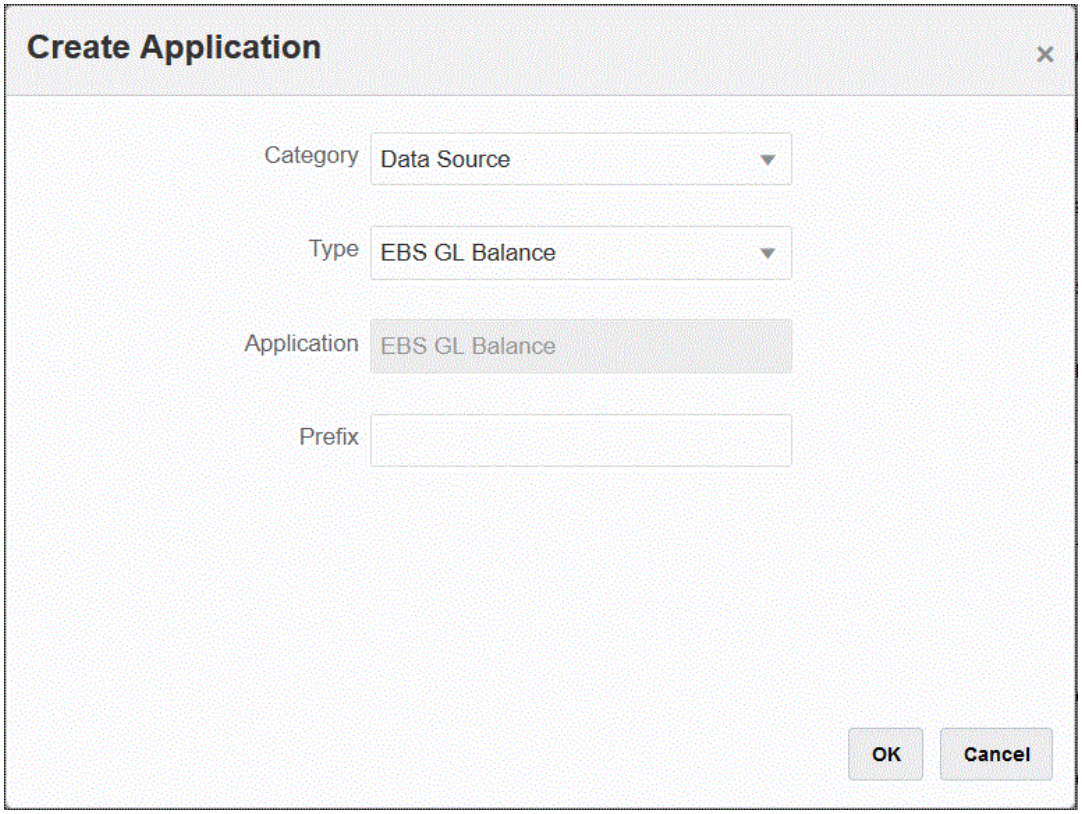
> **Note**
>
> For detailed information on the preseeded EBS GL Balances query, see E-Business Suite General Ledger Preseeded Query Reference.
> For detailed information on the prseeded Peoplesoft GL Balances query, see Peoplesoft General Ledger Preseeded Query Reference.

To load data using a prepackaged on-premises adapter:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
2. On the **Applications** page, click ➕.
3. From the **Create Application** page, and then **Category**, select **Data Source**.

   The application name for the prepackaged adapter populates the **Application Name**.
4. From **Type**, select either **EBS GL Balance** or **Peoplesoft GL Balance**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



5. In **Prefix**, specify a prefix to make the application name unique.

   The prefix is concatenated with the application name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.

6. Click **OK** and then click **Save**.

7. On the **Application** page, click ....

8. On the **Application Details** page, click the **Options** tab.

9. In **Delimiter**, select the type of delimiter used in the file.

   Available delimiter symbols:

   - Comma (,)
   - Exclamation (!)
   - Semicolon (;)
   - Colon (:)
   - Vertical bar (|)

10. In **Credential Store**, specify the type of credential store used by the EPM Integration Agent.

    Available types of credential stores:

    - Cloud
    - File

    For the **Cloud** credential store type, you store the *user name/password/connect* string in the application.

Copyright © 2017, 2026, Oracle and/or its affiliates.

For the **File** credential store type, create a file in the on-premises environment to store the JDBC URL, user name, and password. The file name must be named *appname*`.cred` and stored in the `config` directory.

The file must contain the following lines:

```
driver=oracle.jdbc.driver.OracleDriver
jdbcurl=jdbc:oracle:thin:@host:port/service
username=apps
password=w+Sz+WjKpL8[
```

> **Note**
>
> The password used for both credential store types, must be encrypted.
> When the type is a "Cloud" store, type the password in the user interface in the usual way. The system encrypts and stores the password.
>
> When the type is a "File" store, you must encrypt the password using the `encryptpassword` utility and store the password in the file. For more information about encrypting your password, see Encrypting the Cloud EPM User Password.

11. **Peoplesoft only**: In **JDBC Driver**, select the type of JDBC driver to use when connecting to the Database Management System (DBMS).

    Available types of JDBC drivers include the following:

    - Microsoft SQL Server
    - Oracle

    A JDBC driver is software that enables Java application to communicate with a database. The JDBC driver communicates the connection information to the database and sends the protocol used for transferring the query and result between the client and the database.

12. In **JDBC URL**, specify the JDBC driver URL connection string.

    The JDBC Driver URL connection string enables you to connect to a relational database using Java.

    For an Oracle Thin JDBC driver, the JDBC driver URL includes:

    `jdbc:oracle:thin:@`*`host`*`:`*`port`*`:`*`sid`*

    `jdbc:oracle:thin:@`*`host`*`:`*`port/service`*

    For SQL Server, the JDBC driver URL includes:

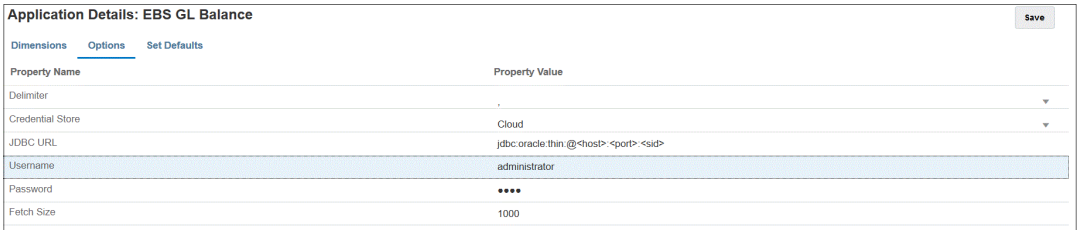
    `jdbc:sqlserver://server:port;DatabaseName=dbname`

13. In **Username**, specify the on-premises database user name.

14. In **Password**, specify the on-premises database password.

15. In **Fetch Size**, specify the number of rows fetched (that the driver processes) with each database round trip for a query.

Copyright © 2017, 2026, Oracle and/or its affiliates.



16. Click **Save**.

E-Business Suite General Ledger Preseeded Query Reference

Explains how to use a preseeded query to import General Ledger balances from Oracle E-Business Suite sources using the EPM Integration Agent.

You can use a preseeded query to import General Ledger balances from E-Business Suite sources and then create and register an application that can be loaded to Oracle Fusion Cloud Enterprise Performance Management target applications using the EPM Integration Agent.

> **Note**
>
> The ledger and period are required parameters for using the E-Business Suite preseeded query.

The following sample shows a prebuilt E-Business Suite query that comes with the EPM Integration agent. Note you can use this query as a starting point to build a custom query.

```
SELECT
 gcc.SEGMENT1           as "Segment1",
 gcc.SEGMENT2           as "Segment2",
 gcc.SEGMENT3           as "Segment3",
 gcc.SEGMENT4           as "Segment4",
 gcc.SEGMENT5           as "Segment5",
 gcc.SEGMENT6           as "Segment6",
 gcc.SEGMENT7           as "Segment7",
 gcc.SEGMENT8           as "Segment8",
 gcc.SEGMENT9           as "Segment9",
 gcc.SEGMENT10          as "Segment10",
 gcc.SEGMENT11          as "Segment11",
 gcc.SEGMENT12          as "Segment12",
 gcc.SEGMENT13          as "Segment13",
 gcc.SEGMENT14          as "Segment14",
 gcc.SEGMENT15          as "Segment15",
 gcc.SEGMENT16          as "Segment16",
 gcc.SEGMENT17          as "Segment17",
 gcc.SEGMENT18          as "Segment18",
 gcc.SEGMENT19          as "Segment19",
 gcc.SEGMENT20          as "Segment20",
 gcc.SEGMENT21          as "Segment21",
 gcc.SEGMENT22          as "Segment22",
 gcc.SEGMENT23          as "Segment23",
 gcc.SEGMENT24          as "Segment24",
 gcc.SEGMENT25          as "Segment25",
 gcc.SEGMENT26          as "Segment26",
```


Copyright © 2017, 2026, Oracle and/or its affiliates.

```
 gcc.SEGMENT27           as "Segment27",
 gcc.SEGMENT28           as "Segment28",
 gcc.SEGMENT29           as "Segment29",
 gcc.SEGMENT30           as "Segment30",
 gb.BEGIN_BALANCE_DR     as "Beg Balance DR",
 gb.BEGIN_BALANCE_CR     as "Beg Balance CR",
 gb.PERIOD_NET_DR        as "Period Net DR",
 gb.PERIOD_NET_CR        as "Period Net CR",
(gb.BEGIN_BALANCE_DR - gb.BEGIN_BALANCE_CR) + (gb.PERIOD_NET_DR-
gb.PERIOD_NET_CR) as "YTD Balance",
(gb.PERIOD_NET_DR-gb.PERIOD_NET_CR) as "Periodic Balance",
CASE
  WHEN ACCOUNT_TYPE IN (''A'',''L'',''O'') THEN ((gb.BEGIN_BALANCE_DR-
gb.BEGIN_BALANCE_CR) + (gb.PERIOD_NET_DR-gb.PERIOD_NET_CR))
  WHEN ACCOUNT_TYPE IN (''R'', ''E'') THEN (gb.PERIOD_NET_DR-gb.PERIOD_NET_CR)
  ELSE (gb.PERIOD_NET_DR-gb.PERIOD_NET_CR)
END as "Balance by Acct Type",
gb.BEGIN_BALANCE_DR_BEQ as "Func Eq Beg Bal DR",
gb.BEGIN_BALANCE_CR_BEQ as "Func Eq Beg Bal CR",
gb.PERIOD_NET_DR_BEQ    as "Func Eq Period Net DR",
gb.PERIOD_NET_CR_BEQ    as "Func Eq Period Net CR",
(gb.BEGIN_BALANCE_DR_BEQ - gb.BEGIN_BALANCE_CR_BEQ) + (gb.PERIOD_NET_DR_BEQ-
gb.PERIOD_NET_CR_BEQ) as "Func Eq YTD Balance ",
(gb.PERIOD_NET_DR_BEQ-gb.PERIOD_NET_CR_BEQ) as "Func Eq Periodic Balance",
CASE
  WHEN ACCOUNT_TYPE IN (''A'',''L'',''Q'') THEN ((gb.BEGIN_BALANCE_DR-
gb.BEGIN_BALANCE_CR) + (gb.PERIOD_NET_DR-gb.PERIOD_NET_CR))
  WHEN ACCOUNT_TYPE IN (''R'', ''E'') THEN (gb.PERIOD_NET_DR-gb.PERIOD_NET_CR)
  ELSE (gb.PERIOD_NET_DR-gb.PERIOD_NET_CR)
END as "Func Eq Balance by Acct Type",
gld.LEDGER_ID           as "Ledger ID",
gld.NAME                as "Ledger Name",
gb.PERIOD_YEAR          as "Period Year",
gb.PERIOD_NAME          as "Period Name",
gb.PERIOD_NUM           as "Perion Number",
gcc.ACCOUNT_TYPE        as "Acoount Type",
gb.CODE_COMBINATION_ID  as "Code Combination ID",
gb.CURRENCY_CODE        as "Currecy Code",
gb.ACTUAL_FLAG          as "Balance Type",
 gb.BUDGET_VERSION_ID    as "Budget Version ID",
gb.ENCUMBRANCE_TYPE_ID  as "Encumbrance Type ID",
gb.TRANSLATED_FLAG      as "Translated",
gb.PERIOD_TYPE          as "Period Type",
 gcc.ENABLED_FLAG        as "Enabled",
gcc.SUMMARY_FLAG        as "Summary Account"
FROM GL_BALANCES gb
,GL_CODE_COMBINATIONS gcc
,GL_LEDGERS gld
WHERE (1=1)
AND gcc.CODE_COMBINATION_ID = gb.CODE_COMBINATION_ID
AND gb.ACTUAL_FLAG = ''A''
AND gb.TEMPLATE_ID IS NULL
AND gld.LEDGER_ID  = gb.LEDGER_ID
AND gld.NAME       = ~LEDGER~
AND gb.PERIOD_NAME = ~PERIOD~
```


Copyright © 2017, 2026, Oracle and/or its affiliates.

Peoplesoft General Ledger Preseeded Query Reference

Explains how to use a preseeded query to import General Ledger balances from Peoplesoft sources sources using the EPM Integration Agent.

You can use a preseeded query to import General Ledger balances from Peoplesoft sources and then create and register an application that can be loaded to an Oracle Fusion Cloud Enterprise Performance Management target applications using the EPM Integration Agent.

The following shows the prebuilt Peoplesoft query that comes with the EPM Integration agent:

```
SELECT
    PL.BUSINESS_UNIT           as  "Business Unit",
    PL.LEDGER                  as  "Ledger",
    PL.ACCOUNT                 as  "Account",
    PL.ALTACCT                 as  "Alt Account",
    PL.DEPTID                  as  "Department",
    PL.OPERATING_UNIT          as  "Operating Unit",
    PL.PRODUCT                 as  "Product",
    PL.FUND_CODE               as  "Fund Code",
    PL.CLASS_FLD               as  "Class",
    PL.PROGRAM_CODE            as  "Program",
    PL.BUDGET_REF              as  "Budget Reference",
    PL.AFFILIATE               as  "Affiliate",
    PL.AFFILIATE_INTRA1        as  "Affiliate Intra1",
    PL.AFFILIATE_INTRA2        as  "Affiliate Intra2",
    PL.CHARTFIELD1             as  "Chartfield1",
    PL.CHARTFIELD2             as  "Chartfield2",
    PL.CHARTFIELD3             as  "Chartfield3",
    PL.PROJECT_ID              as  "Project",
    PL.BOOK_CODE               as  "Book Code",
    PL.GL_ADJUST_TYPE          as  "GL Adjust Type",
    PGA.STATISTICS_ACCOUNT     as  "Stat Account",
    PGA.ACCOUNT_TYPE           as  "Account Type",
    PGA.DESCR                  as  "Account Description",
    PL.CURRENCY_CD             as  "Currency",
    PL.STATISTICS_CODE         as  "Stat Code",
    PL.FISCAL_YEAR             as  "Fiscal Year",
    PL.ACCOUNTING_PERIOD       as  "Accounting Period",
    PL.POSTED_TOTAL_AMT        as  "Posted Total Amount",
    PL.POSTED_BASE_AMT         as  "Posted Base Amount",
    PL.POSTED_TRAN_AMT         as  "Posted Tran Amount",
    PL.BASE_CURRENCY           as  "Base Currency",
    PL.PROCESS_INSTANCE        as  "Process Instance"
FROM PS_LEDGER   PL,
      PS_GL_ACCOUNT_TBL   PGA
WHERE ( 1=1 )
  AND ( PL.BUSINESS_UNIT = ~BU~
  AND PL.LEDGER = ~LEDGER~
  AND PL.FISCAL_YEAR = ~YEAR~
  AND PL.ACCOUNTING_PERIOD = ~PERIOD~
  AND ( PGA.SETID = ( SELECT SETID
                      FROM   PS_SET_CNTRL_REC
                      WHERE  SETCNTRLVALUE    =  ~BU~
```


Copyright © 2017, 2026, Oracle and/or its affiliates.

```
                    AND    RECNAME          = ''GL_ACCOUNT_TBL'' )
AND PGA.EFFDT = ( SELECT MAX(B.EFFDT)
                  FROM PS_GL_ACCOUNT_TBL B

WHERE PGA.SETID = B.SETID AND PGA.ACCOUNT = B.ACCOUNT )
)
AND ( PL.ACCOUNT=PGA.ACCOUNT )
)
```

Importing Data from a File in the EPM Integration Agent Inbox

Explains how to import data from a file located in a local EPM Integration Agent `inbox` folder and then have the agent upload and process the file.

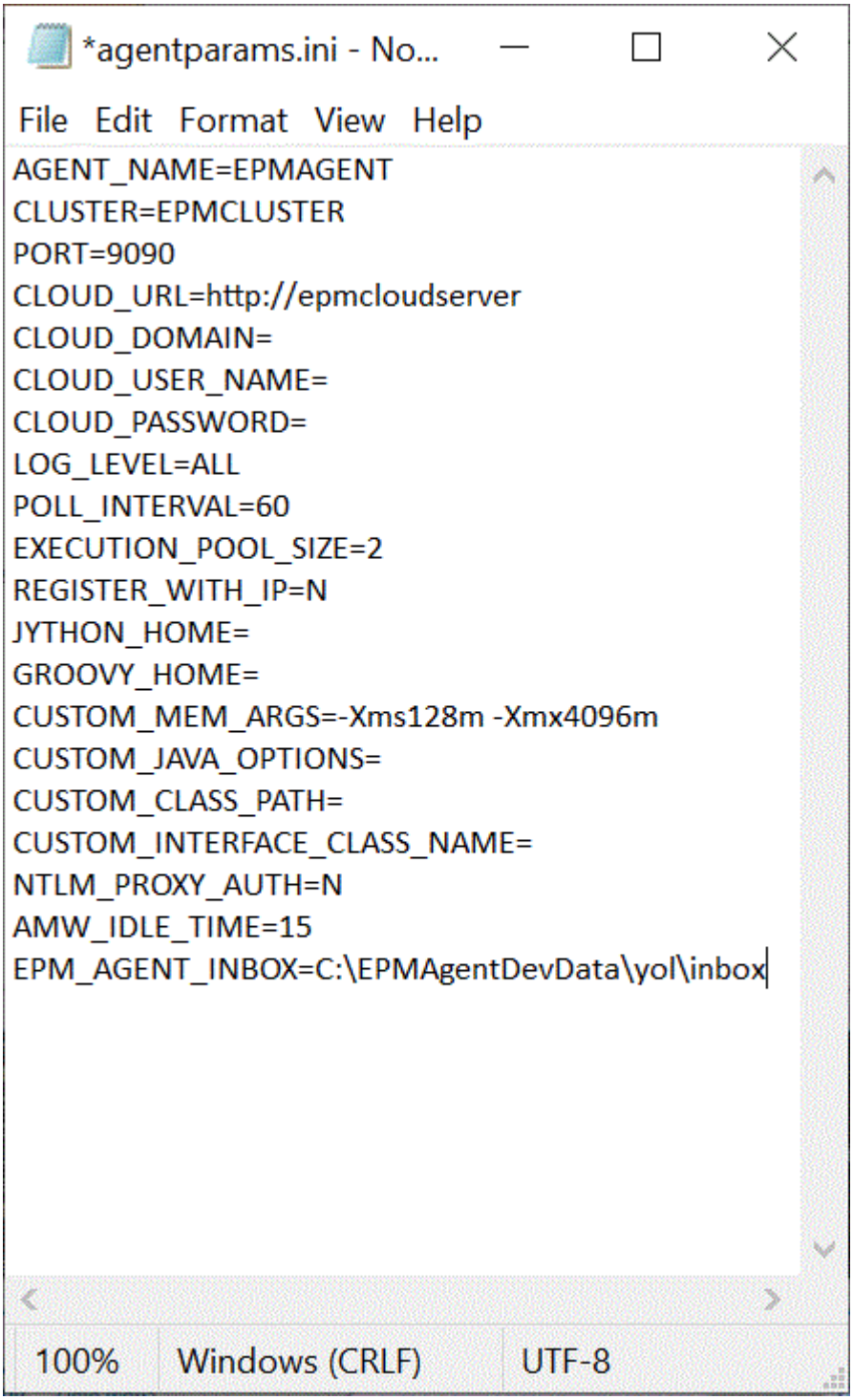
You can import data from a file located in a local EPM Integration Agent `inbox` folder and then have the agent upload and process the file.

> **Note**
>
> This feature enables you to use a file in an EPM Integration Agent inbox without having to use an EPM Automate command to import the file first.

To stage a file to an EPM Integration Agent inbox:

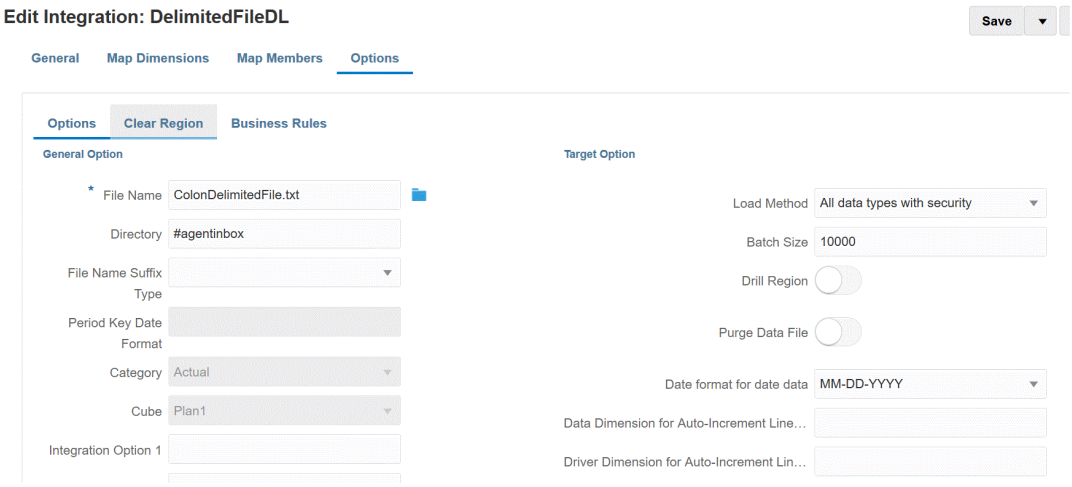
1. Create an inbox folder under the *<EPMAgentData>* application folder to which to copy a file.
2. In agent startup parameters file, specify the location of the inbox in the EPM_AGENT_INBOX parameter field.
   a. Go to the *<EPMAgentDevData\config>* directory.
   b. Double click the **agentparams.ini** file to open it using a text editor such as Notepad.
   c. In the **EPM_AGENT_INBOX** parameter, specify the full directory of the EPM Integration Agent inbox and save the file.

Copyright © 2017, 2026, Oracle and/or its affiliates.



3. On the Data Integration home page, click ... to the right of the file-based integration, and then select **Options**.
4. Select the **Options** tab.
5. In **Directory**, enter **#agentinbox** and click **Save**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



6. Stage the file to the inbox that you created in step 1.

Using an Agent File Adapter

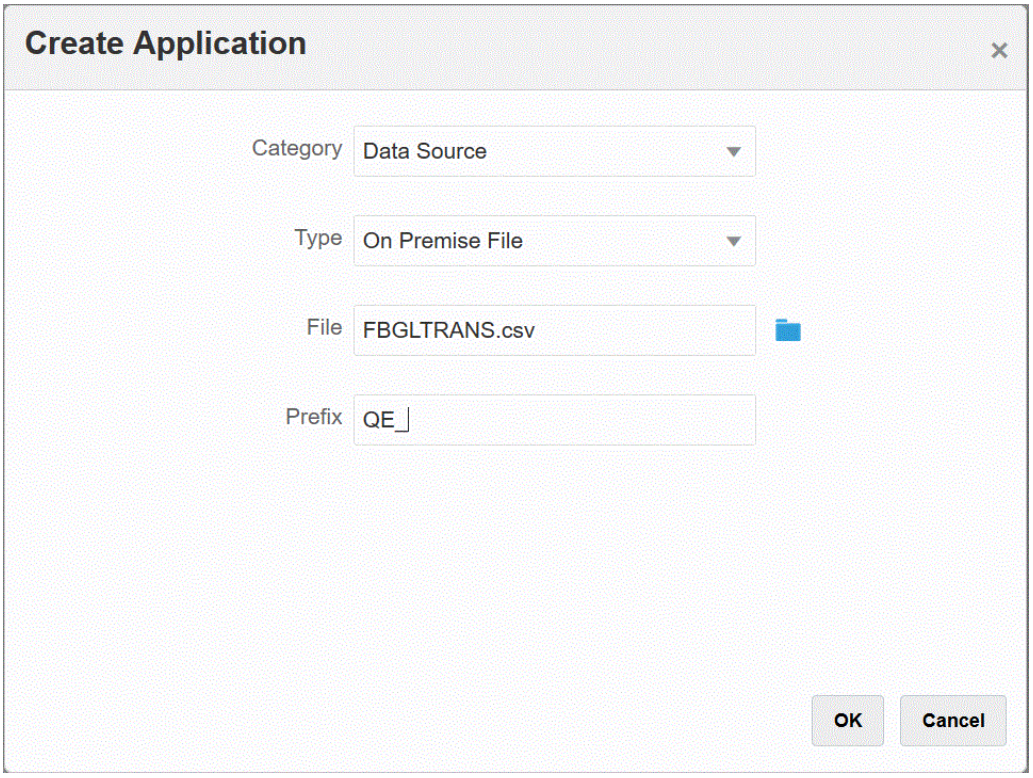
Explains how to load very large files from an on-premises file data source to the Oracle Fusion Cloud Enterprise Performance Management.

An Agent File adapter enables you to load very large files from an on-premises file data source to the Cloud EPM. It extends the Quick Mode method where the Cloud EPM database is bypassed for staging and processing, eliminating any performance bottlenecks and improving the performance and scalability of the load process. The agent file adapter does not include any connection or query parameters. The only required parameter is the name of the file. A delimited file with a header record is required for the load.

To use the Quick Mode method with an agent file adapter:

1. **Create the on-premises file adapter**.
   a. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**
   b. On the **Applications** page, click (Add icon).
   c. From **Category**, select **Data Source**.
   d. From **Type**, select **On Premise File**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



e. From **File**, specify the source data file to use as the template.

   Click to search for a file from the **File Browser** page.

   When selecting a file, note the following:

   - The source data file must be a delimited data file.
   - Data files used must contain a one-line header, which describes the delimited columns.
   - Both numeric and non-numeric data can be loaded.

f. **Optional**: In **Prefix**, specify a prefix to make the application name unique.

   The prefix is concatenated with the application name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.

g. Click **OK**.

2. **Create the integration job between the on-premises data source and Cloud EPM:**

   a. From the **Data Integration** home page, click (Add icon).

   b. On the **Create Integration** page, and then in **Name** and **Description**, enter a name and description for the integration job.

   c. In **Location**, enter a new location name, or pick an existing location to specify where to load data.

   d. Tap the **Quick Mode** slider on.

      When you associate an integration job with the Quick Mode method and then save the job, you cannot reverse the Quick Mode association. However; the integration job can be deleted.

Copyright © 2017, 2026, Oracle and/or its affiliates.

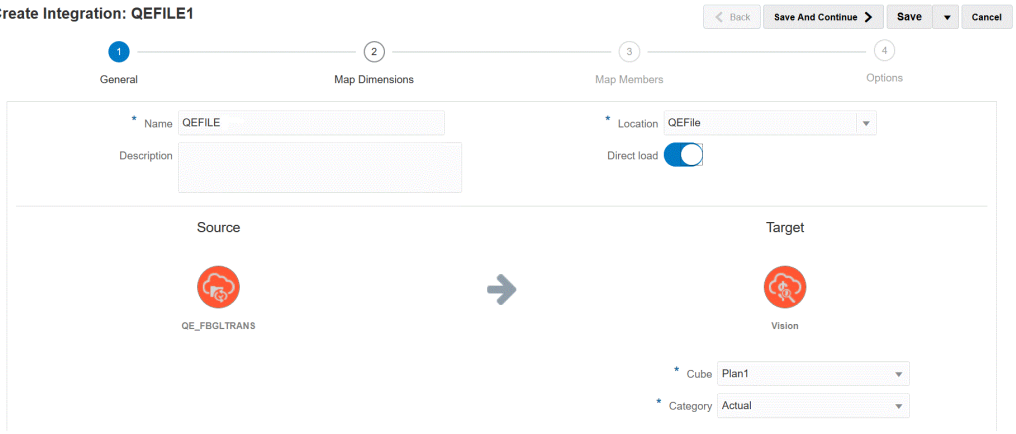
e. Click (Select a Source).

f. From the **Select a Source** page, click next to the on-premises file data source application.

g. Click (Select a Target).

h. From the **Select a Target** page, select the target Cloud EPM application.

i. Click **Save and Continue**.



3. Map the dimensions between the on-premises file data source and the Cloud EPM application:

   a. On the **Mapping Dimensions** page, from **Type**, select the type of data load method.

      Available options:

      - Delimited - Numeric data: supports numeric data types only.
      - Delimited - All data types: supports the following data types to Planning
        - numbers
        - text
        - Smartlists
        - Data

   b. In the mappings grid, map the source columns in the on-premises data source application to the dimensions in the Cloud EPM application by completing the following:

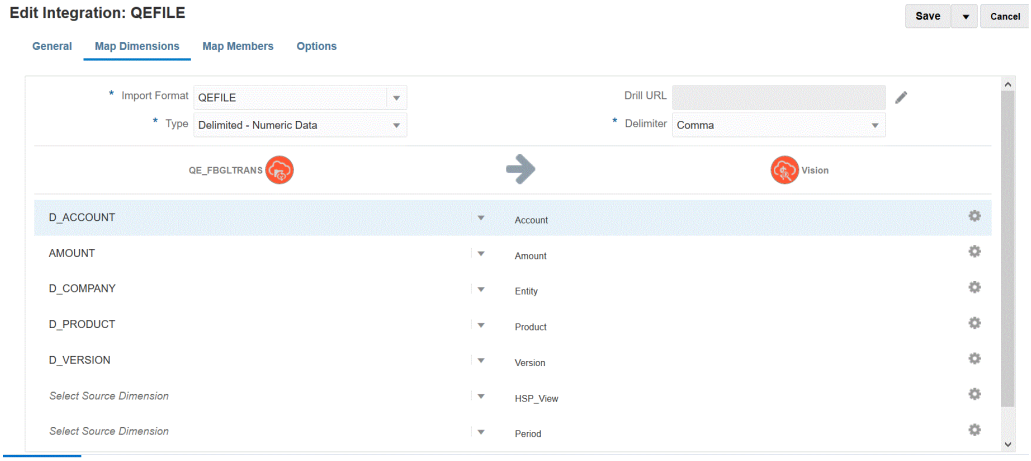
      i. In **Select Source Dimension**, select the name of the source dimension to map to the Cloud EPM dimension.

      ii. **Optional**: Add a target expression for each of the Cloud EPM dimensions.

      For information on using target expressions, see Using Target Expressions.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> You cannot use a SQL target expression type with the Quick Mode method.
> Source expression types are not available with the Quick Mode method.



c. Click **Save and Continue**.

4. **Run the Integration**:

   a. From the **Data Integration** home page, select the integration job associated with the Quick Mode method and then click ▶.

   b. From the **Run Integration** page, and then from **Mode** drop-down, select **Replace** as the export mode for the integration.

   c. From the **Period** drop-down, select the single period from the source file from which to load data.

      If you map a period dimension and specify a target expression for period on the Map Dimension page for the integration, then the Period drop-down is not available for selection because then the period is derived from the mapping.

   d. Click **Run**.

Performing a Quick Mode Load using the EPM Integration Agent

Explains how to use Quick Mode method to load data from your on-premises data sources to Oracle Fusion Cloud Enterprise Performance Management using the EPM Integration Agent.

Use the Quick Mode method to load data from your on-premises data sources to Cloud EPM using the EPM Integration Agent. When you integrate the on-premises data source with the Cloud EPM target using the EPM Integration Agent, the system executes a SQL query on the source data in the on-premises relational database, offloads processing, extracts and transforms the data at the source level, and then loads the data directly to a data export on-premises database file. In this way, the Cloud EPM database is bypassed for staging and processing, eliminating any performance bottlenecks and improving the performance and scalability of the load process.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> For information using Quick Mode method to load data "as is", see Quick Mode to Export Data.

> **Note**
>
> Quick mode data load is available for all business processes except for Account Reconciliation. Quick mode data export with level 0 is only available for a platform-based business process, and does not include Account Reconciliation or Profitability and Cost Management. Enterprise Profitability and Cost Management provides quick mode with level 0 data export.

Considerations:

Consider the following points associated with a Quick Mode method using the EPM Integration Agent:

1. When you create and save a Quick Mode load integration job, you cannot later change it to a standard integration job. However, you can delete the data load integration job.
2. When mapping dimensions, target expressions types *are* supported. Target expressions enable you to transform the source value read from the source to target dimension values to be loaded to the target application.

   Target expressions that can be used for the Period dimension include substring(), split(), map() toPeriod() and toYear().

   With the exception of the SQL target expression type, all remaining target expressions are supported.
3. When mapping dimensions, source expression types *are not* supported.
4. When loading data in Quick Mode using the agent, the Driver dimension expression is not supported.

   The Driver expression is currently only supported for Non-Quick mode and Quick Mode file-based integrations.
5. Mapping members is not supported.
6. If you select the Level 0 data extract method, the system creates a "DM BR Data Export " business rule to perform the data extract.
7. You can specify a single period during execution in which case all the data is loaded to the single period. The other option is to derive the period dimension based on the source system period name using `toPeriod` and `toYear` target expressions.
8. When running the Quick Mode load, only the Replace export mode is supported.

   No import modes are available.
9. A direct drill though to source is required when data is loaded using Quick Mode method. For more information, see Defining Direct Drill.
10. For EPM Automate, you must use the `runIntegration` command to execute Quick Mode load integration.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Quick Mode Process Description

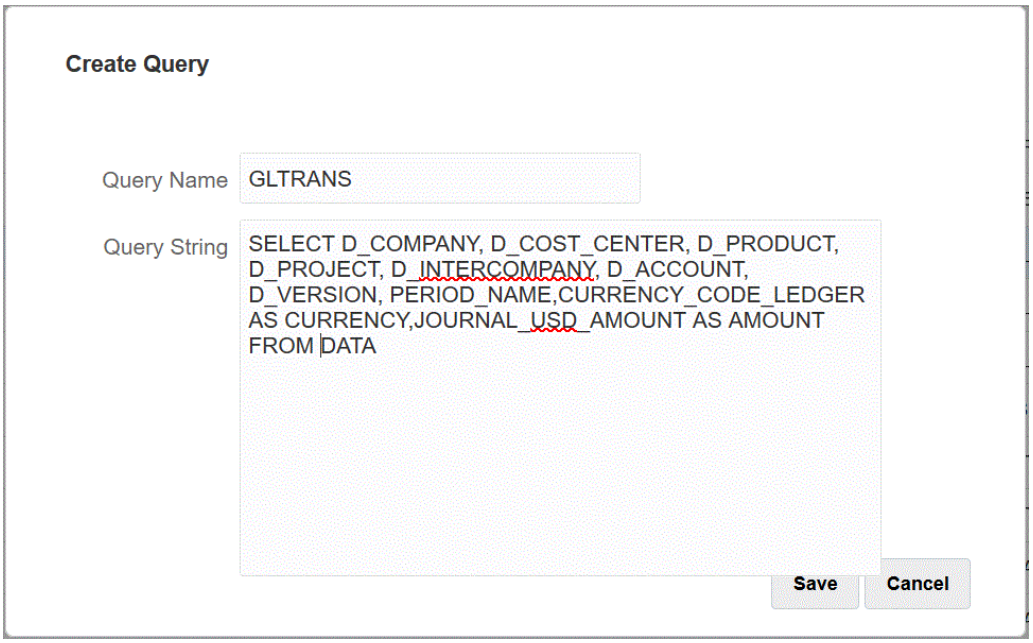
This is the process description for using Quick Mode method to load data from your on-premises data sources to Oracle Fusion Cloud Enterprise Performance Management using the EPM Integration Agent.

This section describes how to use the Quick Mode method to extract data from an on-premises data source and then load the data directly to an on-premises database file using the EPM Integration Agent. The EPM Integration Agent executes a query against an on-premises relational database and then loads the data to a data export file.

To use the quick mode method:

1. **Create the SQL Query:**
   a. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Query**.
   b. On the **Query** page, click ✚.
   c. From the **Create Query** page, and then in **Query Name**, specify the SQL query name.

      The query name is used on the Options tab in Application Details to identify the data extract query when registering the SQL data source for the integration.
   d. Create the query definition and save the SQL query.

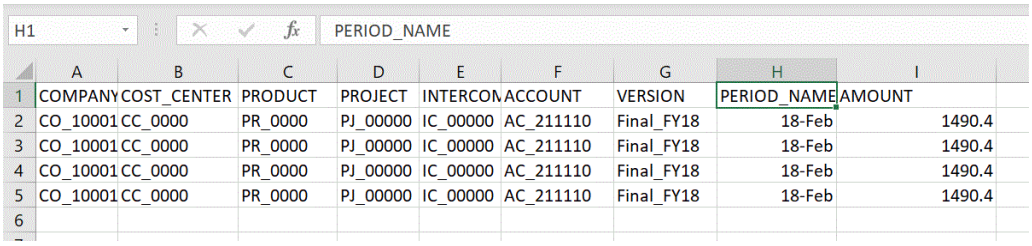


   For more information, see Creating an SQL Query .

2. **Create the on-premises database file**:
   a. Create a file that contains one header row from the on-premises database.

      The header row must exactly match the dimension name in the target dimension. You can use an alias for a column name in the SQL query to the dimension name.
   b. Save the file as a **CSV** format file with a header row.

Copyright © 2017, 2026, Oracle and/or its affiliates.



	A	B	C	D	E	F	G	H	I
1	COMPANY	COST_CENTER	PRODUCT	PROJECT	INTERCON	ACCOUNT	VERSION	PERIOD_NAME	AMOUNT
2	CO_10001	CC_0000	PR_0000	PJ_00000	IC_00000	AC_211110	Final_FY18	18-Feb	1490.4
3	CO_10001	CC_0000	PR_0000	PJ_00000	IC_00000	AC_211110	Final_FY18	18-Feb	1490.4
4	CO_10001	CC_0000	PR_0000	PJ_00000	IC_00000	AC_211110	Final_FY18	18-Feb	1490.4
5	CO_10001	CC_0000	PR_0000	PJ_00000	IC_00000	AC_211110	Final_FY18	18-Feb	1490.4
6									

c. Load the file using the file browser when registering the on-premises database application.

3. **Register the on-premises database file application**:

a. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.

b. On the **Applications** page, click .

c. From **Create Application**, then **Category**, select **Data Export**.

d. From **Type**, select **Data Export to On Premises Database**.

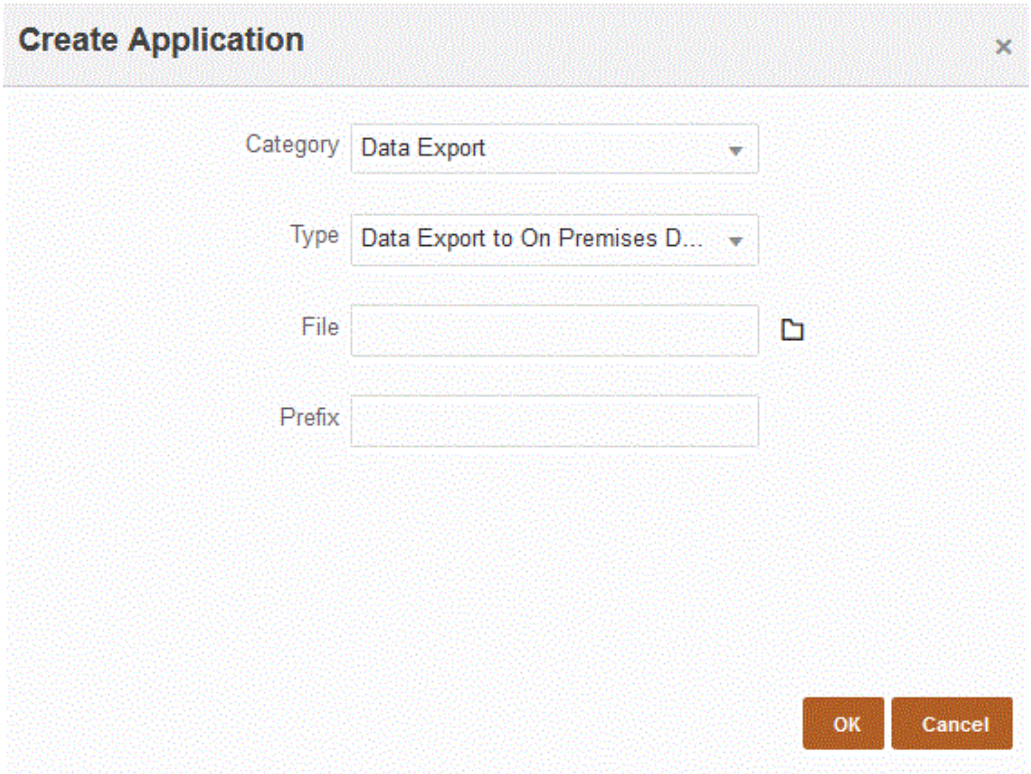
e. From **File**, select the file that you created in step 2.

Click to browse for the file on the File Browser page.

f. **Optional**: In **Prefix**, specify a prefix to make the application name unique.

The prefix is concatenated with the file name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.

g. Click **OK**.

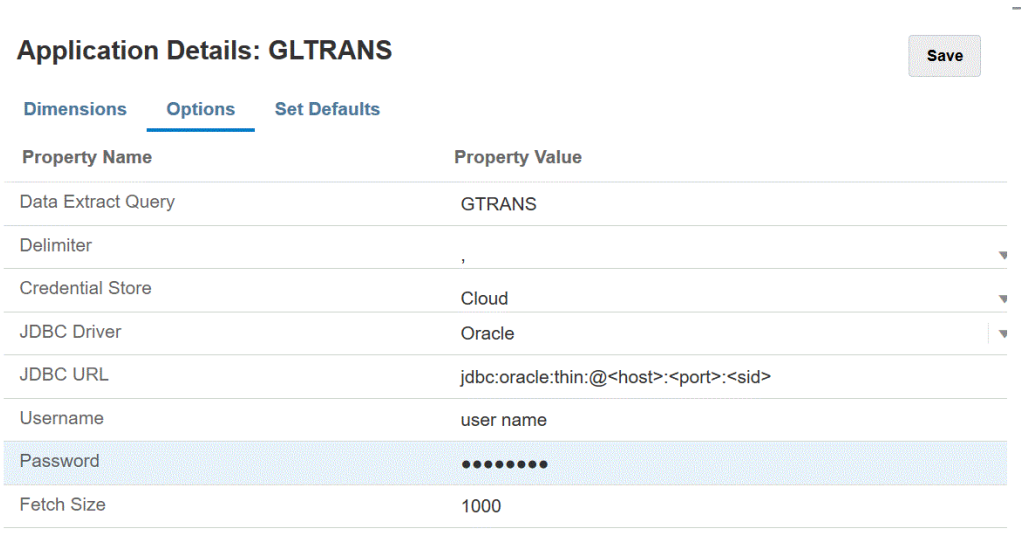


4. **Define the application details for the on-premises database file application**:

Copyright © 2017, 2026, Oracle and/or its affiliates.

a. On the **Application** page, select the on-premises database application and then from the **Actions** menu, select **Application Details** tab.

b. Select the **Options** tab.



c. In **Data Extract Query**, specify the name of the SQL query to run against the file.

d. In **Delimiter**, select the type of delimiter used in the file:

- Comma (,)
- Exclamation (!)
- Semicolon (;)
- Colon (:)
- Vertical bar (|)

e. In **Credential Store**, specify the type of credential store used by the EPM Integration Agent.

Available types of credential stores include the following:

- Cloud
- File

For the **Cloud** credential store type, store the *username/password/connect* string in the application.

For the **File** credential store type, create a file in the on-premises environment to store the JDBC URL, user name, and password. The file name must be named *appname*`.cred` and stored in the `config` directory.

The file must contain the following lines:

```
driver=oracle.jdbc.driver.OracleDriver
jdbcurl=jdbc:oracle:thin:@host:port/service
username=apps
password=w+Sz+WjKpL8[
```


Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> The password used for both credential store types must be encrypted. When the type is a "Cloud" store, type the password in the user interface in the usual way. The system encrypts and stores the password.
>
> When the type is a "File" store, you must encrypt the password using the `encryptpassword` utility and store the password in the file. For more information about encrypting your password, see Encrypting the Cloud EPM User Password.

f. In **JDBC URL**, specify the JDBC driver URL connection string.

   The JDBC Driver URL connection string enables you to connect to a relational database using Java.

   For an Oracle Thin JDBC driver, the JDBC driver URL includes the following:

   `jdbc:oracle:thin:@host:port:sid`

   `jdbc:oracle:thin:@host:port/service`

   For a MS SQL Server, the JDBC driver URL includes:

   `jdbc:sqlserver://server:port;DatabaseName=dbname`

g. In **Username**, specify the on-premises database user name.

h. In **Password**, specify the on-premises database password.

i. In **Fetch Size**, specify the number of rows fetched (that the driver processes) with each database round trip for a query.

j. Click **Save**.

5. **Create the integration job between the on-premises data source and Cloud EPM:**

   a. From the **Data Integration** home page, click (Add icon).

   b. On the **Create Integration** page, and then in **Name** and **Description**, enter a name and description for the integration job.

   c. In **Location**, enter a new location name, or pick an existing location to specify where to load data.

   d. Tap the **Quick Mode** slider on.

      When you associate an integration job with the Quick Mode method and then save the job, you cannot reverse the Quick Mode association. However; the integration job can be deleted.

   e. Click (Select a Source).

   f. From the **Select a Source** drop-down ( ), select the on-premises database data source application.

   g. From the **Select a Target** drop-down ( ), select the on-premises data export file application.

   h. Click **Save and Continue**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

6. **Map the dimensions between the on-premises database data source and the on-premises database file application**:
   a. On the **Mapping Dimensions** page, from **Type**, select the type of data load method.

      Available options:

      - Delimited - Numeric data: supports numeric data types only.
      - Delimited - All data types supports the following data types to Planning:
        - numbers
        - text
        - Smartlists
        - Data

   b. In the mappings grid, map the source columns in the on-premises data source application to the dimensions in the Cloud EPM application by completing the following:

      i. In **Select Source Dimension**, select the name of the source dimension to map to the dimension.

      ii. **Optional**: Add a target expression for each of the Cloud EPM dimensions.

      For information on using target expressions, see Using Target Expressions.

      > **Note**
      >
      > You cannot use a SQL target expression type with the Quick Mode method.
      > Source expression types are not available with the Quick Mode method.

   c. Click **Save and Continue**.

7. From the **Data Integration** home page, click ... to the right of the direct integration, and then select **Options**
8. Click the **Options** tab.
9. In **Category**, specify an explicit source filter for Scenario dimension.

   The Category is a required component of POV for processing so a category must be specified when defining the integration. The category is not used to determine the scenario for Quick Mode processing.
10. In **Data Extract Option**, select the method for extracting data:

    Available options:

    - All Data—Extracts stored values and dynamically calculated values for both the Dense and Spare dimensions.

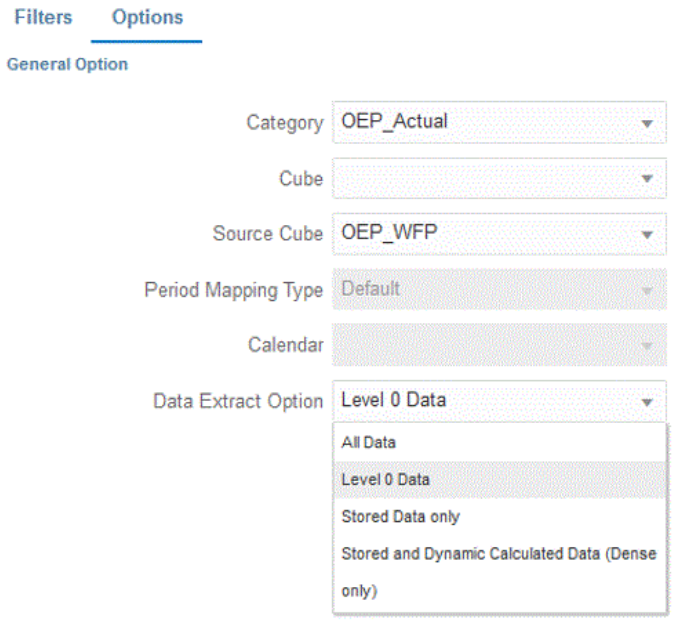
      The All Data data extract option uses the MDX Query Export method to extract data.
    - Stored and Dynamic Calculated Data—Extracts for stored members and dynamic calculated members for the Dense dimension only and not Spare dimensions. The Stored and Dynamic Calculated Data option uses the DATAEXPORT command method to extract data.

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Stored Data Only—Extracts stored data only. Dynamically calculated values are excluded in this type of extract. The Stored Data Only option uses the DATAEXPORT command method to extract data.
- Level 0 Data—Extracts entire members at the bottom of dimension (raw data stored in a database) anatad enables you to apply filters and select columns to include in the output file. This extract option also enables you to export non-numeric data. The Level 0 Data option uses the MAXL Export method to extract data. This method can be used only by the Service Administrator. The application is read only when the extract step is executing.

  If you select the Quick Mode method, the following target options are not available:

  – Export Attribute Columns
  – Accumulate Data
  – Sort Data
  – Pivot Dimension



11. Click **Save**.
12. **Run the Integration**:
    a. From the **Data Integration** home page, select the integration job associated with the Quick Mode load and then click ▶.
    b. From the **Run Integration** page, the default value for the **Mode** is **Replace**.

    When running the Quick Mode load, valid export modes are:

    - For Planning—Replace, Merge, and Accumulate
    - For Financial Consolidation and Close—Replace, Merge, Accumulate

    If using Quick Mode in Merge mode, and there are any errors during the load, the load fails and nothing is loaded. If using Replace mode, valid data will is loaded, and invalid data is not loaded. Replace mode results in a partial load of valid data.

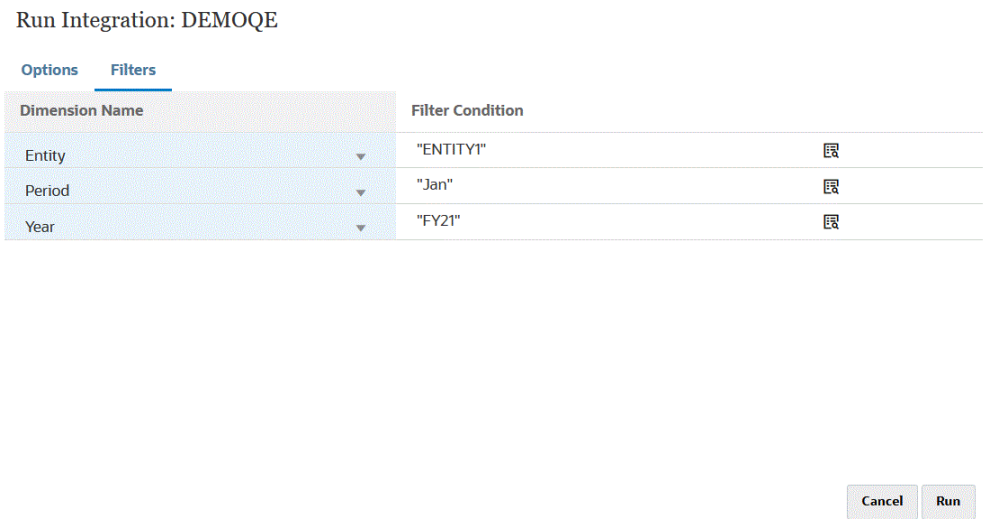
    Replace is the default mode.

Copyright © 2017, 2026, Oracle and/or its affiliates.

c. If no period was selection on the **Options** page, then from the **Period** drop-down, select the single period from the source file from which to load data.

   If you map a period dimension and specify a target expression for period on the Map Dimension page for the integration, then the **Period** drop-down is not available for selection because the period is derived from the mapping.

d. If filters have been defined for the integration, click the **Filters** tab and make any changes as desired.



e. Click **Run**.

The following example shows the result of the data export filtered by entity and period.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Data Export_1981.dat.txt - Notepad

File Edit Format View Help

```
Period,Company,Product,Account,Amount
Jan,ENTITY1,PRODUCT1,ACCOUNT1,-123
Jan,ENTITY1,PRODUCT1,ACCOUNT2,0.09999999999854481
Jan,ENTITY1,PRODUCT1,ACCOUNT3,333.5669999999955
Jan,ENTITY1,PRODUCT1,ACCOUNT4,41111
Jan,ENTITY1,PRODUCT1,ACCOUNT5,51111
Jan,ENTITY1,PRODUCT1,ACCOUNT6,61111
Jan,ENTITY1,PRODUCT1,ACCOUNT7,71111
Jan,ENTITY1,PRODUCT1,ACCOUNT8,81111
Jan,ENTITY1,PRODUCT1,ACCOUNT9,91111
Jan,ENTITY1,PRODUCT1,ACCOUNT1,11112
Jan,ENTITY1,PRODUCT1,ACCOUNT2,21112
Jan,ENTITY1,PRODUCT1,ACCOUNT3,31112
Jan,ENTITY1,PRODUCT1,ACCOUNT4,41112
Jan,ENTITY1,PRODUCT1,ACCOUNT5,51112
Jan,ENTITY1,PRODUCT1,ACCOUNT6,61112
Jan,ENTITY1,PRODUCT1,ACCOUNT7,71112
Jan,ENTITY1,PRODUCT1,ACCOUNT8,81112
Jan,ENTITY1,PRODUCT1,ACCOUNT9,91112
Jan,ENTITY1,PRODUCT1,ACCOUNT1,11113
Jan,ENTITY1,PRODUCT1,ACCOUNT2,21113
Jan,ENTITY1,PRODUCT1,ACCOUNT3,31113
Jan,ENTITY1,PRODUCT1,ACCOUNT4,41113
Jan,ENTITY1,PRODUCT1,ACCOUNT5,51113
Jan,ENTITY1,PRODUCT1,ACCOUNT6,61113
Jan,ENTITY1,PRODUCT1,ACCOUNT7,71113
Jan,ENTITY1,PRODUCT1,ACCOUNT8,81113
Jan,ENTITY1,PRODUCT1,ACCOUNT9,91113
Jan,ENTITY1,PRODUCT1,ACCOUNT1,11114
Jan,ENTITY1,PRODUCT1,ACCOUNT2,21114
Jan,ENTITY1,PRODUCT1,ACCOUNT3,31114
Jan,ENTITY1,PRODUCT1,ACCOUNT4,41114
Jan,ENTITY1,PRODUCT1,ACCOUNT5,51114
Jan,ENTITY1,PRODUCT1,ACCOUNT6,61114
Jan,ENTITY1,PRODUCT1,ACCOUNT7,71114
Jan,ENTITY1,PRODUCT1,ACCOUNT8,81114
Jan,ENTITY1,PRODUCT1,ACCOUNT9,91114
Jan,ENTITY1,PRODUCT1,ACCOUNT1,11115
Jan,ENTITY1,PRODUCT1,ACCOUNT2,21115
Jan,ENTITY1,PRODUCT1,ACCOUNT3,31115
Jan,ENTITY1,PRODUCT1,ACCOUNT4,41115
Jan,ENTITY1,PRODUCT1,ACCOUNT5,51115
```


Copyright © 2017, 2026, Oracle and/or its affiliates.

Defining Periods Used with the Quick Mode Method

When defining periods used with the Quick Mode method, you can either select a single period at run time or you can map a period column in the data source and the Oracle Fusion Cloud Enterprise Performance Management period is derived from this column.

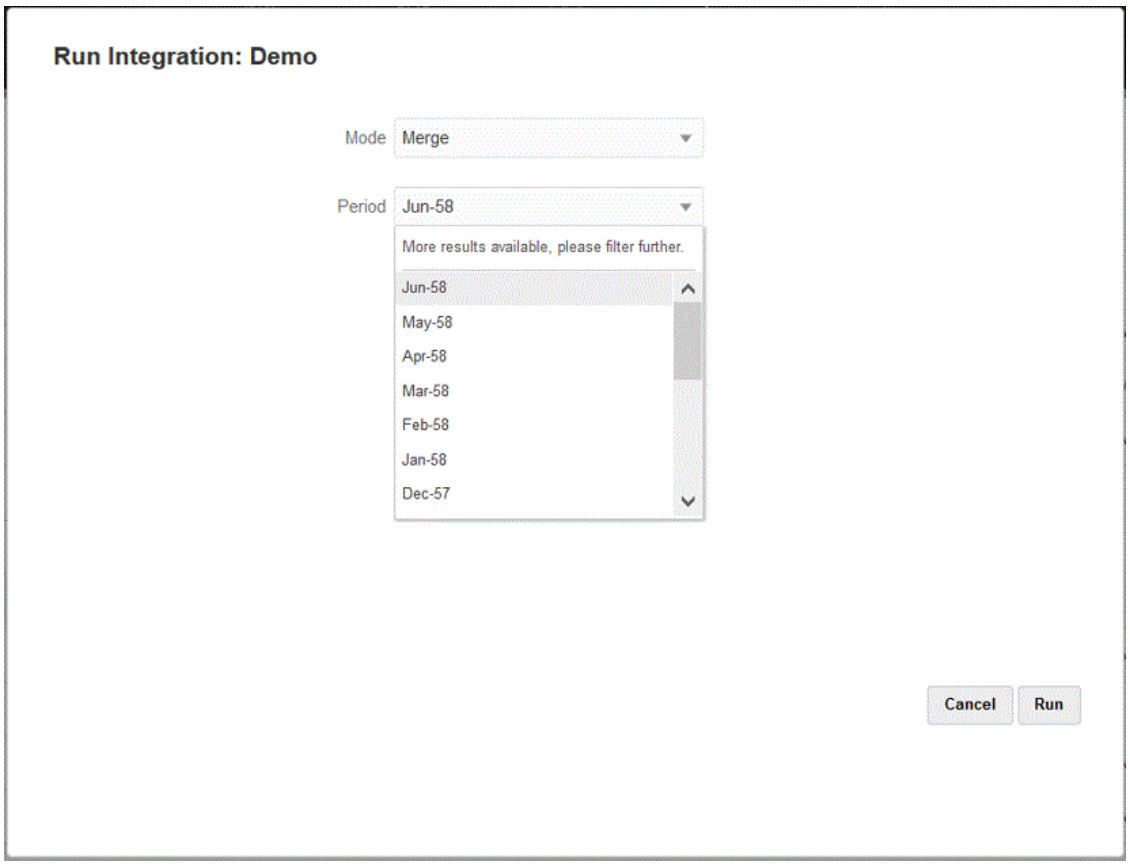
For more information, see:

- Selecting a Single Period for the Quick Mode Method
- Selecting a Period Derived from the Source Column for the Quick Mode Method

Selecting a Single Period for the Quick Mode Method

Describes how to select a single period to which to load data using Quick Mode.

When determining which period to use for the Quick Mode method, select a single period to which to load data. In this case, you simply select the period on the Run Integration page as shown below.



> **Note**
>
> If you map a period dimension and specify a target expression for period on the Map Dimension page for the integration, then the Period drop-down is not available for selection because then the period is derived from the mapping.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Selecting a Period Derived from the Source Column for the Quick Mode Method

Describes how to select a period column in the data source to which to load data using Quick Mode.

If the source data includes a period column in the data source, the Oracle Fusion Cloud Enterprise Performance Management period can be derived from this column.

The source data can include period in various formats like below, such as:

- Jan-20
- January-20
- 01-20
- 01/31/20
- 01-Jan-20

You can also use target expressions to derive the target Year and Period values, or derive the Year and Period using a date field and associated format. For information on using target expressions, see Using Target Expressions.

To derive the target Year and Period values, use the following target expressions:

- `substring()`—Extract and return characters from a string based on a character position (starting position) and substring length (number of characters to extract).

  For more information, see Substring.

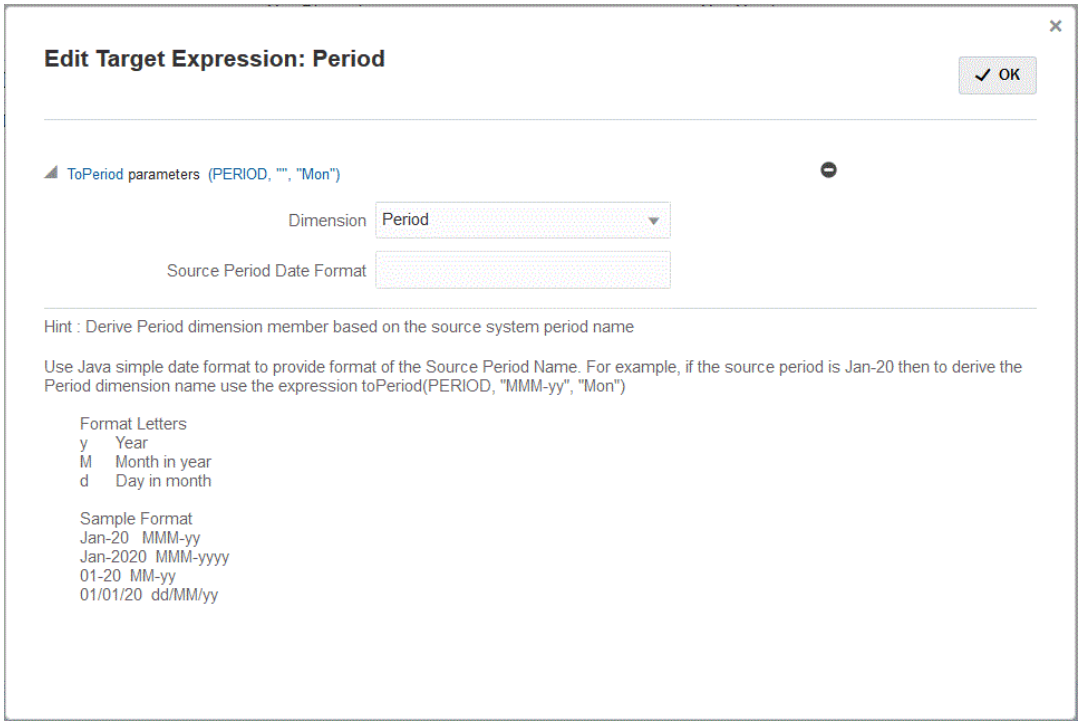
- `split()`—Split the source value based on a delimiter, and return the *n* value after splitting the value. This expression type is useful for splitting segment values from strings.

  For more information, see Split.

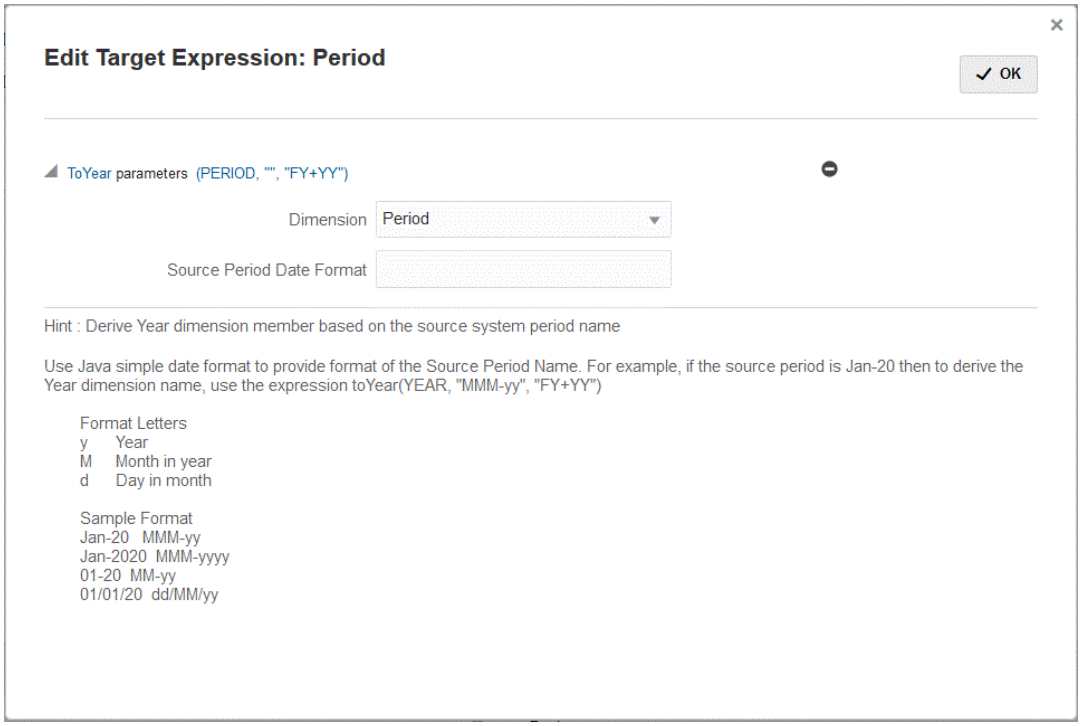
To derive the Year and Period using a date field and associated format, use these target expressions.

- `toPeriod()`—Derive Period dimension member based on the source system period name using the syntax: `toPeriod(field, "<DATE FORMAT>", Mon)`.

Copyright © 2017, 2026, Oracle and/or its affiliates.



- `toYear()`—Derive Year dimension member based on the source system period name using the syntax: `toYear(field, "<DATE FORMAT>", "FY+YY")`.



The `<DATEFORMAT>` is in the Simple Java Date format. For more information about the Simple Java Data format, see https://docs.oracle.com/javase/8/docs/api/java/text/SimpleDateFormat.html.

Copyright © 2017, 2026, Oracle and/or its affiliates.

`Map` is another target expression that enables you to accept an input field or field with an expression like *substr*, *split*, and series of key value pairs for mapping using the syntax: `map(substr(field,1,2),P1:Jan|P2:Feb|P3:Mar|….)`

> **Note**
>
> The `map` target expression can be used for any expression.

Writing Back with the EPM Integration Agent

Describes how to write back with the EPM Integration Agent using either a simple or advanced method.

The EPM Integration Agent supports write-back by enabling you to move data from your Oracle Fusion Cloud Enterprise Performance Management applications to your on-premises database. To use this feature, you register a data export application to export data to the on-premises database.

- Simple method—you register the application and specify the table name and include all the columns from the table in the application. The system generates the INSERT statements automatically and loads the data.

  To use this method, create a CSV file with the list of columns to export. The names of the column must exactly match the column name in the table. The name of the file must be the name of the application.

- Advanced method—you specify an INSERT statement in the query definition page. The INSERT statement contains the table and column for values and the dimension name of the application. Using this method, you can provide a user-friendly dimension name and also use SQL functions like TO_DATE, TO_NUMBER and other functions to perform data type conversions and other string operations.

  To use this method, create a CSV file with the list of columns to export. The list of columns must exactly match the column name you use in the INSERT statement. The name of the file must be name of the application.

Additionally, the agent can execute two event scripts during the write-back execution: `BefExport` and `AftExport`. Using a `BefExport` event, you can perform any action before inserting data to the table or you can override the default insert processing. Using an `AftExport` event, you can do any post processing cleanup after inserting data to the table.

The write-back feature is supported for agents running in both SYNC and ASYNC modes.

Process Description for EPM Integration Agent Write-Back

To write back to an on-premises database using the EPM Integration Agent:

1. Download the **EPMAgent ZIP**.

   If this is your first installation of the EPM Integration, see Installing and Configuring the EPM Integration Agent.

   If you have an existing installation of the EPM Integration Agent, download and unzip the EPMAgent.ZIP. You do not need to update the *ini* file or certificate.

2. Optional: You can execute two event scripts during the write-back execution including the *BefExport* and *AftExport*.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Using a *BefExport* event, you can perform any action before inserting data to the table or you can override the default insert processing. Using an *AftExport* event, you can perform any post processing cleanup after inserting data to the table.

3. Register the data export to an on-premises application.

   For more information, see Registering the Data Export to On-Premises Application.

4. Go to **Application Details Dimensions** tab for the data export application and verify that the column names in the header record must exactly match the column name or column alias in the table to which you are loading data.

   Names are case-sensitive.

   For more information, see Setting the Application Detail Dimensions for the Write-Back.

5. Designate the "Amount" column in the Target Dimension Class and clear the Data Column Name column.

6. Go to the **Application Details Options** tab for the data export to on-premises application and specify how you want to handle loading data. You can specify the table name and have the system generate the INSERT statement automatically. You can also write an INSERT query in the query definition page and use the statement as the method for loading data.

   You also need to specify any credentials, JDBC connection information, and user name and password for the on-premises database.

   For more information, see Defining Application Details for A Data Export Application.

7. Integrate the Oracle Fusion Cloud Enterprise Performance Management application with the data export file by completing the following tasks.

Task	Description	More Information
General	Add or edit an integration for file-based and direct integration sources.	Defining a Data Integration
Map Dimensions	Map the columns in the data source to dimensions in the target application.	Mapping Dimensions
Map Members	Map dimensions to identify how source field values translate to target dimension members.	Mapping Members
Options	Define options for importing and exporting data. Also, define any source filters.	Setting Data Integration Options

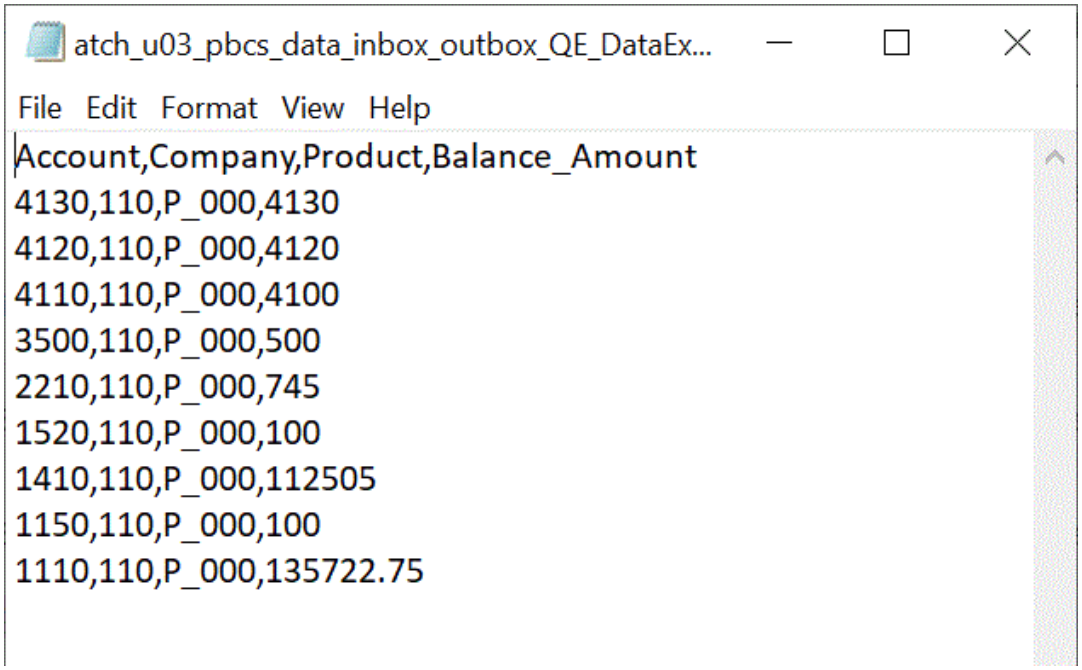
8. Run the integration.

   For information on running an integration, see Running an Integration.

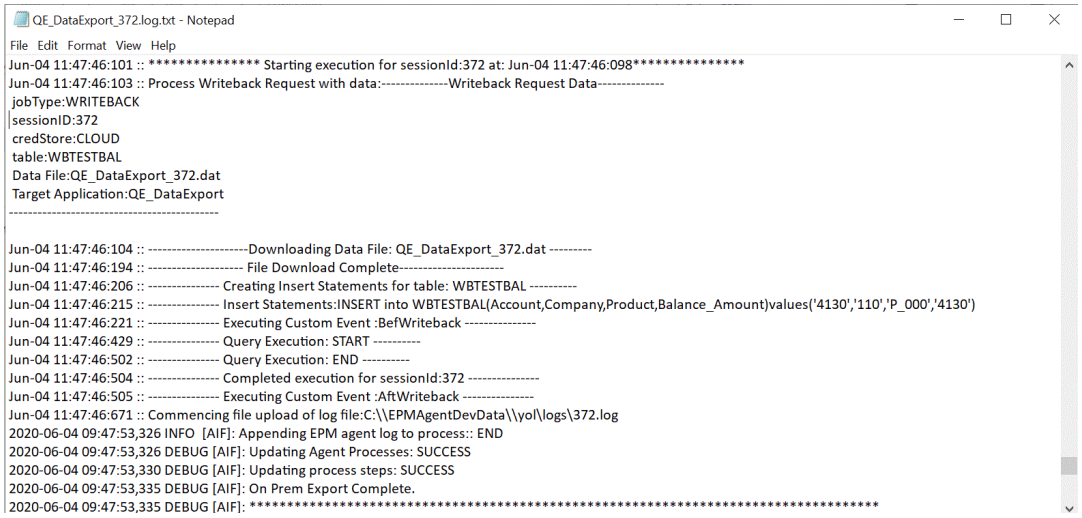
   When the integration is executed, the EPM Integration Agent initiates the export process in the Cloud EPM. The agent downloads the export data. Based on the selected method, the agent also constructs the appropriate INSERT statement and loads the data to the target table.

   You can review the exported data by downloading the output file from the Process Details. The database column names for the write-back are specified as the column headers in the generated file as shown below.

Copyright © 2017, 2026, Oracle and/or its affiliates.



You can view the details of running the write-back by opening the log file for the job from Process Details as shown below:



Registering the Data Export to On-Premises Application

Describes how to register an Data Export to On-Premises application.

Register the data export to an on-premises application to specify the file associated with the columns to which to export data.

To register a data export application

1. Create a CSV file with the list of columns to be exported.

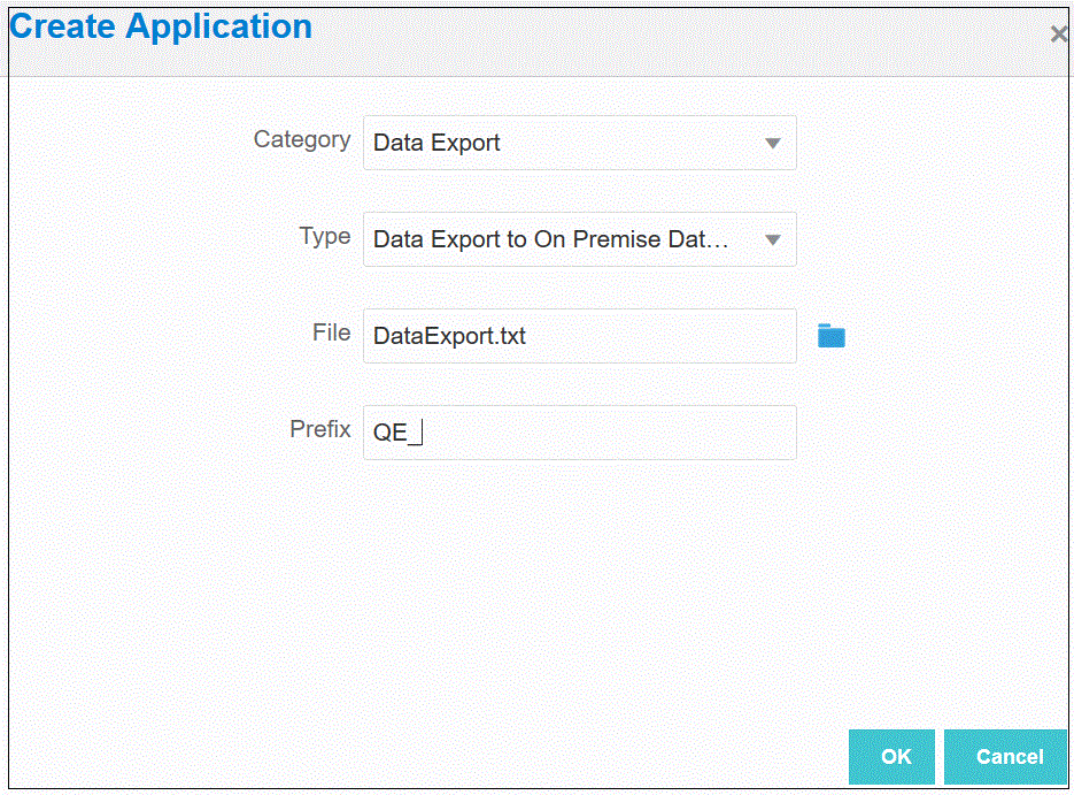
   The name of the CSV file must be name of the application. The name of the columns must be name of the columns in the table if you are using the simple method. For an INSERT query, the name of the columns can be any name but must match the names in the INSERT statement.

Copyright © 2017, 2026, Oracle and/or its affiliates.

2. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Setup** actions drop-down, select **Applications**.
3. On the **Applications** page, click (Add icon).
4. From **Category**, select **Data Export**.
5. From **Type**, select **Data Export to On Premise**.
6. From **File Name**, select the name of the file from **step 1**.

   You can click to navigate and select the file from the **File Browser** page.
7. **Optional**: In **Prefix**, specify a prefix to make the application name unique.

   The prefix is concatenated with the file name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.
8. Click **OK**.



Setting the Application Detail Dimensions for the Write-Back

As part of the process for writing back to the on-premises database, you need to verify that the column names in the header record match exactly.

You must also designate the "Amount" dimension with the Target Dimension Class "Amount" and clear any value in the Data Column Name column.

To designate the Amount column in the application detail:

1. From the **Application** page, click to the right of the data export to on-premise application, and then select **Application Details**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

2. From the **Application Details** page, click the **Dimensions** tab.
3. From the **Dimension Name** column, select the **Amount** dimension.
4. Change the classification to **Amount** and remove the data column name.
5. Click **Save**.

Defining Application Details for A Data Export Application

Describes how to specify credential and JDBC connection information to an on-premises database.

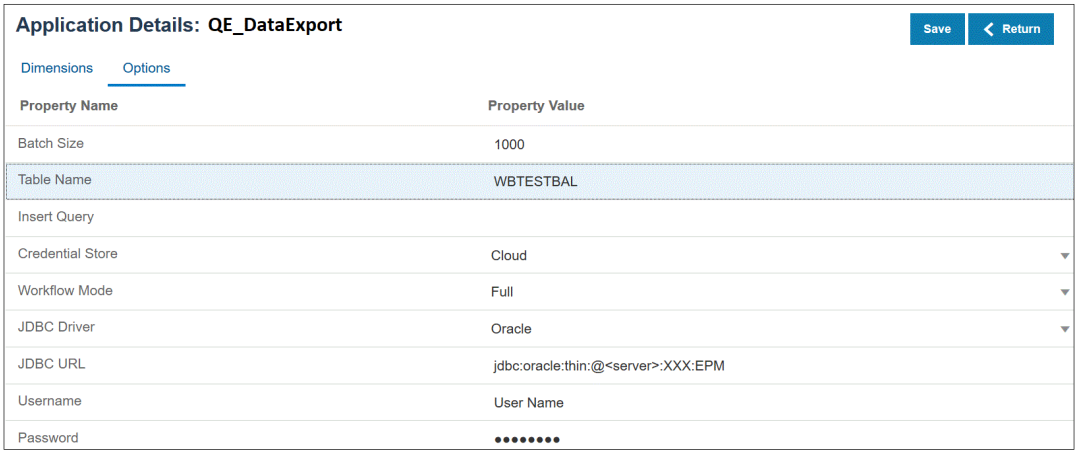
Use Application Details to specify how EPM Integration Agents handles the write-back of data. In addition, you also need to specify credential and JDBC connection information to an on-premises database.

To define application details for an data export to on-premises application:

1. From the **Application** page, click ⋯ to the right of the data export to on-premises application, and then select **Application Details**.
2. From the **Application Details** page, click the **Options** tab.
3. In **Table Name**, specify the table name to which to load the data.

   The name of the dimensions in the data export to on-premises application must match the names of the columns in the table. The system generates the INSERT statement to load the data automatically and inserts text for all fields.

   If you specify a table name in this field, do not specify an INSERT statement in the **Insert Query** field.

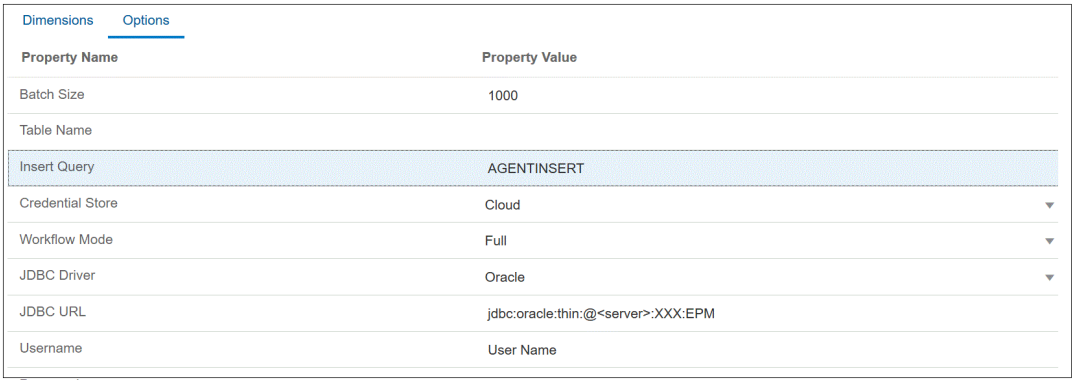


4. In **Insert Query**, specify the name of the SQL query associated with the custom INSERT statement created on the Query Definition page.

   INSERT query must contain the table and columns for values and the dimension name of the application. Using this method, you can provide a user-friendly dimension name and also use SQL functions like TO_DATE, TO_NUMBER and other functions to perform data type conversions and other string operations. For more information, see Writing Back with a Custom INSERT Query.

   If you specify an INSERT query, do not specify a table name in the **Table Name** field.

Copyright © 2017, 2026, Oracle and/or its affiliates.



5. In **Credential Store**, specify the type of credential store used by the EPM Integration Agent.

   Available types of credential stores include the following:

   - Cloud
   - File

   For the **Cloud** credential store type, store the *username/password/connect* string in the application.

   For the **File** credential store type, create a file in the on-premises environment to store the JDBC URL, user name, and password. The file name must be named *appname*`.cred` and stored in the `config` directory.

   The file must contain the following lines:

   ```
   driver=oracle.jdbc.driver.OracleDriver
   jdbcurl=jdbc:oracle:thin:@host:port/service
   username=apps
   password=w+Sz+WjKpL8[
   ```

   > **Note**
   >
   > The password used for both credential store types must be encrypted. When the type is a "Cloud" store, type the password in the user interface in the usual way. The system encrypts and stores the password.
   >
   > When the type is a "File" store, you must encrypt the password using the `encryptpassword` utility and store the password in the file. For more information about encrypting your password, see Encrypting the Cloud EPM User Password.

6. In **JDBC Driver**, select the type of JDBC driver to use when connecting to the Database Management System (DBMS).

   Available types of JDBC drivers include the following:

   - Microsoft SQL Server
   - Oracle

7. In **Username**, specify the on-premises database user name.
8. In **Password**, specify the on-premises database password.
9. Click **Save**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Writing Back with a Custom INSERT Query

When writing back data using the EPM Integration Agent, you can write a custom INSERT query and use this query statement as the method when loading data. This feature enables you to provide user-friendly dimension names and also use SQL functions like TO_DATE, TO_NUMBER and other functions to perform data type conversions and other string operations.

To specify an INSERT query to use when writing back to an on-premises database:

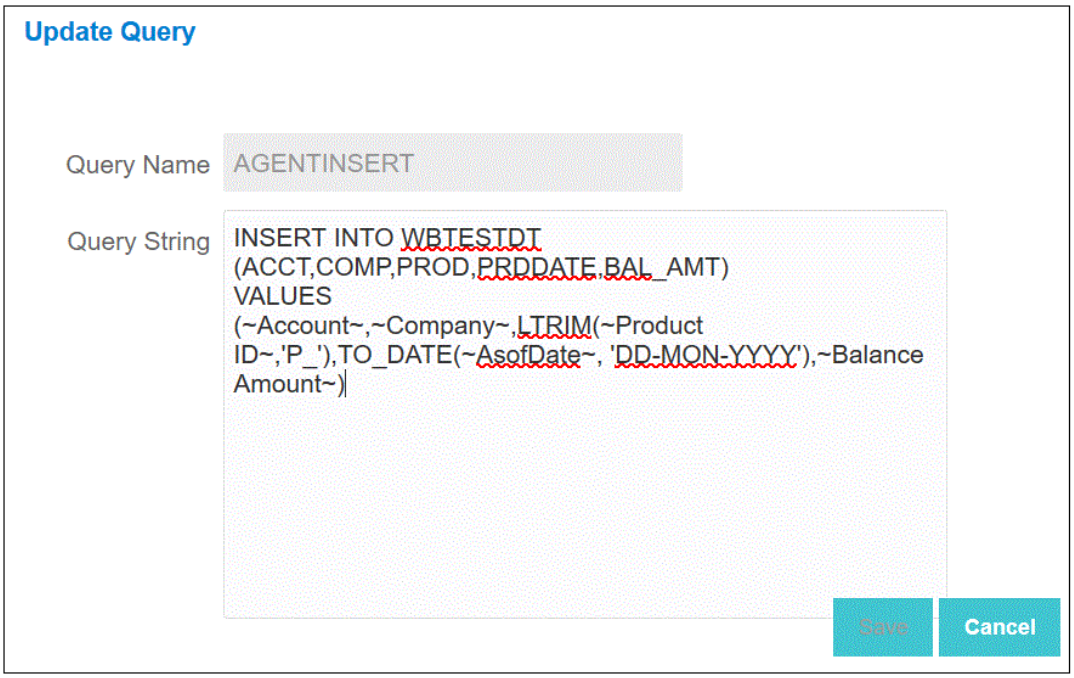
1. From the **Data Integration** home page, click **Actions**, then **Setup**, select **Query**.
2. On the **Query** page, click ✚.
3. From the **Create Query** page, and then in **Query Name**, specify the INSERT Query name.

   The query name is used in the INSERT Query field on the Application Details Options tab. For more information, see Defining Application Details for A Data Export Application.
4. In **Query String**, specify the INSERT statement used to write back the data based on the header row in the file.

   Write an INSERT statement with the table name and column name.

   In VALUES, provide the dimension name from the application enclosed in ~~. The system replaces these with actual values from the exported data file.

   In this example, "ACCT", "COMP", "PROD", "PRDDATE", "BAL_AMT" are in the WBTESTDT table. "Account,"" Company," "Product,""AsofDate," and "Balance Amount" are names of dimensions in the application.



5. Click **Save**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Practical Techniques for Writing Back to the EPM Integration Agent

This topics shows practical techniques for writing back to the EPM Integration Agent.

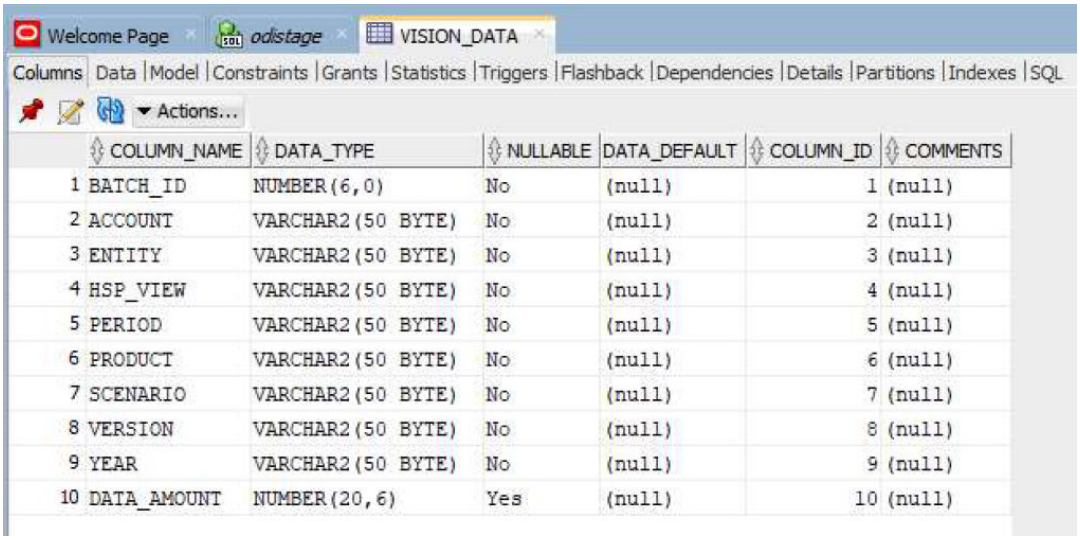
This topic explains the steps required to use the EPM Integration Agent to load data into an on-premises data warehouse from a Oracle Fusion Cloud Enterprise Performance Management application. Two different methods are available for loading data:

1. Load data by a table name in the Application option and the table columns in Application Dimensions respectively. (Insert query is automatically generated by EPM integration agent),
2. Load data by a table name and columns using a custom SQL Insert Query.

To use the EPM Integration Agent to load data into an on-premises data warehouse from an Cloud EPM application:

1. Create a **RDBMS** table where the data will be exported.

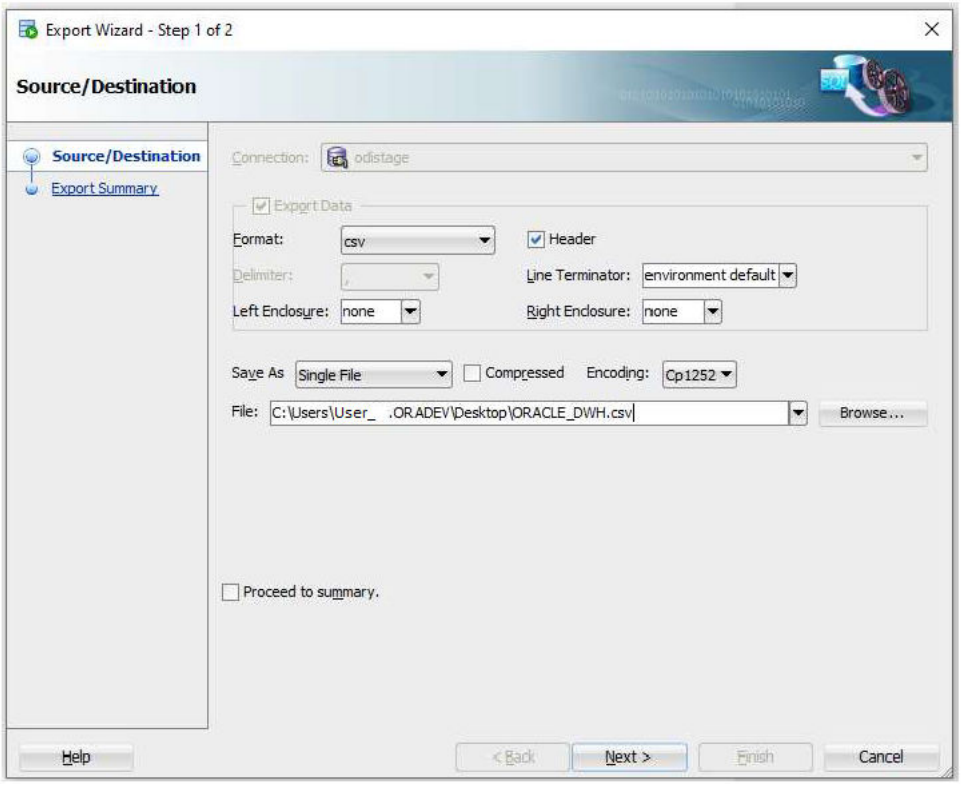
   In this example, a table named VISION_DATA has all the dimensions of the Vision application and two additional columns called BATCH_ID and DATA_AMOUNT. The BATCH_ID column is used to populate the Data Integration Process ID from the Cloud EPM. The DATA_AMOUNT column is used to populate the Amount column.



	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	BATCH_ID	NUMBER(6,0)	No	(null)	1	(null)
2	ACCOUNT	VARCHAR2(50 BYTE)	No	(null)	2	(null)
3	ENTITY	VARCHAR2(50 BYTE)	No	(null)	3	(null)
4	HSP_VIEW	VARCHAR2(50 BYTE)	No	(null)	4	(null)
5	PERIOD	VARCHAR2(50 BYTE)	No	(null)	5	(null)
6	PRODUCT	VARCHAR2(50 BYTE)	No	(null)	6	(null)
7	SCENARIO	VARCHAR2(50 BYTE)	No	(null)	7	(null)
8	VERSION	VARCHAR2(50 BYTE)	No	(null)	8	(null)
9	YEAR	VARCHAR2(50 BYTE)	No	(null)	9	(null)
10	DATA_AMOUNT	NUMBER(20,6)	Yes	(null)	10	(null)

2. Export the table contents to a CSV file for the application registration by including the header record for the column names.

Copyright © 2017, 2026, Oracle and/or its affiliates.

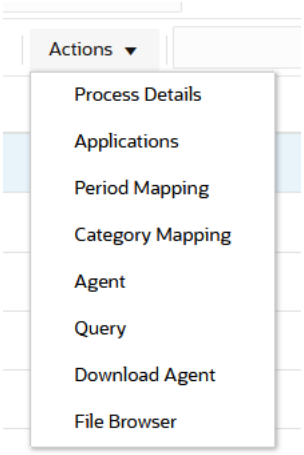


The CSV output file looks like this:

```
1  BATCH_ID,ACCOUNT,ENTITY,HSP_VIEW,PERIOD,PRODUCT,SCENARIO,VERSION,YEAR,DATA_AMOUNT
2
```

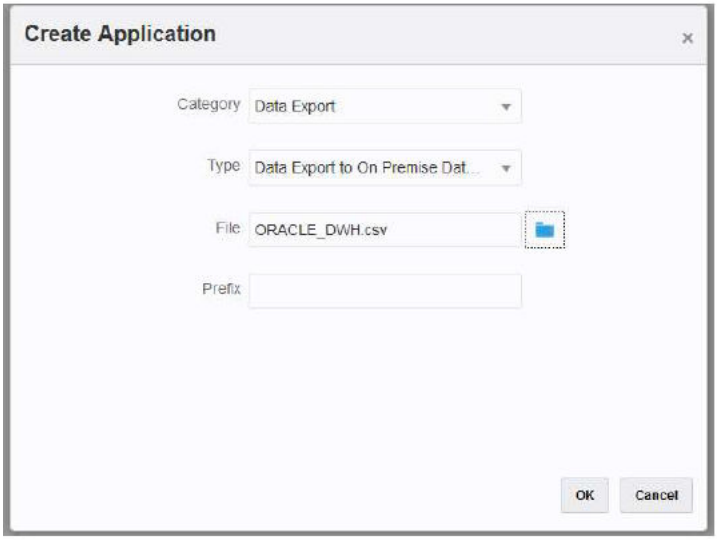
It is assumed that these columns are the exact representation of the target table columns. The SQL insert statement is generated based on the column names.

3. Navigate to **Data Exchange**, then **Data Integration** tab, then the **Action** menu, and then select **Applications**.

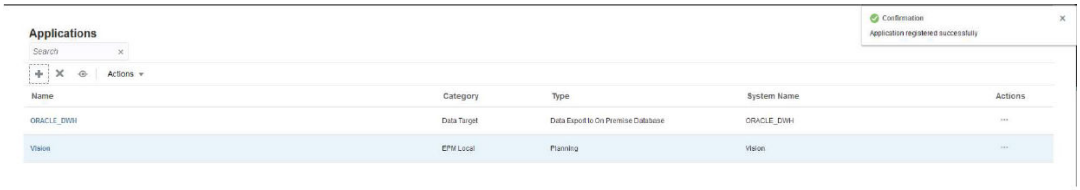


4. On the **Applications** page, register a **Data Export** application for the write-back process by completing the following:
   a. In **Category**, select **Data Export**.
   b. In **Type**, select **Data Export to On Premise Database**.
   c. Upload and select the file created from Step 2.

Copyright © 2017, 2026, Oracle and/or its affiliates.

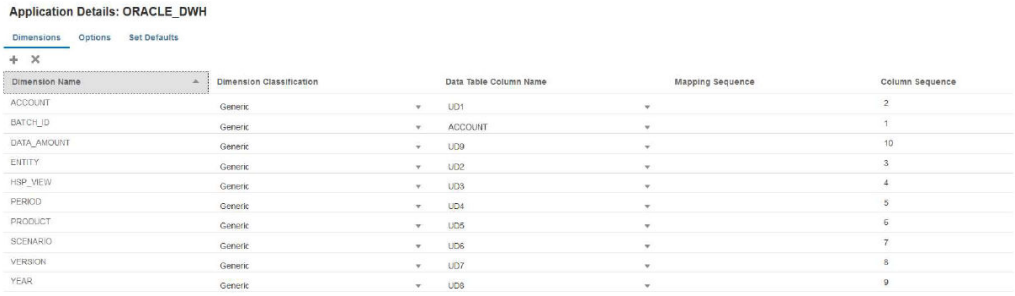


A "Data Export" application is created and the name of the application matches the file name used to register the application.



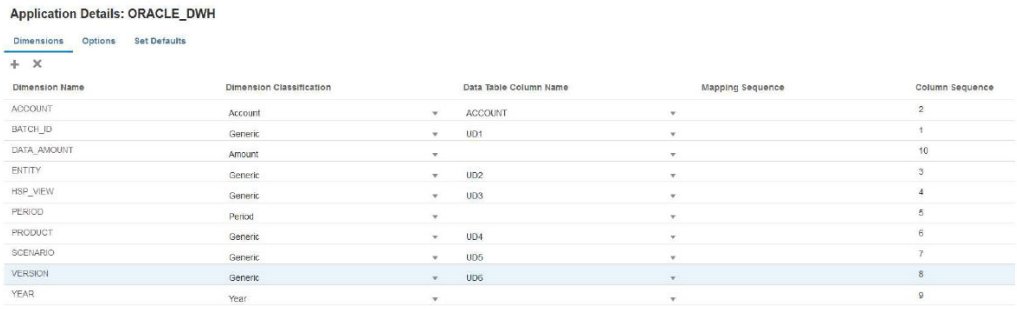
5. On the **Applications** page, click ••• to the right of the application, and then select **Application Details**.
6. Select the **Dimensions** tab.

   The application registration process automatically assigns "Generic" as a dimension classification for all columns in the CSV file and also assigns a "Data Table Column Name" accordingly.



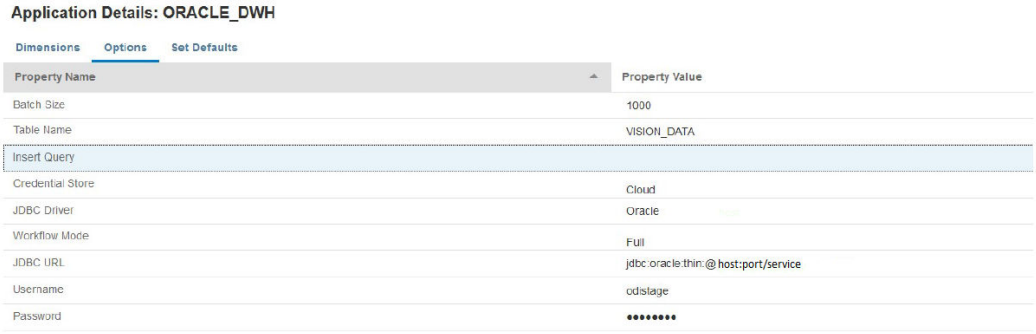
7. Classify the **Account**, **Amount**, **Period**, and **Year** columns appropriately:
   - Classify a column as **Account** when it is mapped to a source column in the import format and always has a value in the source field. In the following example, the "ACCOUNT" column has been classified as "Account." The Data Table Column Name for this dimension has been classified as "ACCOUNT."
   - Classify **DATA_AMOUNT** as "Amount" and leave the **Data Table Column Name** blank.
   - Classify **YEAR** as "Year" and leave the **Data Table Column Name** blank.
   - Classify **PERIOD** as "Period " and leave the **Data Table Column Name** blank.

Copyright © 2017, 2026, Oracle and/or its affiliates.



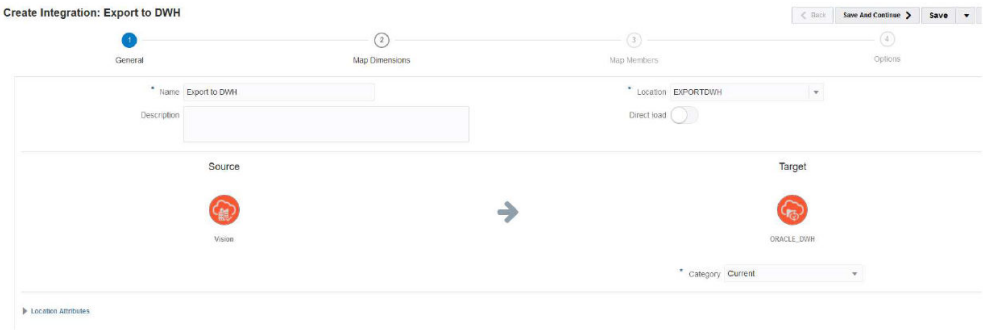
8. From the **Applications** page, click ⋯ to the right of the application, and then select **Application Detail**.
9. Select the **Options** tab.
10. Specify the **Table Name**, **JDBC URL** and database credentials.

    Leave the **Insert Query** field blank. If you don't specify "Insert Query," it is assumed that the application dimensions denote the exact same name of the target database columns. The application dimensions should contain each of the columns of the target table that needs to be populated in the write-back process. Otherwise, the integration fails.



11. Create the integration between the source application and the target Cloud EPM application.

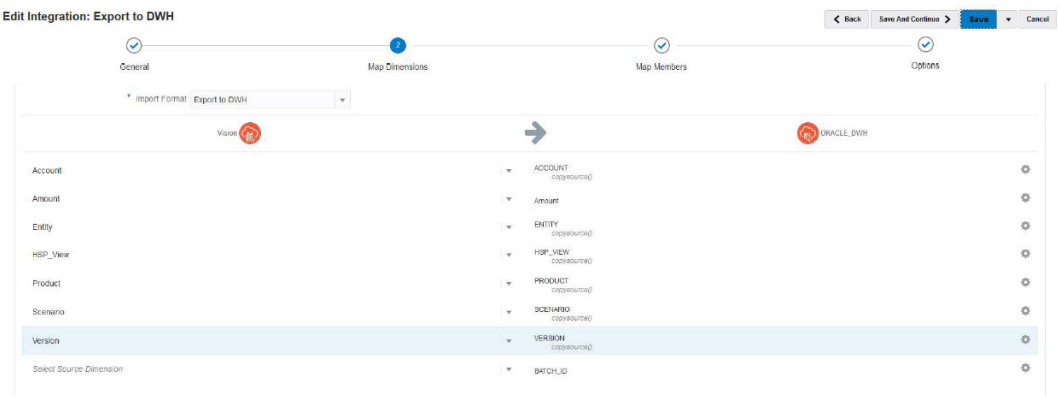
    The following example shows a source "Vision Planning" application and a target Data Export application called "ORACLE_DWH."



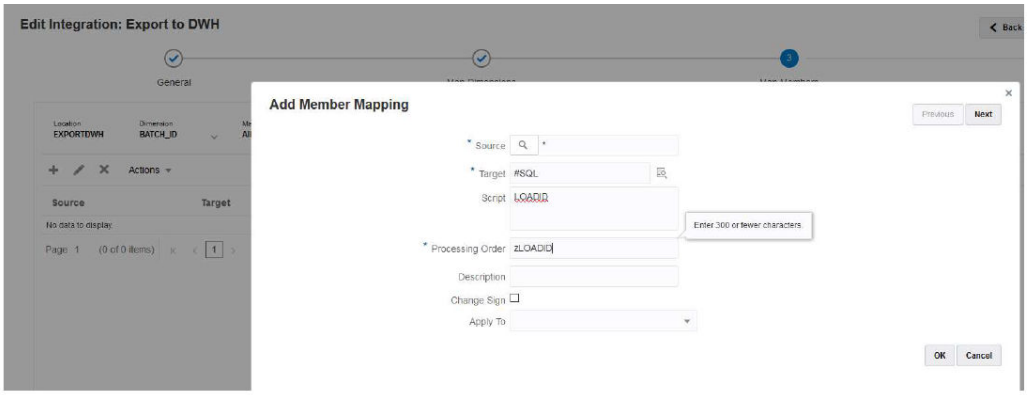
12. Map all the columns and use "copysource()" as the target expression for all dimensions except "Amount" and "BATCH_ID."

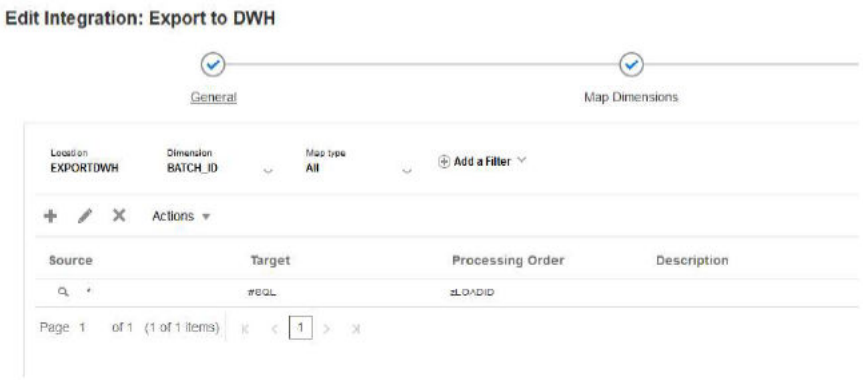
    An expression is applied to "BATCH_ID" in member mappings. The "Amount" dimension doesn't have a target expression.

Copyright © 2017, 2026, Oracle and/or its affiliates.



13. From **Map Members**, create an #SQL mapping for BATCH_ID using the LOADID column of the TDATASEG_T table.

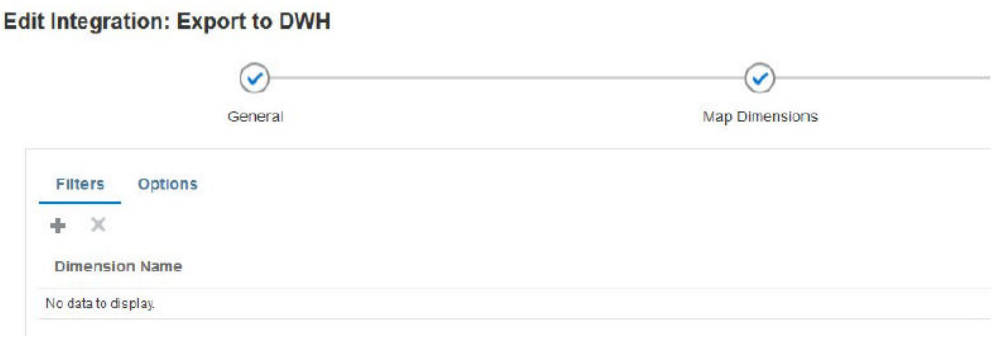




14. From **Options**, then **Filters**, specify any source planning application filters as needed.

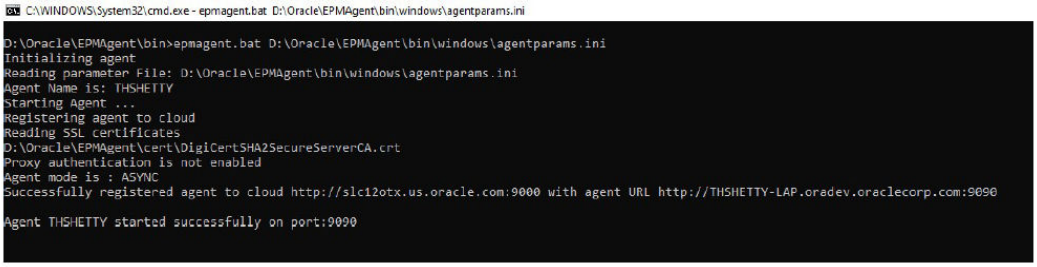
    In the following example, no filters were selected because all data from the Vision application (for the start/end period range) are imported and exported to the target application.

Copyright © 2017, 2026, Oracle and/or its affiliates.



15. From **Options**, then **Source Cube**, specify the source cube (plan type). Then from **Period Mapping Type**, specify **Default**. Then click **Save** as shown below.

16. Start the EPM Integration Agent on the on-premises host where the target database is accessible.

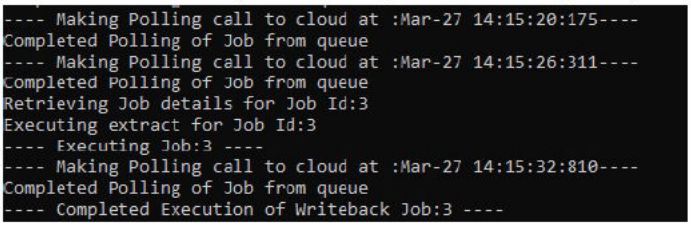


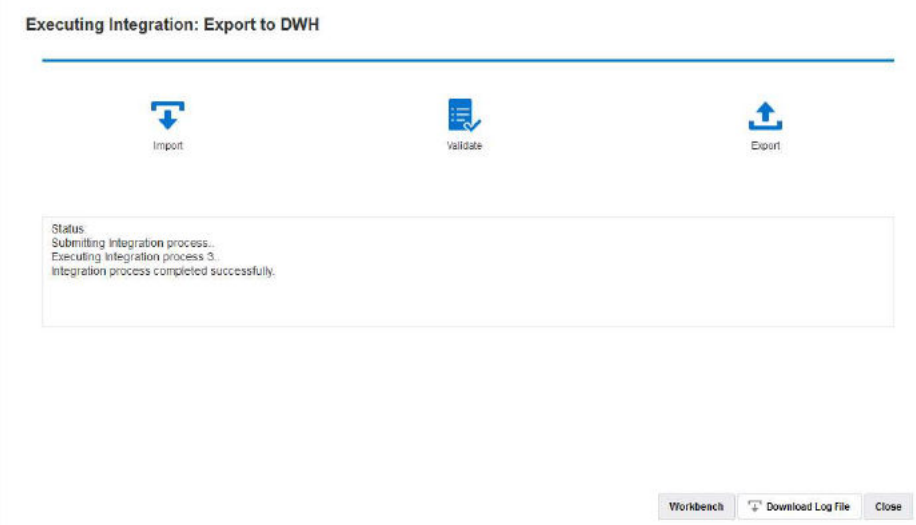
17. In Data Integration, run the integration for the given period range.

    In the following example, the periods Jan-16 to Dec-16 were selected:

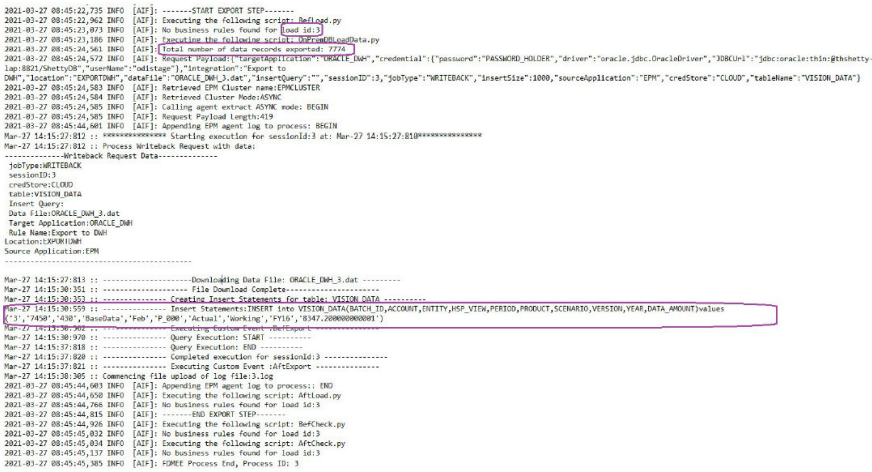
18. Observe the on-premises agent fetching the job details from Cloud EPM and wait until the process completes.

Copyright © 2017, 2026, Oracle and/or its affiliates.





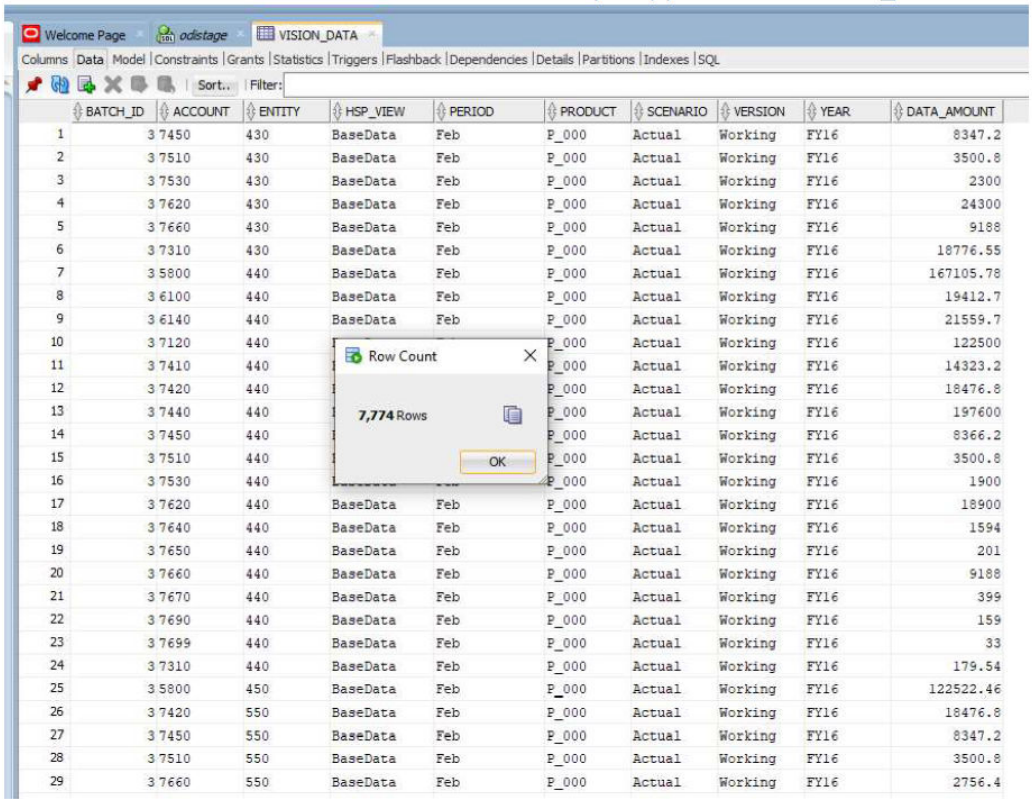
19. From **Process Details**, open the log file and note the total number of data records exported and the Insert Query statement:



20. Verify that the records got exported successfully in the target database.

    Note that the cloud Process ID is successfully mapped to the BATCH_ID column in the database.

Copyright © 2017, 2026, Oracle and/or its affiliates.



	BATCH_ID	ACCOUNT	ENTITY	HSP_VIEW	PERIOD	PRODUCT	SCENARIO	VERSION	YEAR	DATA_AMOUNT
1	3	7450	430	BaseData	Feb	P_000	Actual	Working	FY16	8347.2
2	3	7510	430	BaseData	Feb	P_000	Actual	Working	FY16	3500.8
3	3	7530	430	BaseData	Feb	P_000	Actual	Working	FY16	2300
4	3	7620	430	BaseData	Feb	P_000	Actual	Working	FY16	24300
5	3	7660	430	BaseData	Feb	P_000	Actual	Working	FY16	9188
6	3	7310	430	BaseData	Feb	P_000	Actual	Working	FY16	18776.55
7	3	5800	440	BaseData	Feb	P_000	Actual	Working	FY16	167105.78
8	3	6100	440	BaseData	Feb	P_000	Actual	Working	FY16	19412.7
9	3	6140	440	BaseData	Feb	P_000	Actual	Working	FY16	21559.7
10	3	7120	440			P_000	Actual	Working	FY16	122500
11	3	7410	440			P_000	Actual	Working	FY16	14323.2
12	3	7420	440			P_000	Actual	Working	FY16	18476.8
13	3	7440	440			P_000	Actual	Working	FY16	197600
14	3	7450	440			P_000	Actual	Working	FY16	8366.2
15	3	7510	440			P_000	Actual	Working	FY16	3500.8
16	3	7530	440			P_000	Actual	Working	FY16	1900
17	3	7620	440	BaseData	Feb	P_000	Actual	Working	FY16	18900
18	3	7640	440	BaseData	Feb	P_000	Actual	Working	FY16	1594
19	3	7650	440	BaseData	Feb	P_000	Actual	Working	FY16	201
20	3	7660	440	BaseData	Feb	P_000	Actual	Working	FY16	9188
21	3	7670	440	BaseData	Feb	P_000	Actual	Working	FY16	399
22	3	7690	440	BaseData	Feb	P_000	Actual	Working	FY16	159
23	3	7699	440	BaseData	Feb	P_000	Actual	Working	FY16	33
24	3	7310	440	BaseData	Feb	P_000	Actual	Working	FY16	179.54
25	3	5800	450	BaseData	Feb	P_000	Actual	Working	FY16	122522.46
26	3	7420	550	BaseData	Feb	P_000	Actual	Working	FY16	18476.8
27	3	7450	550	BaseData	Feb	P_000	Actual	Working	FY16	8347.2
28	3	7510	550	BaseData	Feb	P_000	Actual	Working	FY16	3500.8
29	3	7660	550	BaseData	Feb	P_000	Actual	Working	FY16	2756.4

21. Export the same data to another table by the name **VISION_DATA_2** using an Insert Query in the Cloud EPM.

    The INSERT Query enables you to use complex SQL expressions for mapping the columns of the database. The SQL expressions are executed on the target database and not on the cloud database (Oracle). This gives you flexibility to use SQL expressions specific to the type of database, for example, MS SQL server, MYSQ, etc. You can also derive column values using a sub-query from another table.

    In the following example:

    - BATCH_ID has been renamed to: BATCH_NUMBER.
    - ENTITY has been renamed to: ORGANIZATION.
    - The YEAR and PERIOD columns have been removed and a PERIOD_NAME column was created to populate both the Year and Period in the same column.
    - The scale of the DATA_AMOUNT has been reduced to store only 2 decimal places.

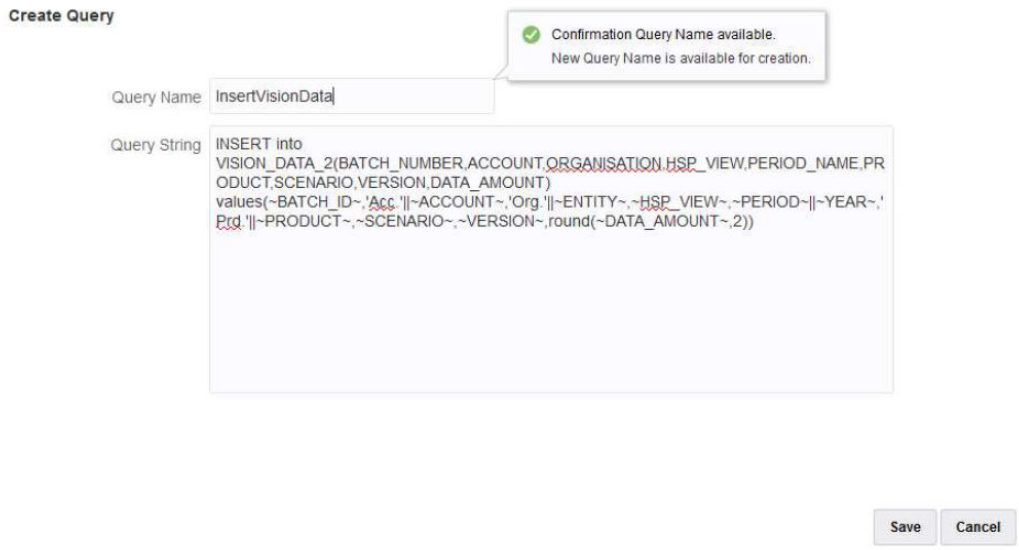
    You don't need a CSV file to register the application because you already have an application registered with a different set of column names. The advantage of using the SQL Insert query is that you can use the column names of the registered application to map the actual column names using the SQL Insert query. Consequently, the application can contain dimensions that are completely different than the actual column names of the target database. The application dimensions can also denote only a subset of the actual columns that are populated in the target database table.

    For more information, see: Writing Back with a Custom INSERT Query.

    The values in the INSERT Query are denoted by enclosing the target dimension name inside ~~ (tildes). For example, if a dimension by the name ABC in the Data Export application needs to be mapped to a DEF column in the database, specify the value for DEF column as **~ABC~**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

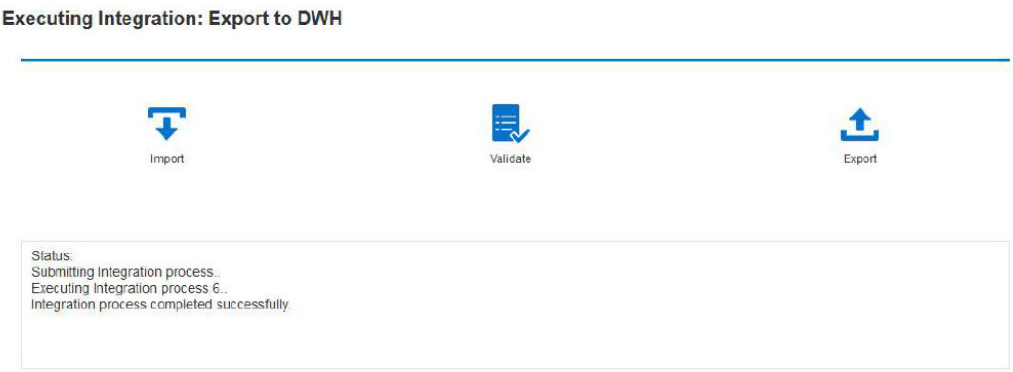
22. Create a new query with the following values:
    a. Remap BATCH_ID to BATCH_NUMBER.
    b. Use the prefix **Acc**, for example, ACCOUNT.
    c. Use the prefix **Org**, for example, ORGANISATION.
    d. Concatenate Period and Year into PERIOD_NAME.
    e. Use the prefix **Prd**, for example, PRODUCT.
    f. Round DATA_AMOUNT to 2 decimal places.



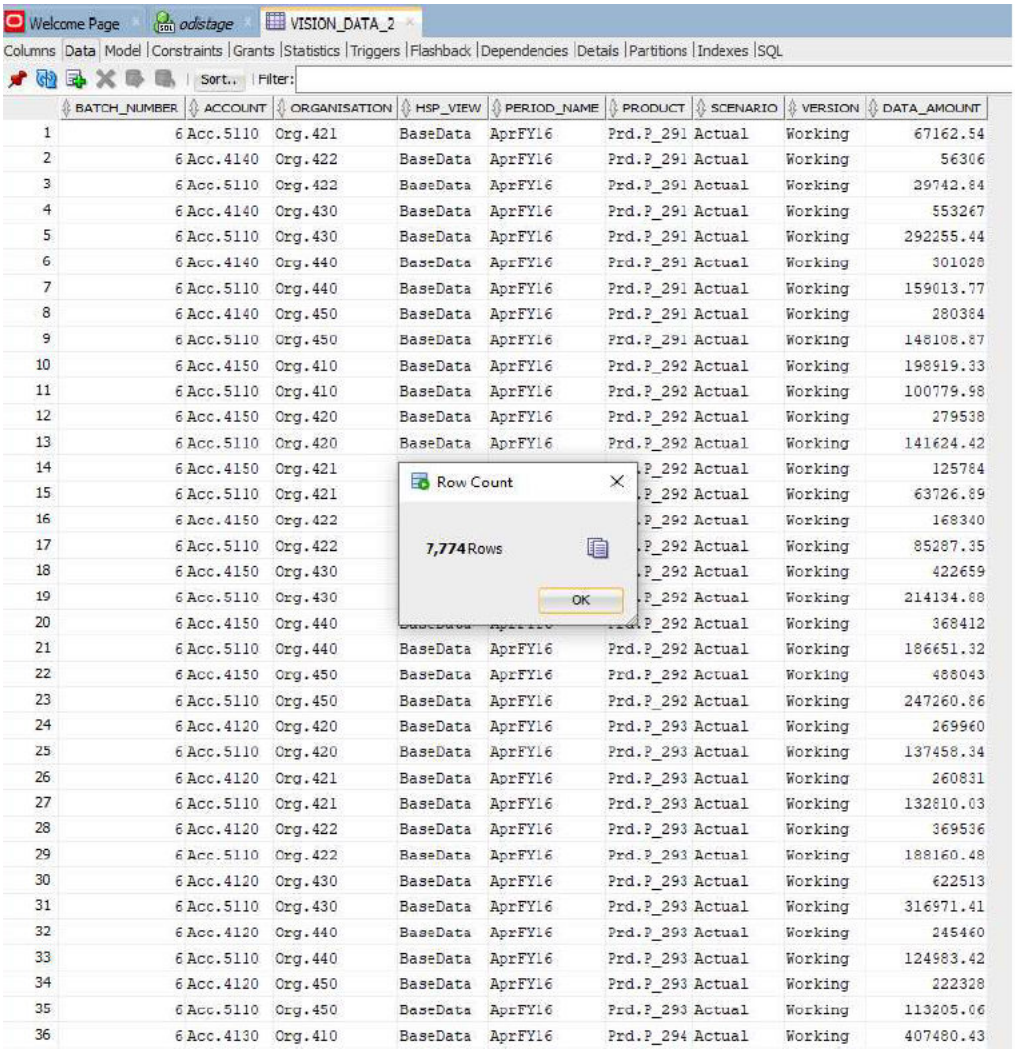
23. In **Application Details**, specify the query name in the registered Data Export application and leave the **Table Name** blank.

24. Run the integration process again for a period range and wait till the process is successful.

Copyright © 2017, 2026, Oracle and/or its affiliates.



25. Validate the destination table has been populated correctly.



	BATCH_NUMBER	ACCOUNT	ORGANISATION	HSP_VIEW	PERIOD_NAME	PRODUCT	SCENARIO	VERSION	DATA_AMOUNT
1	6	Acc.5110	Org.421	BaseData	AprFY16	Prd.P_291	Actual	Working	67162.54
2	6	Acc.4140	Org.422	BaseData	AprFY16	Prd.P_291	Actual	Working	56306
3	6	Acc.5110	Org.422	BaseData	AprFY16	Prd.P_291	Actual	Working	29742.84
4	6	Acc.4140	Org.430	BaseData	AprFY16	Prd.P_291	Actual	Working	553267
5	6	Acc.5110	Org.430	BaseData	AprFY16	Prd.P_291	Actual	Working	292255.44
6	6	Acc.4140	Org.440	BaseData	AprFY16	Prd.P_291	Actual	Working	301020
7	6	Acc.5110	Org.440	BaseData	AprFY16	Prd.P_291	Actual	Working	159013.77
8	6	Acc.4140	Org.450	BaseData	AprFY16	Prd.P_291	Actual	Working	280384
9	6	Acc.5110	Org.450	BaseData	AprFY16	Prd.P_291	Actual	Working	148108.87
10	6	Acc.4150	Org.410	BaseData	AprFY16	Prd.P_292	Actual	Working	198919.33
11	6	Acc.5110	Org.410	BaseData	AprFY16	Prd.P_292	Actual	Working	100779.98
12	6	Acc.4150	Org.420	BaseData	AprFY16	Prd.P_292	Actual	Working	279538
13	6	Acc.5110	Org.420	BaseData	AprFY16	Prd.P_292	Actual	Working	141624.42
14	6	Acc.4150	Org.421			P_292	Actual	Working	125784
15	6	Acc.5110	Org.421			P_292	Actual	Working	63726.89
16	6	Acc.4150	Org.422			P_292	Actual	Working	168340
17	6	Acc.5110	Org.422			P_292	Actual	Working	85287.35
18	6	Acc.4150	Org.430			P_292	Actual	Working	422659
19	6	Acc.5110	Org.430			P_292	Actual	Working	214134.88
20	6	Acc.4150	Org.440			P_292	Actual	Working	368412
21	6	Acc.5110	Org.440	BaseData	AprFY16	Prd.P_292	Actual	Working	186651.32
22	6	Acc.4150	Org.450	BaseData	AprFY16	Prd.P_292	Actual	Working	488043
23	6	Acc.5110	Org.450	BaseData	AprFY16	Prd.P_292	Actual	Working	247260.86
24	6	Acc.4120	Org.420	BaseData	AprFY16	Prd.P_293	Actual	Working	269960
25	6	Acc.5110	Org.420	BaseData	AprFY16	Prd.P_293	Actual	Working	137458.34
26	6	Acc.4120	Org.421	BaseData	AprFY16	Prd.P_293	Actual	Working	260831
27	6	Acc.5110	Org.421	BaseData	AprFY16	Prd.P_293	Actual	Working	132810.03
28	6	Acc.4120	Org.422	BaseData	AprFY16	Prd.P_293	Actual	Working	369536
29	6	Acc.5110	Org.422	BaseData	AprFY16	Prd.P_293	Actual	Working	188160.48
30	6	Acc.4120	Org.430	BaseData	AprFY16	Prd.P_293	Actual	Working	622513
31	6	Acc.5110	Org.430	BaseData	AprFY16	Prd.P_293	Actual	Working	316971.41
32	6	Acc.4120	Org.440	BaseData	AprFY16	Prd.P_293	Actual	Working	245460
33	6	Acc.5110	Org.440	BaseData	AprFY16	Prd.P_293	Actual	Working	124983.42
34	6	Acc.4120	Org.450	BaseData	AprFY16	Prd.P_293	Actual	Working	222328
35	6	Acc.5110	Org.450	BaseData	AprFY16	Prd.P_293	Actual	Working	113205.06
36	6	Acc.4130	Org.410	BaseData	AprFY16	Prd.P_294	Actual	Working	407480.43

EPM Integration Agent Write-Back Event Scripts

Describes how to use the `BefExport` and `AftExport` event to execute custom code, by-pass the standard processing, and generate a custom data set.

Copyright © 2017, 2026, Oracle and/or its affiliates.

The EPM Integration Agent executes two events during the write-back executions: `BefExport` and `AftExport`. Use these events to execute custom code, by-pass the standard processing, and generate a custom data set, which is included in the data file made available to the on-premise database.

Custom API Parameters passed for the `BefExport` and `AftExport` events.

Name	Description
JOBID	Job Id of the write-back execution
JOBTYPE	Type of the job "WRITEBACK"
EXPORT_DATA_FILE	Name of the export data file in the full path

Using a BefExport Event

When writing back using the EPM Integration Agent, use the `BefExport` event to perform any action before inserting data to the table or you can override the default insert processing.

The following script example shows how to call an external API that executes before the write-back.

The `BefExport` event in this example:

- prints the contents of the `agentContextParams` map
- fetches and prints the table name into which the data is inserted.
- fetches and prints the insert query
- prints an info message to the agent process log. This entry is logged to the process log in `EPM_APP_DATA_HOME\logs` and `epmagent.log`.

For information about EPM Integration Agent Context functions, see EPM Integration Agent Context Functions.

Note that this script is provided only as an example, and it is not warranted against defects, and users may not file a service request with Oracle support in regard to any questions or issues related to the script.

```
import sys
import java

'''
Before export custom script. This script will be called before the writeback
begins execution.
'''
#print Begin: BefExport.py

#print 'Event Type is: ' + event

'''
Print the contents of the agentContextParams map which is an unmodifiable
map.
'''
#print 'JOBTYPE: ' + agentContext["JOBTYPE"]
#print 'EPM_APP_DATA_HOME: ' + agentContext["EPM_APP_DATA_HOME"]
#print 'WRITEBACK_DATA_FILE: ' + agentContext["WRITEBACK_DATA_FILE"]
#print 'JOBID: ' + str(agentContext["JOBID"])
```


Copyright © 2017, 2026, Oracle and/or its affiliates.

```
#print 'INTEGRATION: ' + agentContext["INTEGRATION"]
#print 'LOCATION: ' + agentContext["LOCATION"]
#print 'SOURCE_APPLICATION: ' + agentContext["SOURCE_APPLICATION"]
#print 'TARGET_APPLICATION: ' + agentContext["TARGET_APPLICATION"]



'''
getTable() Method to fetch the table name into which the data will be
inserted. This is
passed from cloud to the agent during the writeback execution call.
'''
#print "Printing Table Name: " + agentAPI.getTable()


'''
getInsertQuery() Method to fetch the insert query. This is the query which is
passed from cloud to the agent during the writeback execution call.
'''
#print "Printing Query: " + agentAPI.getInsertQuery()


'''
Log an info message to the agent process log. This entry will be logged only
to the process log in EPM_APP_DATA_HOME\logs
folder and not to epmagent.log. The log entry will be created at INFO log
level.
'''
#agentAPI.logInfo("SAMPLE: INFO log message from script")


'''
Log an severe message to the agent process log. This entry will be logged
into the process log in EPM_APP_DATA_HOME\logs
folder and also into epmagent.log. The log entry will be created at SEVERE
log level.
'''
#agentAPI.logError("SAMPLE: SEVERE log message from script")


'''
Uncomment to skip the export data execution. The writeback execution can be
skipped only during the
BEFORE_EXPORT event. This will skip the execution of the insert statements.
The cloud process will be marked as failed in the Export data step.
'''
#agentAPI.skipAction('true')


'''
Return false in case of error, which will throw an exception in the agent.
'''

#print "End: BefExport.py"
```


Copyright © 2017, 2026, Oracle and/or its affiliates.

Additional Example Showing the use of a BefExtract Event Script

The following event script shows how to make a database call to clear data from the target table, and then use `sqlldr` to populate the target table with a dataset extracted from EPM. This example shows how to populate a cloud database with `sqlldr`, but the same technique may be used to populate other third party non-Oracle databases. Like the prior example, this script is provided as an example and is not warranted against defects, and users may not file a service request with Oracle support in regards to any questions or issues related to the script.

The default write-back process uses individual insert statements to post data to the target database, and if the write-back dataset is large, it is recommended to use the specific database utility to load bulk data rather than by using the default process. In this example, the script shows how to call the Oracle `sqlldr` utility, and it is assumed that the utility has been downloaded from OTN and is installed and accessible to the event script.

```
#--------------------------------------------------------------------------
------------#
# befExport.py
# This script is used to perform database operations as part of the writeback
process
# The script also uses sqlldr, and this must be installed prior to executing
the script
# References to sqlldr and the control file will be different from this
script and should
# be updated to reference the location where sqlldr is installed in your
environment.
#----------------------------------------#
# Housekeeping and startup               #
#----------------------------------------#
from java.sql import DriverManager, SQLException
import subprocess

def main():
  #------------------------------------------------#
  # Print integration context details to the log file #
  #------------------------------------------------#
  agentAPI.logInfo("#--------------------------------#")
  agentAPI.logInfo("# Delete data in table MC_WB_TEST ")
  agentAPI.logInfo("# Location:    " + agentContext["LOCATION"])
  agentAPI.logInfo("# Integration: " + agentContext["INTEGRATION"])
  agentAPI.logInfo("# Data File:   " + agentContext["WRITEBACK_DATA_FILE"])
  agentAPI.logInfo("#--------------------------------#")

  #---------------------------------------#
  # Retrieve user, password, and JDBC URL #
  #---------------------------------------#
  cred = agentAPI.getConnectionDetails()
  url = cred.getJDBCUrl()
  user = cred.getUserName()
  password = cred.getPassword()
  agentAPI.logInfo("# Connection Details")
  agentAPI.logInfo("# URL:  " + url)
  agentAPI.logInfo("# User: " + user)

  #--------------------------------------------------------------------------
```


Copyright © 2017, 2026, Oracle and/or its affiliates.

```
#
  # Open connection to the database and execute SQL
#
  # This step deletes existing data before loading new data
#
  # For large tables it is recommended to truncate and the recreate the table
#
  #-------------------------------------------------------------------------
#
  cnx = DriverManager.getConnection(url, user, password)
  agentAPI.logInfo("# Successfully connected to Oracle Cloud DB using
DriverManager")
  stmt = cnx.createStatement()
  stmt.executeQuery('Delete from MC_WB_TEST')
  agentAPI.logInfo("# Deleted all rows from MC_WB_TEST #")
  agentAPI.logInfo("#---------------------------------#")

  #-----------------------------------------------#
  # Define the SQL*Loader command and its arguments #
  #-----------------------------------------------#
  agentAPI.logInfo("# Starting SQLLDR")

  #---------------------------------------#
  # Replace \ with \\ in the filename path #
  #---------------------------------------#
  data_file = agentContext["WRITEBACK_DATA_FILE"]
  new_file = data_file.replace("\\", "\\\\")

  #-----------------------------------------------#
  # Specify control file and include \\ in path #
  #-----------------------------------------------#
  ctr_file = "C:\\EPMAgent\\bin\\MyData\\scripts\\instantclient\\load.ctl"

  #-----------------------------------------------------------#
  # Specify user, password and tnsnames.ora entry           #
  # The format is user/password@<SID from tnsnames.ora>     #
  # for cloud databases.                                    #
  #-----------------------------------------------------------#
  user = user + "/" + password + "@mcebs19c_medium"

  #------------------------#
  # Specify path to sqlldr #
  #------------------------#
  sqlldr_path = "C:\\EPMAgent\\bin\\MyData\\scripts\\instantclient\\sqlldr"

  #----------------------#
  # Build sqlldr command #
  #----------------------#
  sqlldr_command = [
    sqlldr_path,
    "userid=" + user,
    "control=" + ctr_file,
    "data=" + new_file
  ]

  #---------------------------#
```


Copyright © 2017, 2026, Oracle and/or its affiliates.

```
  # Run the SQL*Loader command #
  #----------------------------#
  with open("sqlldr_output.txt", "w") as fout:
    exit_code = subprocess.call(sqlldr_command, stdout=fout, shell=True)
  agentAPI.logInfo("# exit code: " + str(exit_code))
  agentAPI.logInfo("#-------------------------------#")

  #-------------------------------------------------------------#
  # Skip additional agent sql processing                        #
  # This ensures that the default insert processing is bypassed #
  #-------------------------------------------------------------#
  agentAPI.skipAction('true')

  #---------------------------------------------#
  # Raise an error if needed as part of         #
  # the process.  This will halt the processing  #
  # of the write-back process                   #
  #---------------------------------------------#
  #agentAPI.logInfo("# Manually raising a ValueError to stop the process")
  #raise ValueError("This is a manually raised ValueError")

  return

#-------------------------------------------------------------#
# Check for location and process if needed                    #
# The same script is used for data load and write-back,       #
# so make sure that each specific case is correctly captured #
#-------------------------------------------------------------#
#
if agentContext["LOCATION"] == "1WB_TEST":
  main()
else:
  agentAPI.logInfo("#--------------------------------------#")
  agentAPI.logInfo("# Location: " + str(agentContext["LOCATION"]))
  agentAPI.logInfo("# Script not required, exiting now...")
  agentAPI.logInfo("#--------------------------------------#")
```

Using a AftExport Event

When writing back using the EPM Integration Agent, use the `AftExport` to do any post processing cleanup after inserting data to the table.

The following script example shows how to call an external API that executes after the write-back.

The `AftExport` event in this example:

- prints the contents of the `agentContextParams` map.
- prints an info message to the agent process log. This entry is logged to the process log in `EPM_APP_DATA_HOME\logs`.
- prints any severe error message to the agent process log. This entry is logged to the process log in `EPM_APP_DATA_HOME\logs`.

For information about EPM Integration Agent Context functions, see EPM Integration Agent Context Functions.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Note that this script is provided only as an example, and it is not warranted against defects, and users may not file a service request with Oracle support in regard to any questions or issues related to the script.

```
import sys

'''
After export custom script. This script will be called after the writeback
finishes execution.
'''
#print "Begin: AftExport.py"

#print 'Event Type is: ' + event

'''
Print the contents of the agentContextParams map which is an unmodifiable
map.
'''
#print 'JOBTYPE: ' + agentContext["JOBTYPE"]
#print 'EPM_APP_DATA_HOME: ' + agentContext["EPM_APP_DATA_HOME"]
#print 'WRITEBACK_DATA_FILE: ' + agentContext["WRITEBACK_DATA_FILE"]
#print 'JOBID: ' + str(agentContext["JOBID"])
#print 'INTEGRATION: ' + agentContext["INTEGRATION"]
#print 'LOCATION: ' + agentContext["LOCATION"]
#print 'SOURCE_APPLICATION: ' + agentContext["SOURCE_APPLICATION"]
#print 'TARGET_APPLICATION: ' + agentContext["TARGET_APPLICATION"]



'''
Log an info message to the agent process log. This entry will be logged only
to the process log in EPM_APP_DATA_HOME\logs
folder and not to epmagent.log. The log entry will be created at INFO log
level.
'''
#agentAPI.logInfo("SAMPLE: INFO log message from script")


'''
Log an severe message to the agent process log. This entry will be logged
into the process log in EPM_APP_DATA_HOME\logs
folder and also into epmagent.log. The log entry will be created at SEVERE
log level.
'''
#agentAPI.logError("SAMPLE: SEVERE log message from script")


'''
Return false in case of error, which will throw an exception in the agent.
'''
returnValue = 'true'

#print "End: AftExport.py"
```


Copyright © 2017, 2026, Oracle and/or its affiliates.

Understanding Agent Clusters

Describes how to use clusters to distribute integration jobs.

You can use clusters to distribute integration jobs.

The following are examples of some of the business scenarios where you may want to define multiple clusters:

- The source systems are geographically distributed. In order to avoid network latency, you may want to deploy one or more agents in each data center location.
- Different business units in your organization want to secure and manage their jobs independently.
- Your organization uses different types of source systems. For example, E-Business Suite (EBS) and Peoplesoft are located in different data centers. The security and data volume may be different and you want to manage the data extraction process independently.
- The data load frequency for various sources are different. You may have a data warehouse staging system used as a source for reporting and drill-down. Data may be loaded daily in an interactive mode and you want to support drill down. Another ERP system may be used as a source for the month-end consolidation process only and all data loads are batch processed. You can define a different cluster for each of these data sources.

You can allocate the integration to the different clusters by defining cluster assignments. You can assign integrations to the appropriate cluster by entity type:

- Integration (data rule)
- Location
- Target application

When an integration job is initiated, the system checks where a job has been assigned, determines the cluster, and assigns the job to that cluster. The order of precedence for executing a job is determined by the entity type. Integration takes precedence over location, which takes precedence over a target application.

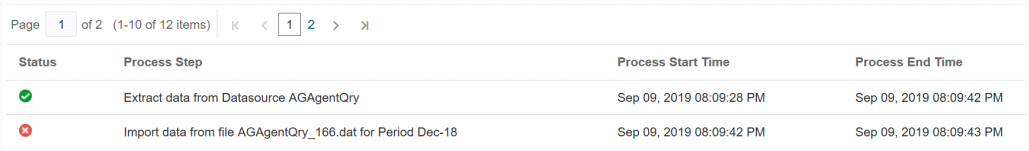
Within each cluster you can have multiple agents to provide further load balancing and high availability. You associate an agent to a cluster in the agent configuration INI file. When you start the EPM Integration Agent, it automatically associates the agent with a cluster and executes the jobs assigned to the cluster. The procedure for load balancing within a cluster depends whether the execution mode is synchronous or asynchronous.

In synchronous mode, the system uses a round robin process to assign the jobs to the agents that belong to a cluster. For more information, see Configuring Synchronous Mode.

In asynchronous mode, you set up the agent to start at different times to achieve high availability. For example, you can set the interval as 10 minutes and start another agent 5 minutes after the hour, and then another agent 10 minutes after the hour. In effect, you have an agent checking every 5 minutes.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> When a selected entity (location, application, or integration) is reassigned to another cluster, the previous cluster assignment is deleted and replaced with the new assignment automatically.
> Extracting an integration job does not affect the import and validation status of the job. It is possible to have a successful extract, but a failed import and validation as shown below:

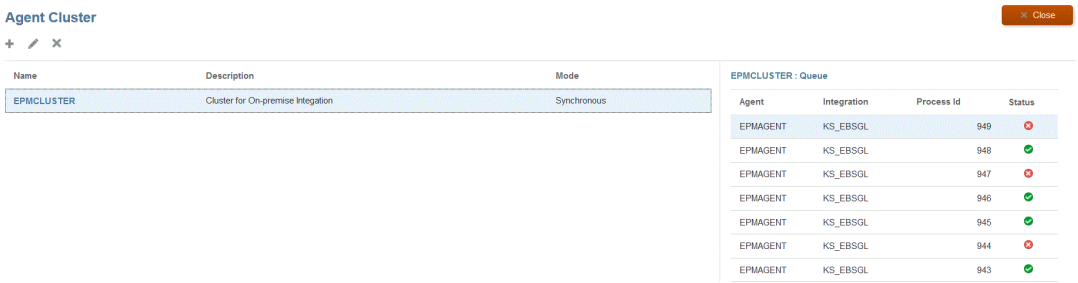


Adding a Cluster

Describes how to add a cluster.

To add a cluster:

1. From the Home page, click **Application** ( ).
2. Click **Data Exchange** ( ), and then select the **Data Integration** tab.
3. Optionally, you can launch Data Integration by clicking **Navigator** ( ), and then from **Application**, selecting **Data Exchange** (Data Exchange).
4. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Configure** actions drop-down, select **Agent**.



5. From the **Agent Cluster** page, click **Add**.
6. On the **Create Agent Cluster** page, specify the cluster name in the **Cluster Name** field.

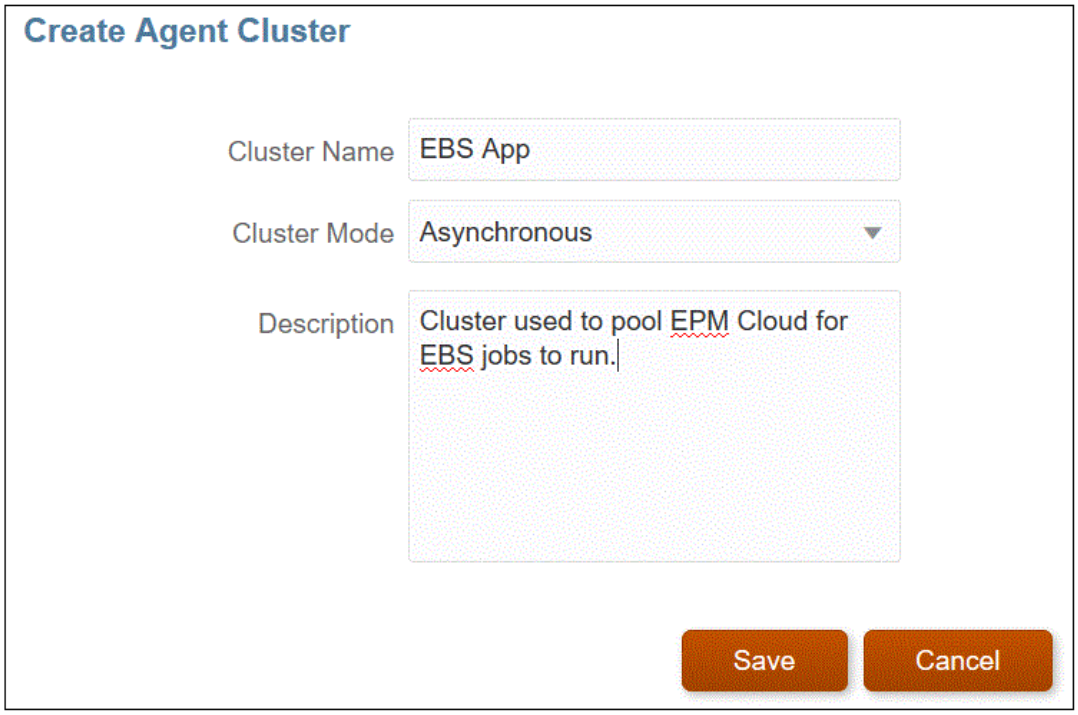
   Only alphanumeric characters can be used in the name. Do not use special characters, such as sign (@) or ampersand (&). The name cannot be modified once the cluster has been created.
7. In **Cluster Mode**, select the integration flow.

   Available modes include the following:

   - Synchronous
   - Asynchronous
8. In **Description**, specify any additional information about the cluster.

Copyright © 2017, 2026, Oracle and/or its affiliates.

9. Click **Save**.



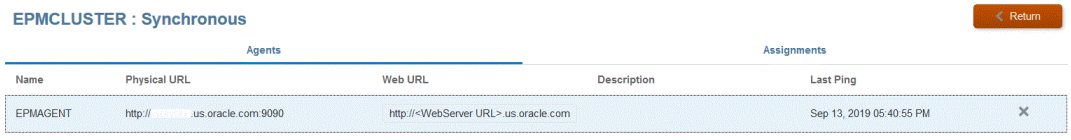
Assigning Integration Jobs

Describes how to add an assignment to a cluster.

Assignments let you prioritize when integration jobs are extracted in the cluster by the agent running the extraction. You assign entities (location, application, or integration) to associate them with a selected cluster and to set up their order of extraction.

To assign an integration job:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Configure** actions drop-down, select **Agent**.
2. From the **Agent Cluster** page, select the name of the cluster to which to add an integration job.
3. Click the **Agents** tab, and select the name of the agent from the cluster to use with the assignment.



Note the following:

- Name—The Name of agent assigned to the cluster.
- Physical URL—represents the IP address and port to which the Web URL redirects the request through the reverse proxy.
- Web URL—represents the web address to which the Oracle Fusion Cloud Enterprise Performance Management sends requests.

  The Web URL field can be modified.

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Last Ping—Date and time the system last checked when the host was available.

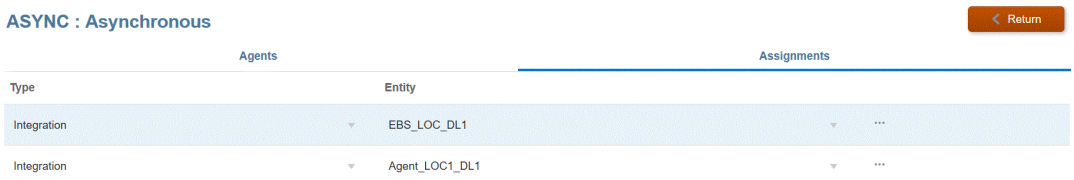
4. Click the **Assignments** tab, and then from the **Type** drop-down, select the type of entity.

   Valid entity types include the following:

   - Application
   - Integration
   - Location

5. From the **Entity** drop-down, select the entity.

6. **Optional**: Click the **Add/Delete** button ( ) to add a new assignment or delete an existing assignment.



EPM Integration Agent Scripting

Explains the standard functionality that allows the user to connect to any data source through the use of Java, or Jython/Groovy scripting, or to conditionally change the defined query based on business requirements.

The EPM Integration Agent provides extensions to the standard functionality that allows the user to connect to any data source through the use of Java, or Jython/Groovy scripting, or to conditionally change the defined query based on business requirements. The standard SQL processing with the EPM Integration Agent produces a data set that is uploaded to the Oracle Fusion Cloud Enterprise Performance Management, and with scripting, you can by-pass the standard processing, and generate a custom data set, which is uploaded to the Cloud EPM as part of the EPM Integration Agent processing.

Events

The steps performed by the EPM Integration Agent are as follows:

1. Process SQL defined by the EPM Integration Agent data source specified in the Oracle Fusion Cloud Enterprise Performance Management.
2. Prepare and transmit the SQL result set to the Cloud EPM from the defined data source.

These steps are referred to as the "Extract" and "Upload" steps. Scripting in the EPM Integration Agent supports four events related to these two process steps where you can define custom scripts. The events are as follows:

Event	Script Name	Description
`Before Extract`	`BefExtract.py` `BefExtract.groovy`	Script executed before the agent extract processing. If you want to perform any processing prior to the SQL processing, that code should be included in this script.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Event	Script Name	Description
After Extract	AftExtract.py AftExtract.groovy	Script executed after the agent extract processing. After the extract a file is prepared, which includes the job id with the dat suffix in the agent/MyData/data local folder
Before Upload	BefUpload.py BefUpload.groovy	Script executed before the data file is uploaded to the Cloud EPM. The file uploaded to the Cloud EPM is the <jobID>.dat file from the agent/MyData/data folder.
After Upload	AftUpload.py AftUpload.groovy	Script executed after the data file is uploaded to the Cloud EPM.

> **Note**
>
> For information on the EPM Integration Agent BefExport and AftExport events used during write-back executions, see EPM Integration Agent Write-Back Event Scripts.

Sample scripts are available in the agent/Sample/jython and agent/Sample/groovy folders on the local machine where the agent was installed. The system runs an instance of a script if it is placed in the agent/MyData/scripts folder. For example, if you want to only execute the BefExtract script, only save this script to the agent/MyData/scripts folder.

You may also choose to create your own implementation of the Java class that implements the four custom methods described above. An example is provided in the Agent/Sample folder for reference, but consider the following if this approach is selected:

- If you choose to implement your integration logic directly in Java, then you are not required to save any scripts to the agent/MyData/scripts folder. The Java implementation always executes the four events, and if a script file is not present for any of those events, it just processes the Java logic define by the customer/partner.
- You can create a Java class file CustomEvent.java, which should implement the interface EPMAgentInterface (for example, public class CustomEvent implements EPMAgentInterface{), present in the agent-interface.jar.
- The CustomEvent.class can be packaged inside the agent-interface.jar as oracle.epm.aif.agent.agentinterface.CustomEvent.class. If you choose to package it in a separate jar other than agent-interface.jar, then this jar must be included in the CUSTOM_CLASS_PATH agent startup parameter.
- If you choose to use a different class name from oracle.epm.aif.agent.agentinterface.CustomEvent.class, then you must set the startup parameter CUSTOM_INTERFACE_CLASS_NAME with the fully qualified java class name.
- Note that the agent-interface.jar is updated each time development provides a fix or a new version during the periodic Cloud EPM release cycle. The custom event class may be used for development and testing purposes, but for production deployment, it is recommended you maintain a separate jar so that customizations are not overwritten.

EPM Integration Agent API Methods

The EPM Integration Agent provides a number of methods that may be used in scripts to perform actions such as the specification of text in the log file, updates to bind variables and

Copyright © 2017, 2026, Oracle and/or its affiliates.

queries, and actions that instruct the EPM Integration Agent to skip the SQL processing step for the case where a special query is used, or a non-SQL data source is used.

The EPM Integration Agent API methods are as follows:

API Method	Description	Example
`logInfo()`	Log an info message to the agent process log. This entry is logged to the process log in the `EPM_APP_DATA_HOME\logs` folder, the job log in the Oracle Fusion Cloud Enterprise Performance Management, but not to the local epmagent.log. The log entry is created at the INFO log level.	`agentAPI.logInfo("SAMPLE: INFO log message from script")`
`logError()`	Log an error message to the agent process log. This entry is logged to the process log in `EPM_APP_DATA_HOME\logs` folder, the job log in the Cloud EPM, and also into epmagent.log. The log entry is created at the SEVER log level.	`agentAPI.logError("SAMPLE: SEVER log message from script")`

`setBindVariables()`	Use the `setBindVariables()` method to update the extract query's bind variables. This is applicable only in the `befExtract` script.
	The bind variables must be passed as a java map entry for each variable with variable `NAME` as key and `VALUE`.
		`newBindVar = dict({'PERIOD':'Feb-05', 'LEDGER':'Vision Operations (USA)'})`
		`jmap = java.util.HashMap()`
		`for key in newBindVar.keys(): jmap[key] = newBindVar[key]`
		`agentAPI.setBindVariables(jmap)`

`getBindVariables()`	Use the `getBindVariables()` method to fetch the bind variables for the extract query. Each bind variable is stored in a map, which uses the keys `NAME` and `VALUE` to define the bind variable.	`bindVariables = agentAPI.getBindVariables()` `for entry in bindVariables.entrySet(): print entry.key, entry.value`
`updateQuery()`	Use the `updateQuery()`method to update the extract query. This is only applicable in the `befExtract` script.	`agentAPI.updateQuery("SELECT * FROM TDATASEG")`
`getQuery()`	Use the `getQuery()` method to fetch the query that is passed from Cloud EPM to the EPM Integration Agent during the extract execution call.	`print "Printing Query: " + agentAPI.getQuery()`

Copyright © 2017, 2026, Oracle and/or its affiliates.

API Method	Description	Example
`skipAction()`	Use the `skipAction()` method to skip the extract data step, when a custom extract routine is used by the EPM Integration Agent. Note that the extract execution can be skipped only during the `befExtract` script. If you want to provide a data file to upload to the Cloud EPM rather than executing the saved query, a file with the name `<process ID>.dat` must be saved to the `MyData/data` folder in time to be uploaded to the Cloud EPM. This means that the file must be saved to this folder in the `befExtract`, `aftExtract`, or `befUpload` scripts.	`agentAPI.skipAction('true')`

`getConnectionDetails()`	Use the `getConnectionDetails()` method to fetch the connection Credential data object from the cloud that is passed from the Cloud EPM to the EPM Integration Agent during the extract execution call.	`cred = agentAPI.getConnectionDetails()`
		`url = cred.getJDBCUrl()`
		`user = cred.getUserName()`
		`password = cred.getPassword()`
		`agentAPI.logInfo ("Connection Details: ")`
		`agentAPI.logInfo("URL: " + url)`
		`agentAPI.logInfo('User: ' + user)`

`setCustomConnection(conn)`	Use the `setCustomConnection(conn)` method to set the custom created connection into the EPM Integration Agent during the extract execution call. The Agent uses this connection to execute. The conn object must be a type 4 JDBC connection and an implementation of `java.sql.Connection` interface.
	This is only applicable in the `befExtract` and `befExport` scripts. See sample scripts `BefExtract.py` and the java `CustomEventSample` and `CustomWribackEvent` for more details on usage.
		To create a custom connection to an Oracle database:
		`java.lang.Class.forName("oracle.jdbc.driver.OracleDriver")`
		`conn = java.sql.DriverManager.getConnection('jdbc:oracle:thin:@<host>:<port>:<sid>', '<user/schema name>', '<password>')`
		`agentAPI.setCustomConnection(conn)`

Integrating Oracle Autonomous Database Data Using the EPM Integration Agent

Describes how to integrate data directly between the Oracle Autonomous Database and the Oracle Fusion Cloud Enterprise Performance Management using the EPM Integration Agent.

Copyright © 2017, 2026, Oracle and/or its affiliates.

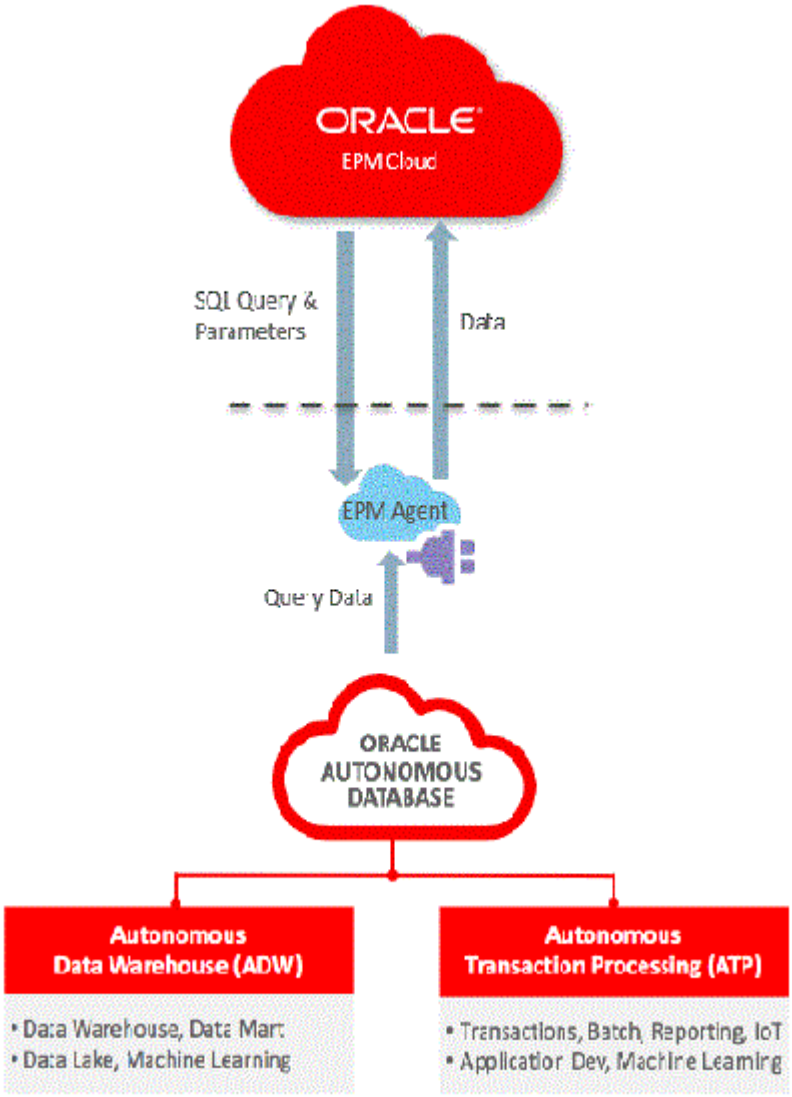
You can integrate data directly to and from the Oracle Autonomous Database and the Cloud EPM using the EPM Integration Agent. This type of integration enables you to extract source data from staging or other applications running on the Oracle Autonomous Database. You can also export data from the Cloud EPM to selected reporting data warehouses on the Oracle Autonomous Database.

The Oracle Autonomous Database is a comprehensive cloud experience with fully automated data warehousing and transaction processing workloads optimized on high-end Oracle hardware systems. The Oracle Autonomous Database provides an easy-to-use, fully autonomous database that scales elastically and delivers fast query performance.

As a service, the Oracle Autonomous Database does not require database administration. You do not need to configure or manage any hardware or install any software. The Oracle Autonomous Database handles provisioning the database, backing up the database, patching and upgrading the database, and growing or shrinking the database. For more information about the Oracle Autonomous Database, see FAQs For Autonomous Database.

When loading or writing back data to and from the Oracle Autonomous Database, the EPM Integration Agent is used to establish the connectivity between the Oracle Autonomous Database and the Cloud EPM. Customers can either install the agent on a local network (similar to the way customers currently use the agent to extract data from on-premises data sources) or install the agent in a Oracle Cloud Infrastructure (OCI) Compute instance and then configure the database connectivity. The Compute instance provides the processing power and memory capacity for the virtual servers they host. This approach provides customers with scalability and flexibility for their solutions.

Copyright © 2017, 2026, Oracle and/or its affiliates.



The Cloud EPM and Oracle Autonomous Cloud integration also enables customers to drill down on data and to perform loads using the Quick Mode method.

Process Description for Connecting to the Oracle Autonomous Database Using EPM Integration Agent

The EPM Integration Agent provides an out of the box solution that enables customers to perform two way integrations between the Oracle Fusion Cloud Enterprise Performance Management and the Oracle Autonomous Database.

At a high level, here are the steps for integrating data between the Cloud EPM and the Oracle Autonomous Database:

1. Connect to the Oracle Cloud Infrastructure Autonomous Database.
2. Install the EPM Integration Agent on the server for the local network or Oracle Cloud Infrastructure Compute instance.

Copyright © 2017, 2026, Oracle and/or its affiliates.

3. Download the wallet from the Oracle Cloud Infrastructure.
4. Copy the wallet to the agent server and then unzip it.
5. Configure the **Oracle Autonomous Database - Agent** data source application and/or configure Data Export application.
6. Set up the integration.
7. Run the integration.

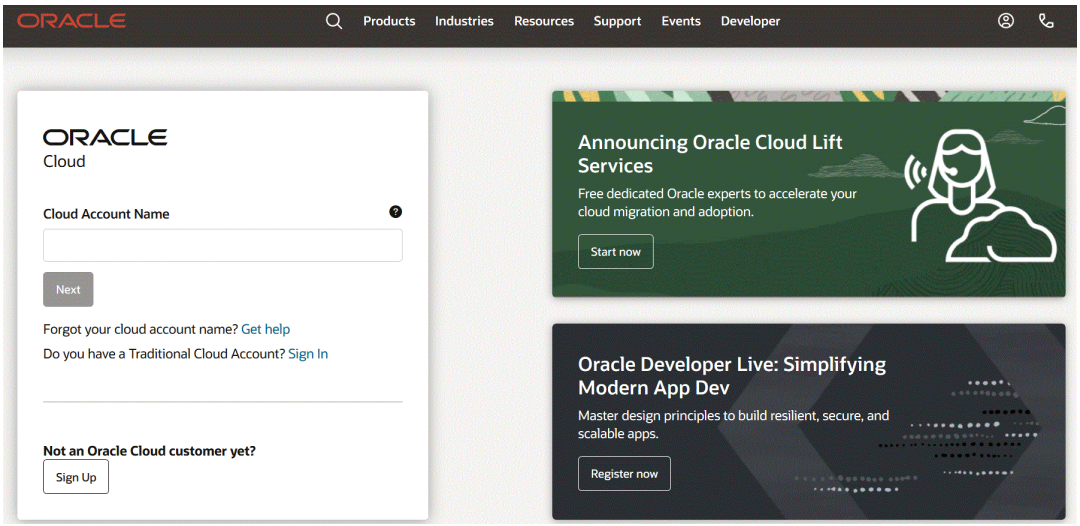
Connecting to the Oracle Cloud Infrastructure

Describes how to connect to the Oracle Cloud Infrastructure provided you already have an autonomous database instance.

The following steps assume that you are connecting to an autonomous database instance that has been created. For more information about creating an autonomous database instance, see Getting Started and Provisioning Autonomous Transaction Processing.

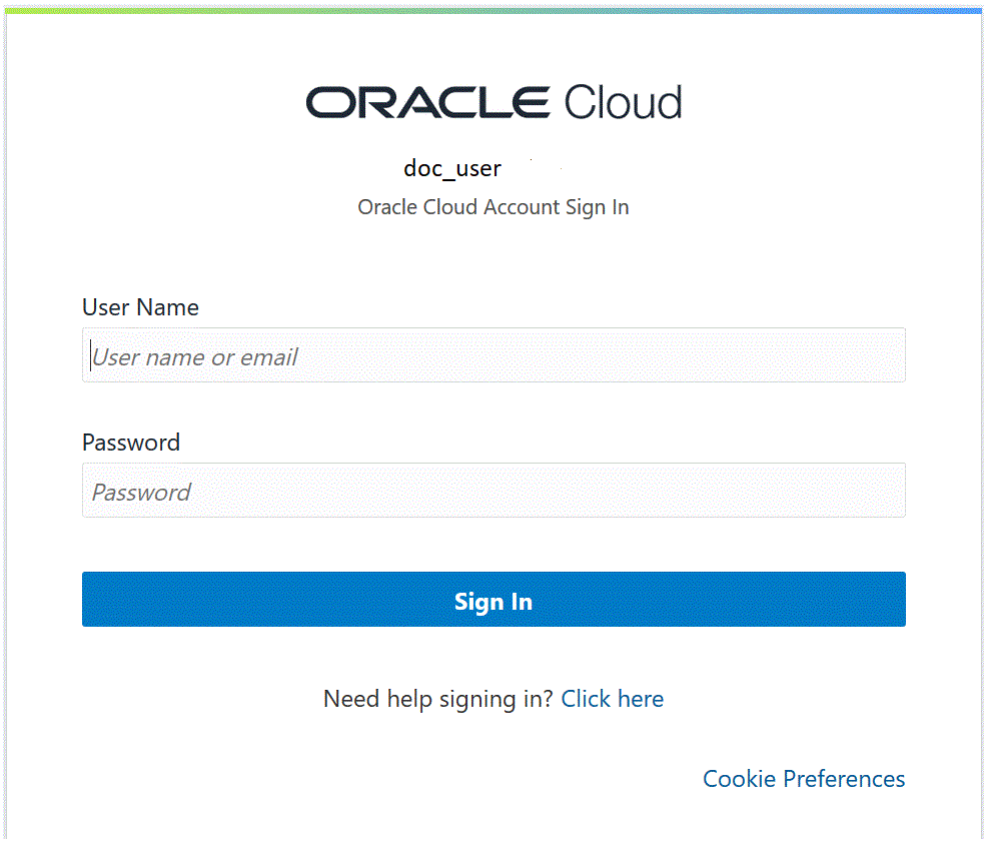
To connect to an Oracle Autonomous Database:

1. Sign in from **https://cloud.oracle.com**.
2. In **Cloud Account Name**, enter your cloud account name and click **Next**.

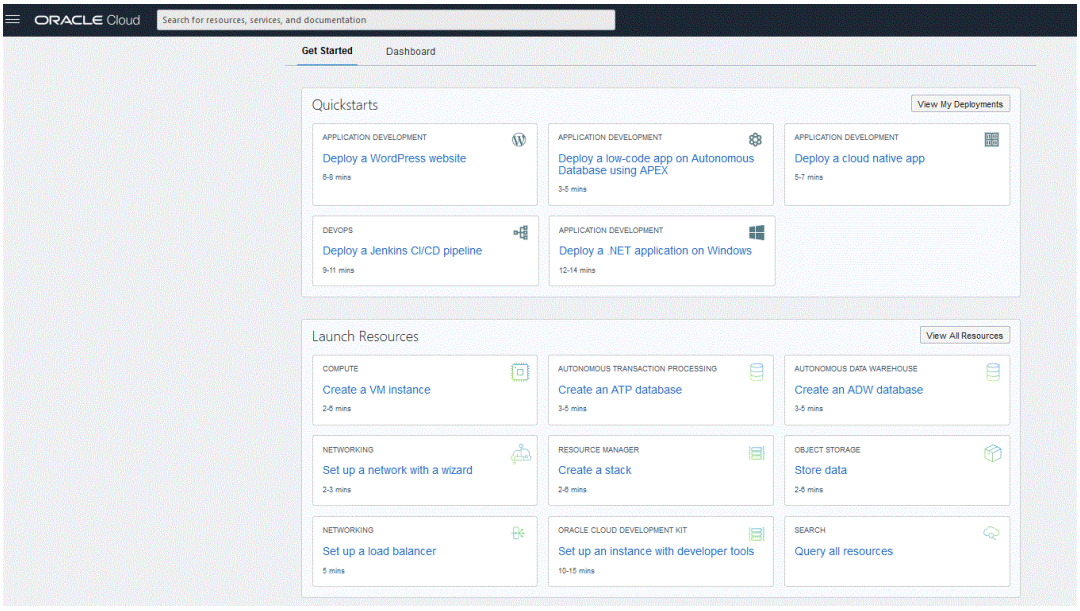


3. On the **Oracle Cloud Account Sign In** page, enter your user name in **User Name** and password in **Password** and then click **Sign In**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

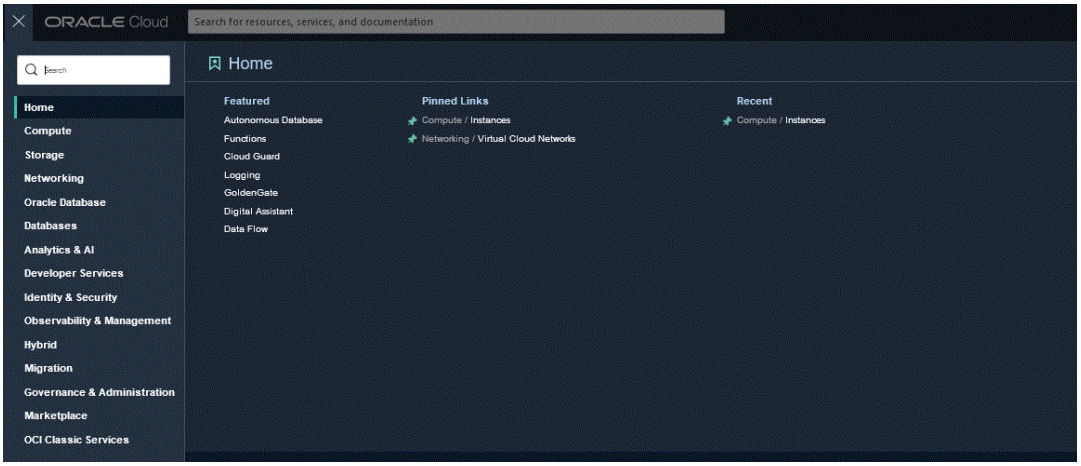


4. On the **Getting Started** page, click **Navigator** ( ) in the upper left to show the top-level navigation choices.



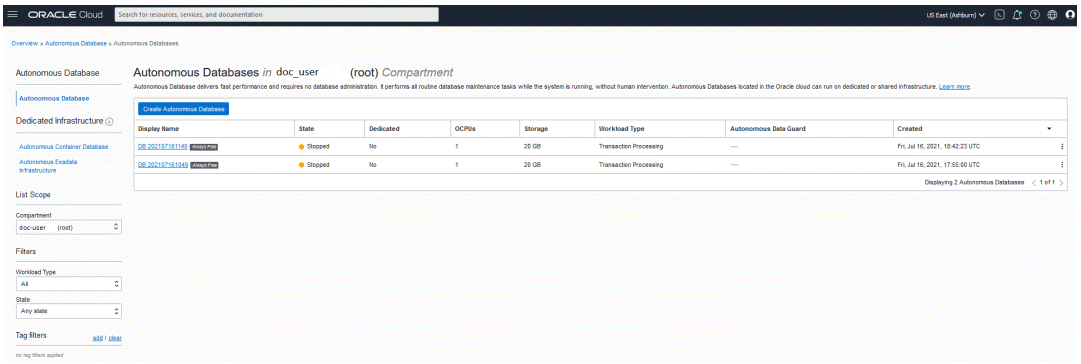
5. Click **Home** page, and then under **Featured**, click **Autonomous Database**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

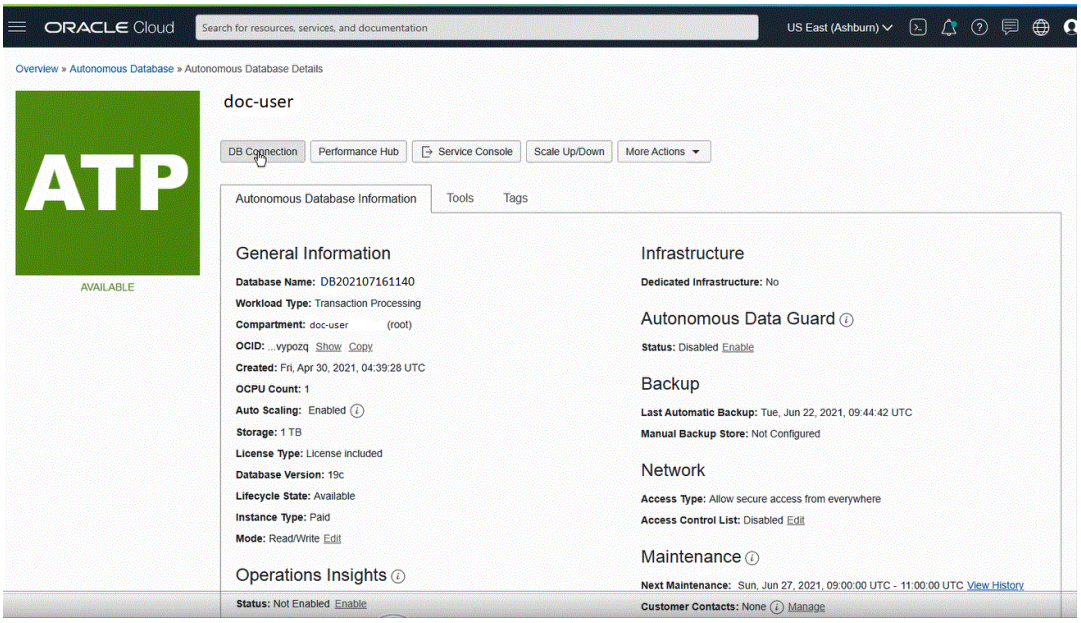


The Autonomous Data page opens showing the list of databases in your current region and compartment.

6. On the **Autonomous Database** page, under **Display Name**, select the autonomous database to which to connect.



7. Click **DB Connection**.



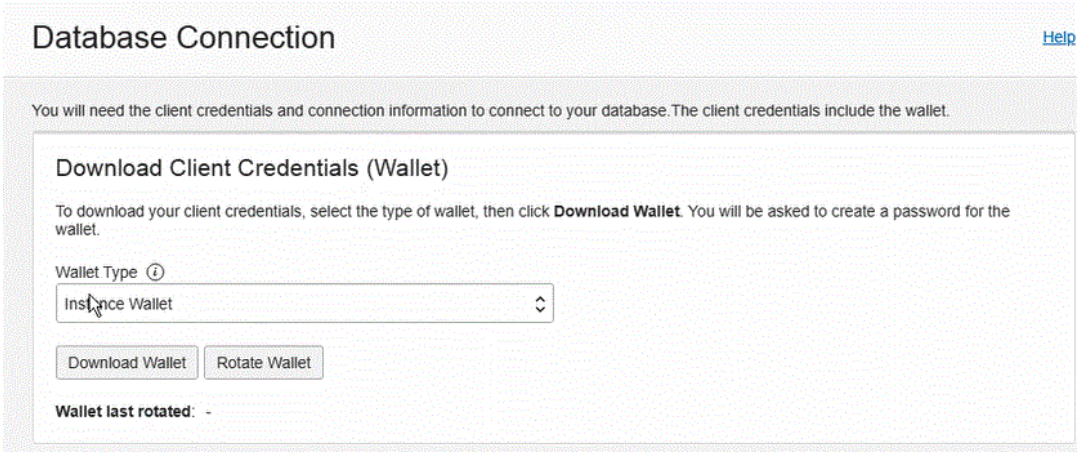
Copyright © 2017, 2026, Oracle and/or its affiliates.

8. On the **Data Connections** page, specify the client credentials and connection information to connect to your database by selecting the wallet type.

   A wallet is a password-protected container that stores authentication and signing credentials, including private keys, certificates, and trusted certificates, all of which are used by SSL for strong authentication.

   Available wallet types:

   - **Instance Wallet**: Wallet for a single database only; this provides a database-specific wallet.
   - **Regional Wallet**: Wallet for all Autonomous Databases for a given tenant and region (this includes all service instances that a cloud account owns).

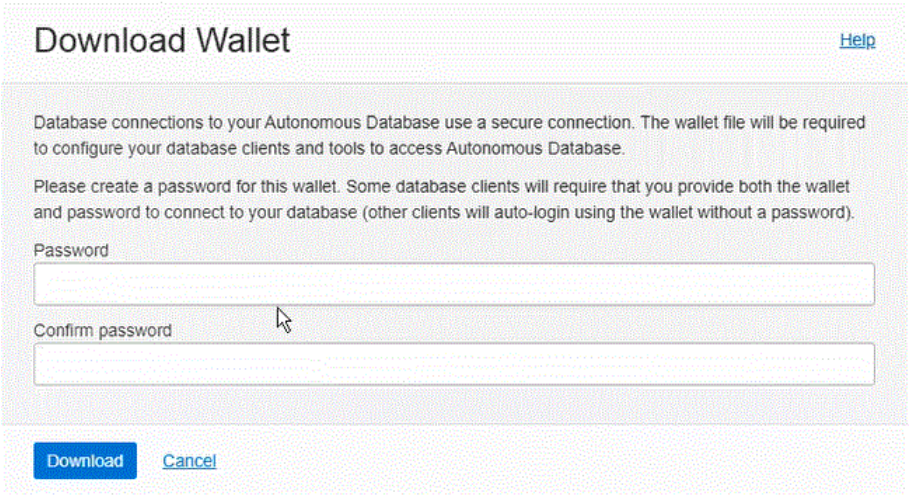


> **Note**
>
> Oracle recommends you provide a database-specific wallet using **Instance Wallet**, to end users and for application use whenever possible. Regional wallets should only be used for administrative purposes that require potential access to all Autonomous Databases within a region.

9. Click **Download Wallet**.
10. In the **Download Wallet** dialog, enter a wallet password in the **Password** field and confirm the password in the **Confirm Password** field.

    You are prompted to provide a password to encrypt the keys inside the wallet. The password must be at least 8 characters long and must include at least 1 letter and either 1 numeric character or 1 special character. This password protects the downloaded Client Credentials wallet.

Copyright © 2017, 2026, Oracle and/or its affiliates.



11. Click **Download** to save the client security credentials zip file.

    By default the filename is: `Wallet_`*databasename*`.zip`. You can save this file as any filename and to any local folder.

12. Click **Close**.

> **Note**
>
> To connect to the Oracle Autonomous Database using SQL Developer, see Connect SQL Developer to Autonomous Transaction Processing.

Configuring the EPM Integration Agent to Connect to the Oracle Autonomous Database

This section describes how to configure the EPM Integration Agent in order to connect to the Oracle Autonomous Database. It describes configuration steps for:

- Configuring the EPM Integration Agent on a Local Network
- Configuring the EPM Integration Agent on a Compute Instance

Configuring the EPM Integration Agent on a Local Network

Describes how to install the EPM Integration Agent on a local computer within your organization's network.

When loading or writing back data from and to the Oracle Autonomous Database, you install the agent on a local computer within your organization's network.

To configure the EPM Integration Agent in a local network:

1. In Data Integration, download the latest **EPMAgent ZIP**.

    a. Click **Data Exchange** (), and then select the **Data Integration** tab.

    Optionally, you can launch Data Integration by clicking **Navigator** (), and then from **Application**, selecting **Data Exchange** (Data Exchange).

    b. From the Data Integration home page, click **Actions**, then **Configure**, then select **Download Agent**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

c. From the **File Manager** page, extract the **EPMAgent ZIP** to your destination folder.

   The destination folder is `AGENT_HOME`.

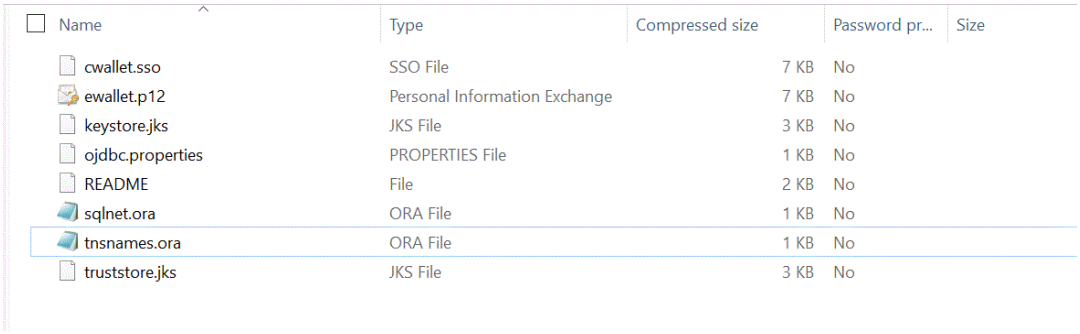
2. Navigate to folder where you downloaded the wallet when you created the connection to the Oracle Autonomous Database.

   By default the filename of the zip is: `Wallet_`*`databasename`*`.zip`.

   For information on creating the wallet, see Connecting to the Oracle Cloud Infrastructure.

3. Copy the wallet to a configuration directory such as <*EPMAgentDevData**config*> or create a wallet directory under *EPM_APP_DATA_HOME*.

4. Select the wallet and unzip it.



Name	Type	Compressed size	Password pr...	Size
cwallet.sso	SSO File	7 KB	No	
ewallet.p12	Personal Information Exchange	7 KB	No	
keystore.jks	JKS File	3 KB	No	
ojdbc.properties	PROPERTIES File	1 KB	No	
README	File	2 KB	No	
sqlnet.ora	ORA File	1 KB	No	
tnsnames.ora	ORA File	1 KB	No	
truststore.jks	JKS File	3 KB	No	

5. Add a proxy entry in the service description by opening the `tnsnames.ora` file in a text editor and replacing the proxy service and proxy port as shown in the example below:

```
myadb_high = (description= (retry_count=20)(retry_delay=3)
(address=(https_proxy=myproxy.sample.com)(https_proxy_port=80)
```

   The `tnsnames.ora` file is a configuration file that contains network service names mapped to connect descriptors for the local naming method, or net service names mapped to listener protocol addresses. This proxy information is required because you are running the agent within the network.

6. Save the `tnsnames.ora` file.

Deploying the EPM Integration Agent in a Cloud Infrastructure (OCI) Compute Instance

Describes how to install the EPM Integration Agent in a Cloud Infrastructure (OCI) Compute Instance.

When loading or writing back data to and from the Oracle Autonomous Database, you can put the EPM Integration Agent in an Oracle Cloud Infrastructure (OCI) Compute instance.

Copyright © 2017, 2026, Oracle and/or its affiliates.

The Oracle Cloud Infrastructure provides Compute instances so that you can provision and manage compute hosts. You can create instances as needed to meet your compute and application requirements such as providing the processing power and memory capacity for the virtual servers they host. After you create an instance, you can access it securely from your computer, restart it, attach and detach volumes, and terminate it when you're done with it. Any changes made to the instance's local drives are lost when you terminate it.

Deploying the EPM Agent on an Oracle Cloud Infrastructure Compute instance provides the following advantages:

- Simple installation
- No special network configuration is required such as a firewall or opening a port in an on-premises environment.
- All data traffic is in the Cloud and there is no data transfer to the on-premises environment.

When a Compute instance is used, the agent establishes connectivity between the Oracle Fusion Cloud Enterprise Performance Management and the Oracle Autonomous Database. This approach provides customers with scalability and flexibility for their solution. You still need to create the integration definitions including the Oracle Autonomous Database adapter, SQL query, and mapping within Data Integration.

Creating a Compute Instance

Describes how to create an Oracle Cloud Infrastructure Compute instance.

You create an Oracle Cloud Infrastructure Compute instance to which you install the EPM Integration Agent and then load and write back data to the Oracle Autonomous Database. In addition, you get local access to the other tools, utilities and other resources on the host system. This implementation uses the Secure Shell (SSH) client software to establish a secure connection and log in as an `opc` user.

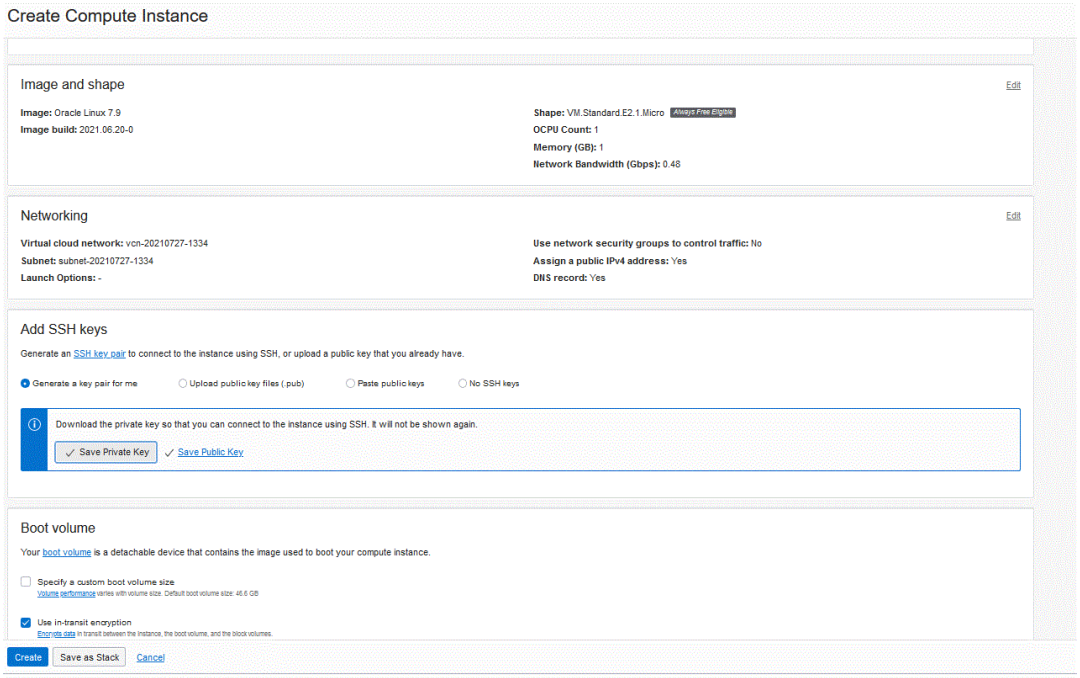
Before you begin creating the Compute instance, you need the following:

- An Oracle Cloud user account that has access rights to manage Compute instances in the given compartment, to use the given VCN and subnet, and to view information about the dedicated database you will be connecting to.
- The name of the compartment, VCN and subnet to use when creating the Oracle Cloud Compute instance. Your autonomous database administrator or tenancy administrator should be able to provide you this information.

To create to a Compute instance:

1. Sign in from **https://cloud.oracle.com**.
2. In **Cloud Account Name**, enter your cloud account name and click **Next**.
3. Enter your user name in **User Name** and password in **Password** and then click **Sign In**.
4. On the **Oracle Cloud Infrastructure** page, click the Navigation menu in the upper left to show top level navigation choices.
5. Click **Compute** and then under **Compute**, click **Instances**.
6. On the **Instances** page, click **Create Instance**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

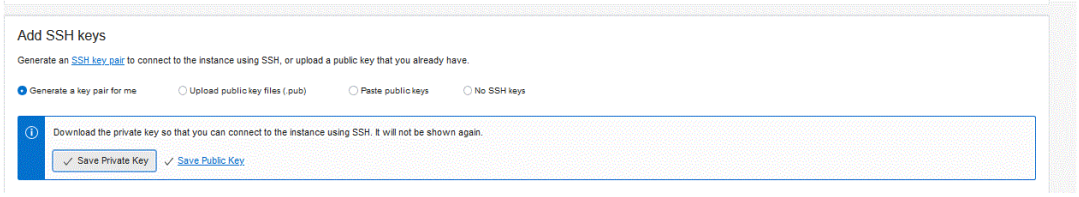


7. On the **Create Compute Instance** page, go to the **Add SSH Keys** pane, and then click **Generate a key pair for me**.

   SSH is a cryptographic network protocol that uses two keys, a public key and a private key, to provide secure communication between two computers. SSH uses port **22** by default.

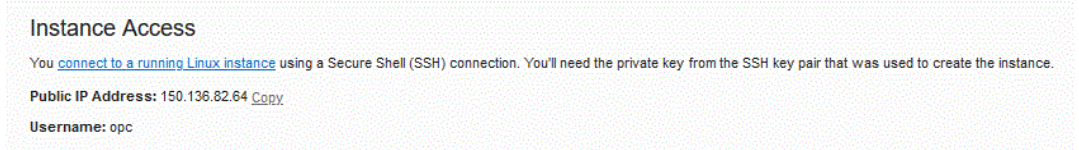
   Private key stays with the user (and only there), while the public key is sent to the server typically with the ssh-copy-id utility. The server stores the public key (and "marks" it as authorized). The server will now allow access to anyone who can prove they have the corresponding private key. Private key stays with the user (and only there), while the public key is sent to the server.

8. Select the options to **Save Private Key** and **Save Public Key** and save both keys to a directory where you can reference them later.



9. Click **Create**.
10. On the **Instances** page, click the name of the instance to display the instance detail.
11. From **Instance Access**, under **Public IP Address**, copy and save the IP address.

    You are required to specify the public IP address when connecting to the remote host.



Copyright © 2017, 2026, Oracle and/or its affiliates.

Connecting to the Compute Instance Using WSCP

Describes how to connect to a running instance by using a Secure Shell (SSH) or Remote Desktop connection.

You can connect to a running instance by using a Secure Shell (SSH) or Remote Desktop connection. Most UNIX-style systems include an SSH client by default. Windows 10 and Windows Server 2019 systems should include the OpenSSH client, which you need if you created your instance using the SSH keys generated by Oracle Cloud Infrastructure. For other Windows versions, you can connect to the command line using a free SSH client called PuTTY available for download.

Before you begin, you'll need the following information to connect to the instance:

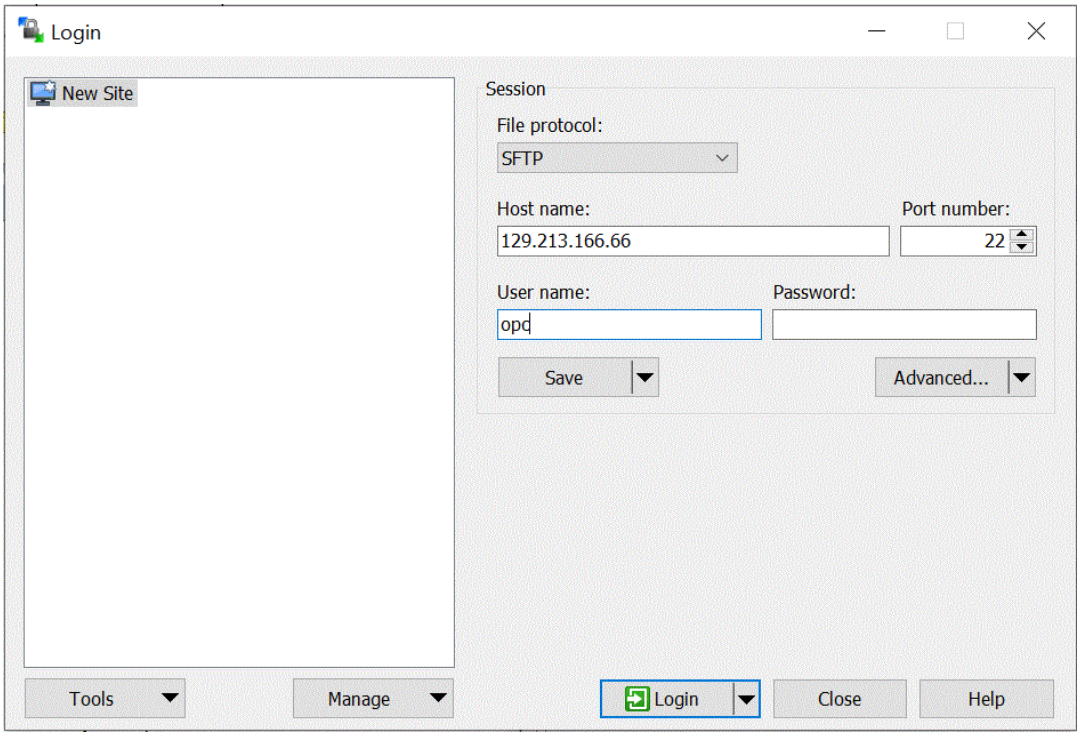
- The public IP address of the instance. You can get the address from the Instance Details page in the Console. Open the navigation menu and click **Compute**. Under **Compute**, click **Instances**. Then, select your instance. Alternatively, you can use the Core Services API ListVnicAttachments and GetVnic operations.
- The default username for the instance. If you used a platform image for Linux, CentOS, or Windows to launch the instance, the username is `opc`. If you used an Ubuntu platform image to launch the instance, the username is `ubuntu`.
- For Linux instances: The full path to the private key portion of the SSH key pair that you used when you launched the instance. For more information about key pairs, see Managing Key Pairs on Linux Instances.
- For Windows instances: If you're connecting to the instance for the first time, you will need the initial password for the instance. You can get the password from the Instance Details page in the Console.

To connect to the Compute instance:

1. Launch **WinSCP**.
2. On the **WinSCP** page, select **Session** and then **New Session**.
3. On the **Login** page, and then **Host Name**, enter the **Public IP Address** that you copied and saved.
4. In **Port number**, leave the default value **22**.
5. In **User name**, enter **opc**.

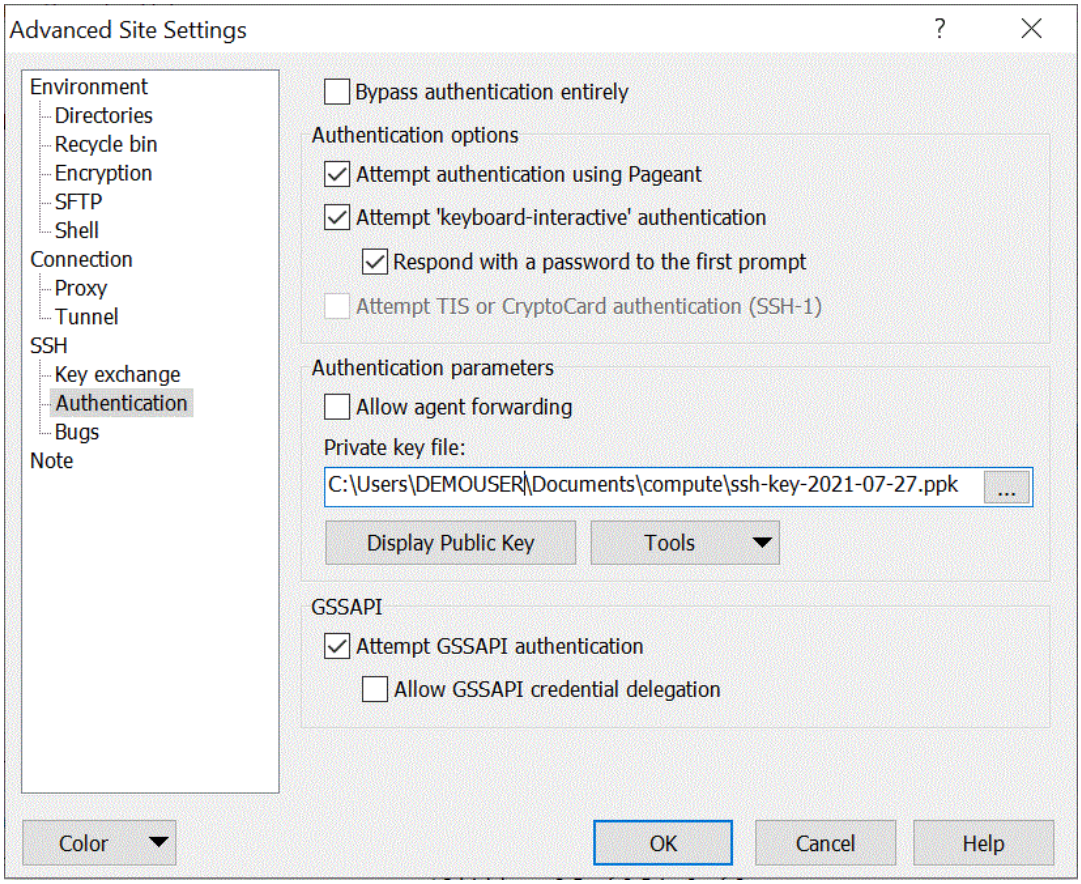
   The `opc` user can perform operations that require root access to the compute node, such as backing up or patching; this user can use the `sudo` command to gain root access to the compute instance.

Copyright © 2017, 2026, Oracle and/or its affiliates.



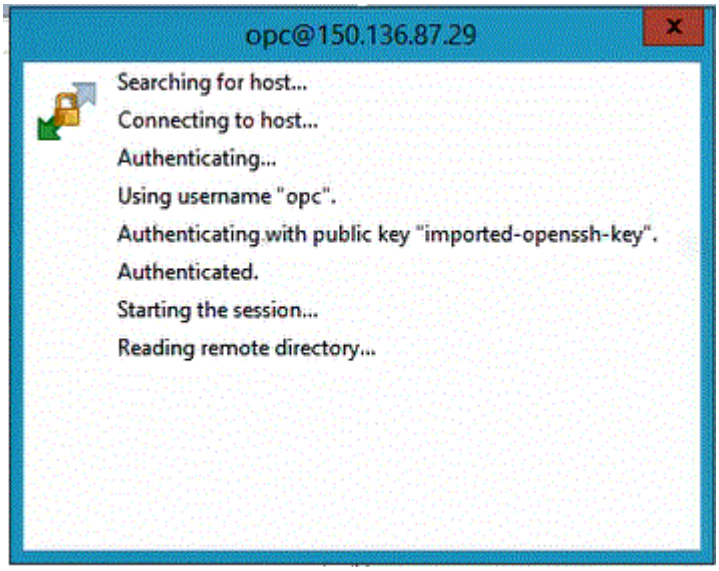
6. From the **Advanced** drop-down, select **Advanced**.
7. On the **Advanced Site Settings** page, select **Connection** and then **Proxy**.
8. From **Proxy Type** drop-down, select **HTTP**.
9. In **User name**, enter the customer proxy server.
10. In **Port number**, leave the port number as **80**.
11. Click **SSH**, then **Key exchange**, and then **Authentication**.
12. From **Private key file**, click and navigate to the private key that you saved.
13. Click **OK**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



14. On the **Login** page, click **Login**.

    Messages are shown authenticating the opc login information.



Copyright © 2017, 2026, Oracle and/or its affiliates.

Connecting to the Compute Instance Using Putty

Describes how to connect to a running instance by using a Secure Shell (SSH) or Remote Desktop connection. For other Windows versions, you can connect to the command line using a free SSH client called PuTTY.

You can connect to a running instance by using a Secure Shell (SSH) or Remote Desktop connection. Most UNIX-style systems include an SSH client by default. Windows 10 and Windows Server 2019 systems should include the OpenSSH client, which you need if you created your instance using the SSH keys generated by Oracle Cloud Infrastructure. For other Windows versions, you can connect to the command line using a free SSH client called PuTTY available from https://www.putty.org

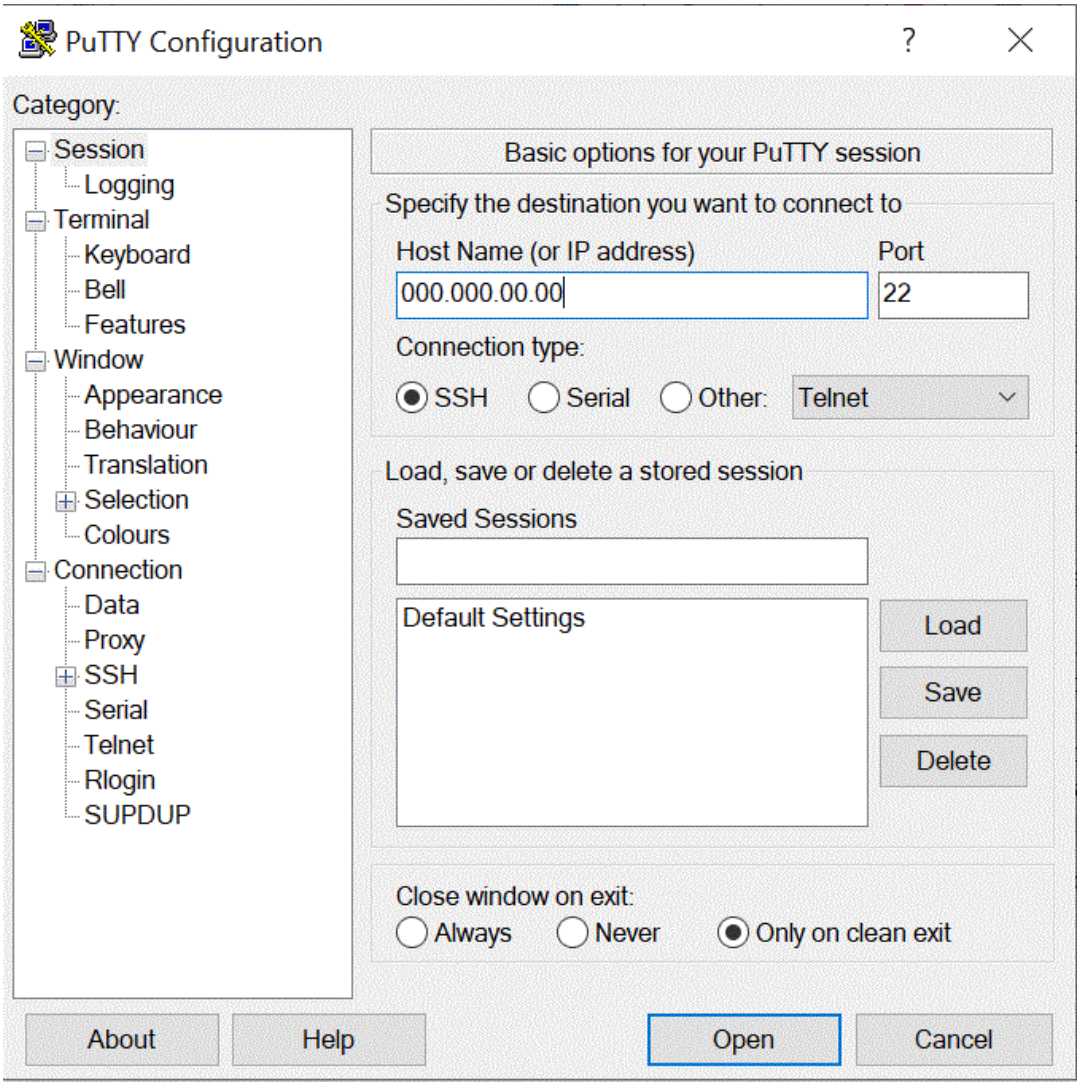
Before you begin, you'll need the following information to connect to the instance:

- The public IP address of the instance. You can get the address from the Instance Details page in the Console. Open the navigation menu and click **Compute**. Under **Compute**, click **Instances**. Then, select your instance. Alternatively, you can use the Core Services API ListVnicAttachments and GetVnic operations.
- The default username for the instance. If you used a platform image for Linux, CentOS, or Windows to launch the instance, the username is `opc`. If you used an Ubuntu platform image to launch the instance, the username is `ubuntu`.
- For Linux instances: The full path to the private key portion of the SSH key pair that you used when you launched the instance. For more information about key pairs, see Managing Key Pairs on Linux Instances.
- For Windows instances: If you're connecting to the instance for the first time, you will need the initial password for the instance. You can get the password from the Instance Details page in the Console.

To connect to the Compute instance:

1. Launch **PuTTY**.
2. On the **Basic** page, in **Host Name**, enter the public IP address that you copied and saved when creating the Compute instance.
3. In **Port number**, leave the default value **22**.

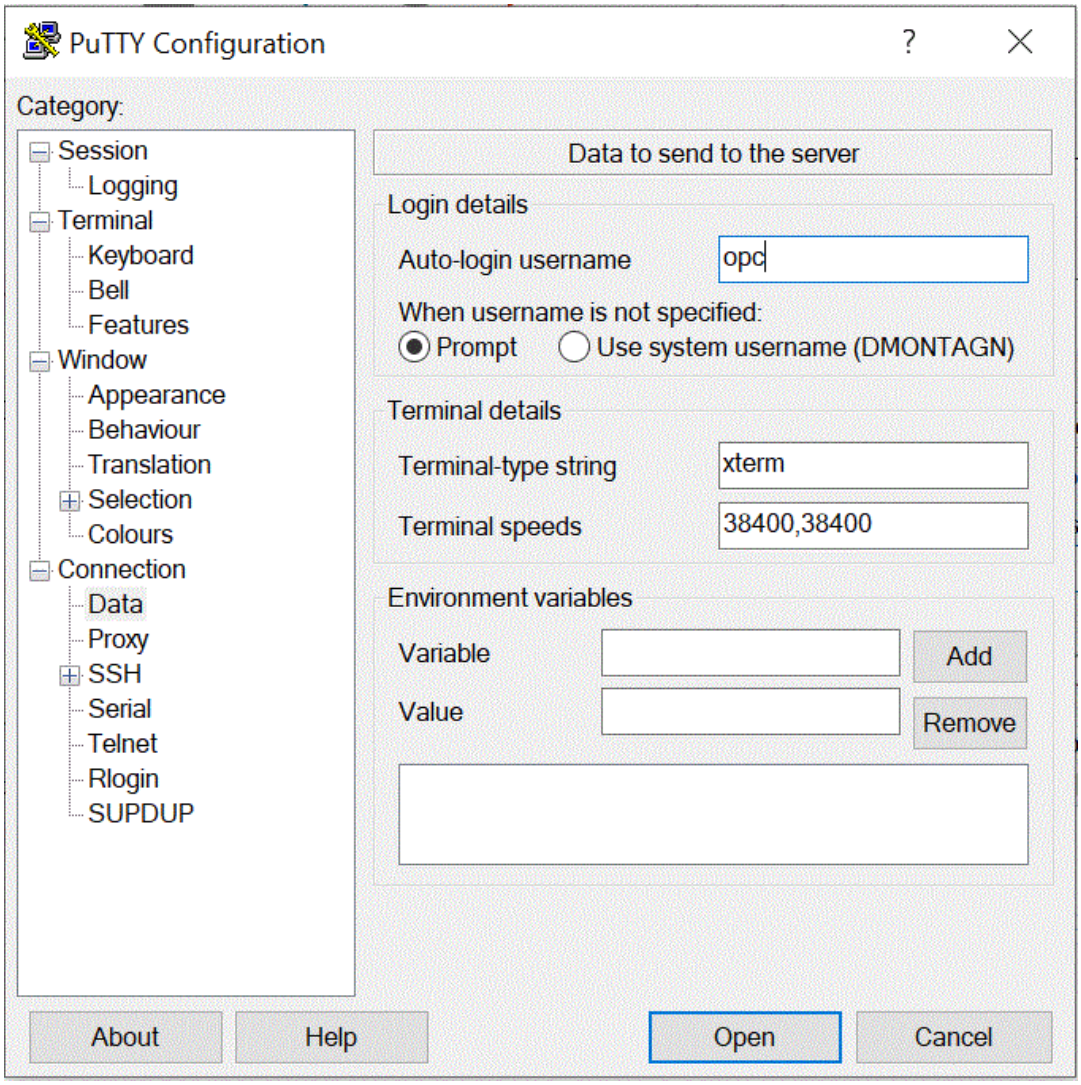
Copyright © 2017, 2026, Oracle and/or its affiliates.



4. Under **Connection**, then **Data**, and then **Auto-login user name**, enter **opc**.

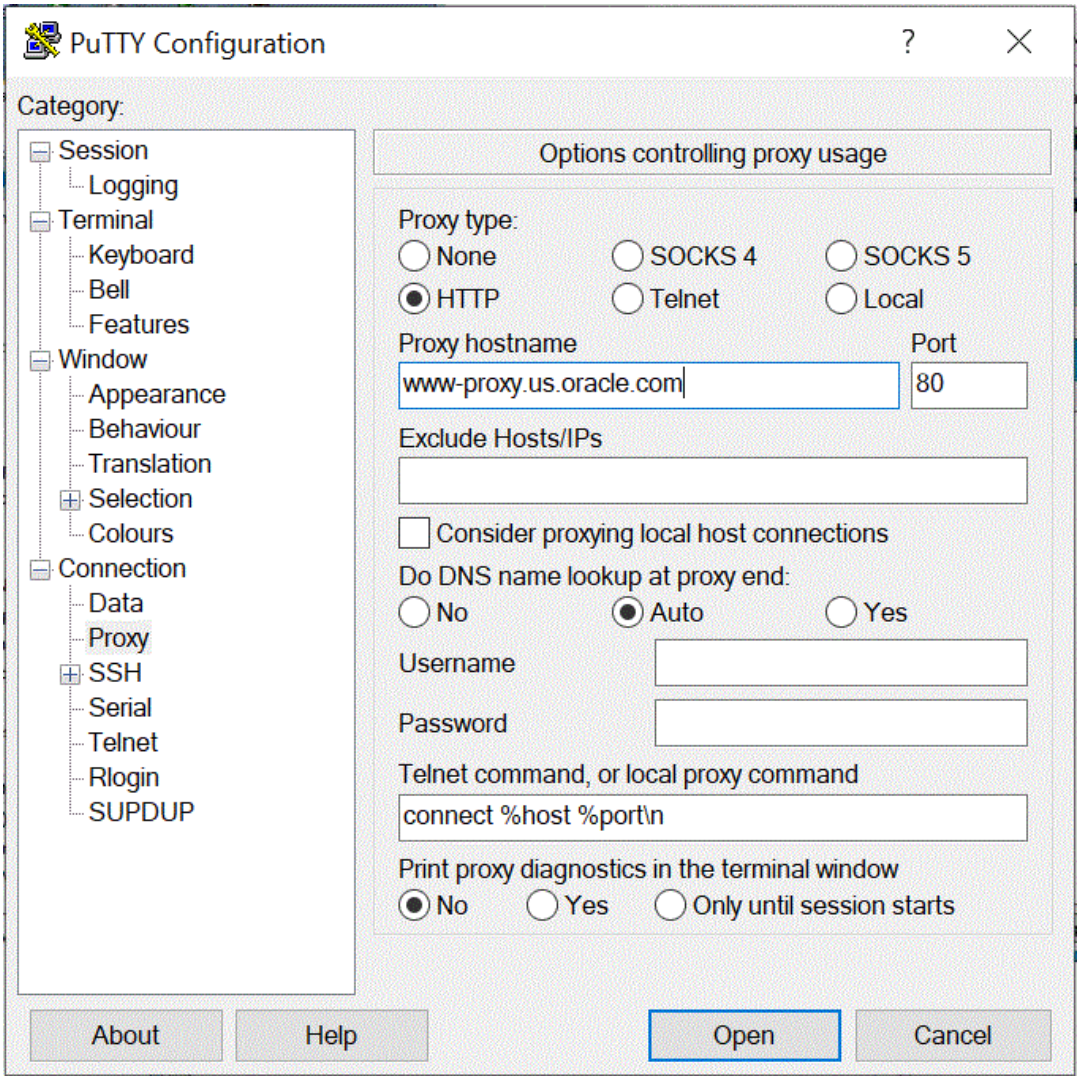
   The `opc` user can perform operations that require root access to the compute node, such as backing up or patching; this user can use the `sudo` command to gain root access to the compute instance.

Copyright © 2017, 2026, Oracle and/or its affiliates.



5. Under **Connection**, then **Proxy**, and then **Proxy Type**, select **HTTP**.
6. In **Proxy hostname**, enter the customer proxy server address.
7. In **Port number**, leave the port number as **80**.

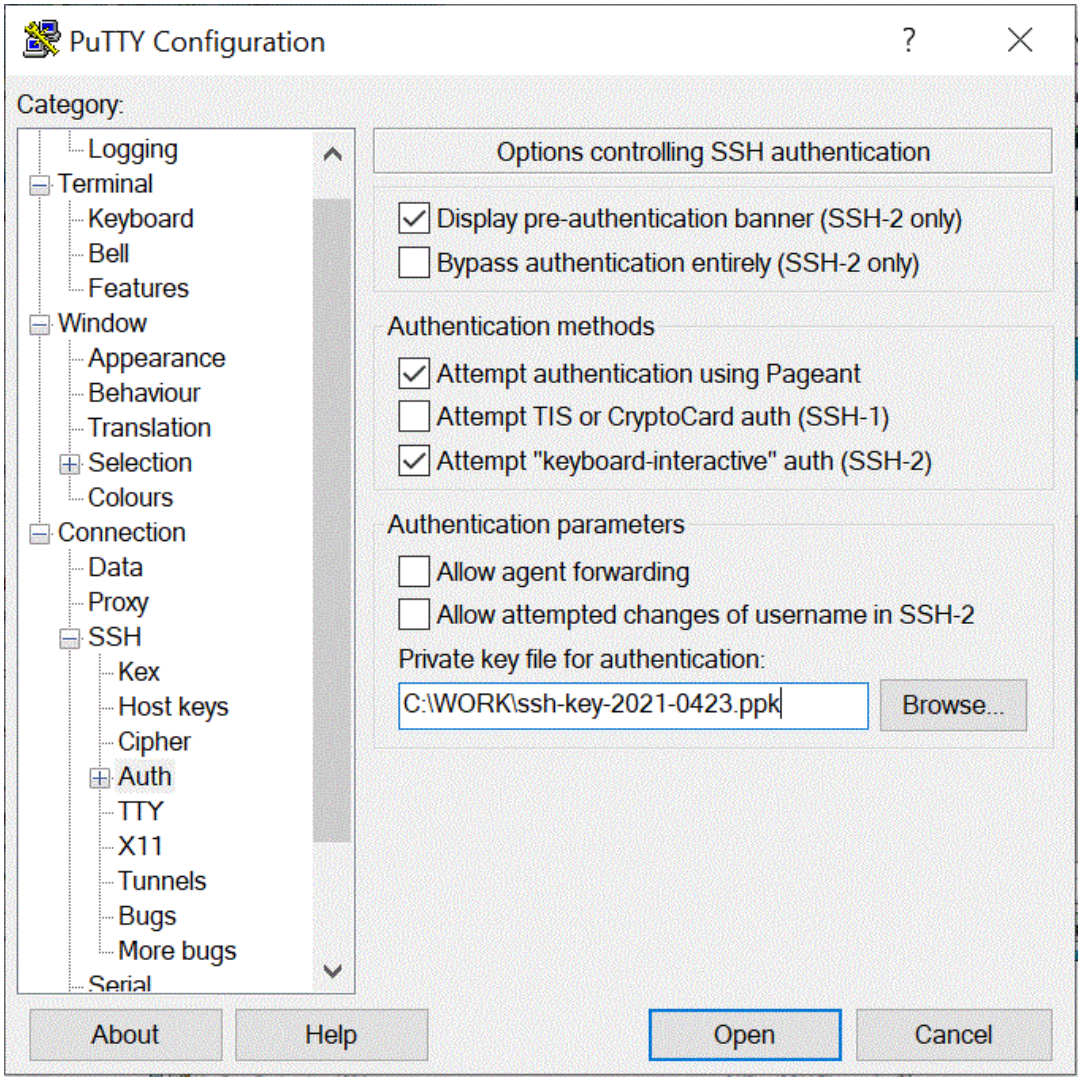
Copyright © 2017, 2026, Oracle and/or its affiliates.



8. Under **Connection**, expand **+SSH**, and then select **Auth**.
9. From **Private key file for authentication**, enter the private key that you received when creating the Compute instance.

   You can also click ... and navigate to the private key that you saved.

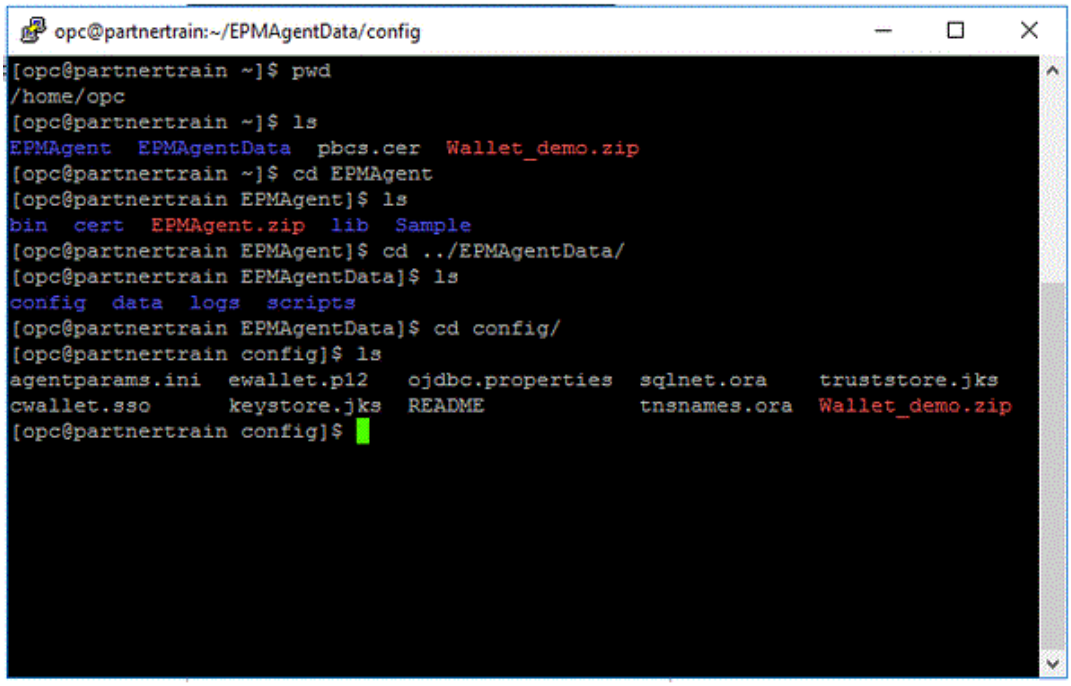
Copyright © 2017, 2026, Oracle and/or its affiliates.



10. Click **Open** to launch the Compute instance.

    In the following example, the agent has already been installed to the Compute instance and has been started.

Copyright © 2017, 2026, Oracle and/or its affiliates.



Configuring the EPM Integration Agent on a Compute Instance

Describes how to configure the EPM Integration Agent on a Compute instance.

After you have created an Oracle Cloud Compute instance, you need to connect to it, and then transfer and install the EPM Integration Agent and other related components.

> **Note**
>
> This implementation uses the Secure Shell (SSH) client software to establish a secure connection and log in as a `opc` user.

Before you begin creating compute instance system, you need the following:

- An Oracle Cloud user account that has access rights to manage Compute instances in the given compartment, to use the given VCN and subnet, and to view information about the dedicated database you will be connecting to.
- The name of the compartment, VCN and subnet to use when creating the Oracle Cloud Compute instance. Your autonomous database administrator or tenancy administrator should be able to provide you this information.
- For installing and configuring the EPM Integration Agent on the Compute instance, you will need the following files:
  - EPMAgent.zip
  - Wallet
  - SSL certificate for use with the Oracle Fusion Cloud Enterprise Performance Management
- WinSCP for transferring files. WinSCP is a free download available at: https://winscp.net/eng/index.php

To configure the EPM Integration Agent on a Compute instance:

Copyright © 2017, 2026, Oracle and/or its affiliates.

1. Connect to the Compute instance with the host name, user name, password or SSH key.

   For more information about connecting to the Compute Instance, see Connecting to the Compute Instance Using Putty or Connecting to the Compute Instance Using WSCP.

   Additional connection information is available at **Connecting to an Instance**.

2. Install **Oracle Java** on Oracle Linux running in your OCI compute shape by using RPMs available from the OCI yum service.

   For instructions on installing Oracle Java, see How to Install Oracle Java in Oracle Cloud Infrastructure.

3. Log into **WinSCP** with the host name, user name, and password, and then click **Login**.

   WinSCP opens in a directory structure similar to Windows File Explorer.

4. Transfer the following files by dragging and dropping them from your source directory to a directory on the Compute instance:

   - **EPMAgent.zip**.

     The latest EMPAgent.zip is available for download from your local network from Data Integration. For more information, see Downloading the EPMAgent ZIP.

   - **Wallet**—The wallet files store the client credentials from the Autonomous Database Service Console.

     For more information about the wallet, see Connecting to the Oracle Cloud Infrastructure.

   - **SSL certificate**—Certificate used with the Cloud EPM.

     The certificate is required to enable an encrypted connection between the Cloud EPM and the EPM Integration Agent.

     Optionally, you can also use the `scp` command line utility to securely copy files and directories between your local system to a remote system. The `SCP` command syntax uses the following format: `scp [OPTION][user@]SRC_HOST:]file1 [user@]DEST_HOST:]file2`

     Here are sample `scp` commands that you might use when transferring the EPM Integration Agent components required for this configuration on Linux and Mac.

     ```
     scp -i /Users/Oracle/Documents/oci/ssh-key-private.key epmagent.jar opc@<OCI INSTANCE PUBLIC IP ADDRESS>:/home/opc/epmagent.jar
     ```

     ```
     scp -i /Users/Oracle/Documents/oci/compute/ssh-key- private.key epm.cer opc@<OCI INSTANCE PUBLIC IP ADDRESS>:/home/opc/epm.cer
     ```

     ```
     scp -i /Users/Oracle/Documents/oci/compute/ssh-key- private.key Wallet-epm.zip opc@<OCI INSTANCE PUBLIC IP ADDRESS>:/home/opc/Wallet-epm.zip
     ```

5. Extract the **EPMAgent.zip** in your destination folder.

   Typically, the destination folder is *<AGENT_HOME>*.

6. Copy the wallet to a configuration directory or create a wallet directory at *<EPMAgentDevData\config>* or *<EPM_APP_DATA_HOME>* and then unzip it.

7. Copy the certificate to the *<EPM_AGENT_HOME/cert>* directory.

8. Start the agent on the Compute instance.

   For more information, see Starting the EPM Integration Agent.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Loading Data from the Oracle Autonomous Database to the Cloud EPM Using EPM Integration Agent

Describes how to load data from the Oracle Autonomous Database to the Oracle Fusion Cloud Enterprise Performance Management.

You can extract data from the Oracle Autonomous Database by executing a relational database SQL query against a database in the Oracle Autonomous Database and then load the data or metadata to the Cloud EPM using the EPM Integration Agent in your local network.

The EPM Integration agent is defined as a data source for an integration, and when executed provides data that is subsequently mapped and loaded to the selected target application.

As a prerequisite to loading data you must be connected to the Oracle Autonomous Database. In addition, you must have already downloaded the Oracle Wallet. The wallet provides an simple and easy method to manage database credentials across multiple domains.

To load data from the Oracle Autonomous Database to the Cloud EPM:

1. In Data Integration, create the query definition and save the SQL query.

   To do this, see Creating an SQL Query.

   If you already have an SQL query registered in Data Integration, skip this step.
2. Create a file that contains one header row from the source data.

   The header row must exactly match the dimension name in the target dimension. You can use an alias for a column name in the SQL query to the dimension name.

	A	B	C	D	E
1	Entity	Account	Product	Amount	
2	580	DPO	P_200	1233.98	
3	580	7300	P_200	45100.56	
4	580	7850	P_200	4000.33	
5	580	Capital Eq	P_200	-11900.7	
6	511	NI	P_200	17744.81	
7	580	NI	P_200	100	
8	580	NI	P_200	700	

3. Save the file as a CSV format file with a header row.

   Upload the file using the file browser when registering the application.
4. Register the Oracle Autonomous data source application:
   a. From the **Data Integration** home page, and then **Actions**, select **Applications**.
   b. On the **Applications** page, click .
   c. From **Create Application**, then **Category**, select **Data Source**.
   d. From **Type**, select **Oracle Autonomous Database - Agent**.
   e. From **File**, select the file that you created in step 8.

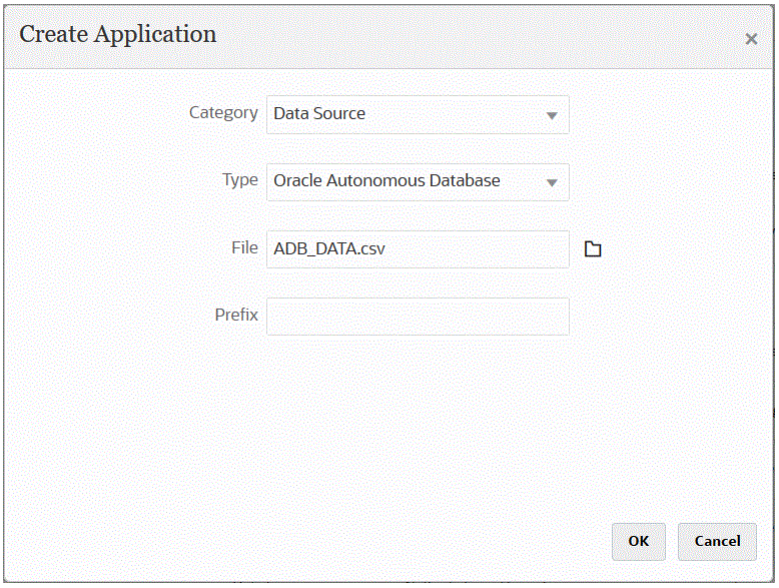
      Click to browse for the file on the File Browser page.

Copyright © 2017, 2026, Oracle and/or its affiliates.

f. In **Prefix**, specify a prefix to make the application name unique.

   The prefix is concatenated with the application name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.

g. Click **OK** and then click **Save**.



5. Click **OK** and then click **Save**.
6. On the **Application** page, click ... next to Oracle Autonomous data source application and then select **Application Details**.
7. On the **Application Details** page, click the **Options** tab.
8. In **Data Extract Query**, specify the name of the SQL query to run against the file.
9. In **Delimiter**, select the type of delimiter used in the file.

   Available delimiter symbols include the following:

   - Comma (,)
   - Exclamation (!)
   - Semicolon (;)
   - Colon (:)
   - Vertical bar (|)

10. In **Credential Store**, specify the type of credential store used by the EPM Integration Agent.

    Available types of credential stores include the following:

    - Cloud
    - File

    For the **Cloud** credential store type, store the *username/password/connect* string in the application.

Copyright © 2017, 2026, Oracle and/or its affiliates.

For a **File** credential store type, create a file that stores the JDBC URL, user name and password for the Oracle Autonomous Database connection. The URL format must include the JDBC URL, path to wallet folder, user name, and password. The file name must be named *appname*`.cred` and stored in the `config` directory.

The file must contain the following lines:

```
jdbcurl=jdbc:oracle:thin:@<tns_name>?TNS_ADMIN=<path_to_wallet_folder>
```

An example of an entry in the `.cred` file might look like this:

```
jdbcurl=jdbc:oracle:thin:@epmdevdb0_low?TNS_ADMIN=C:\ATP
Connection\epmdevdb0_wallet
username=apps
password=w+Sz+WjKpL8[
```

> **Note**
>
> The password used for both credential store types must be encrypted.
>
> When the type is a "Cloud" store, type the password in the user interface in the usual way. The system encrypts and stores the password.
>
> When the type is a "File" store, you must encrypt the password using the `encryptpassword` utility and store the password in the file. For more information about encrypting your password, see Encrypting the Cloud EPM User Password.

11. In **Username**, specify the user name that you used to connect to the Oracle Autonomous Database.

    If you connect to the Oracle Autonomous Database using SQL Developer, specify the SQL Developer username and password.

12. In **Password**, specify the password that you used to connect to the Oracle Autonomous Database.

13. In **Service Name**, specify the service name.

    You can find the service name by opening the `tnsnames.ora` file and selecting a service name associated with the connection. A service name is an alias mapped to a database network address contained in a connect descriptor.

    You are required to select a service name when you connect to the database. The service names for Autonomous Data Warehouse connections are in the format:

    - *databasename*_high
    - *databasename*_medium
    - *databasename*_low

    These services map to the `LOW`, `MEDIUM`, and `HIGH` consumer groups.

    For example, if you create an Autonomous Database with a Data Warehouse workload type and specify the database name as `DB2020`, your service names are:

    - db2020_high
    - db2020_medium
    - db2020_low

Copyright © 2017, 2026, Oracle and/or its affiliates.

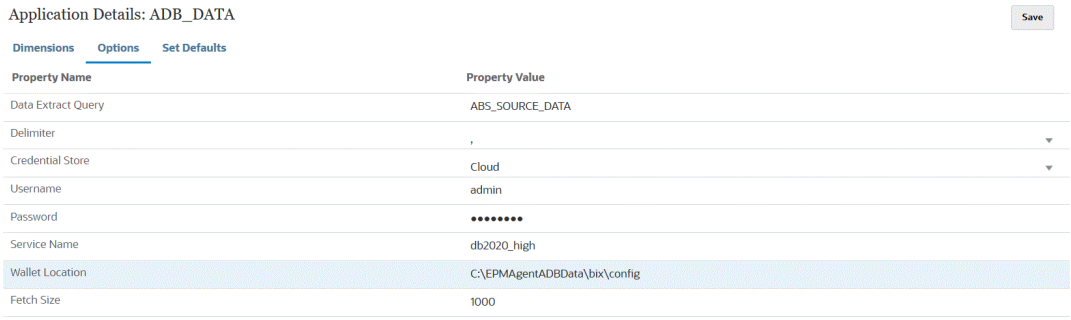
If you connect using the `db2020_low` service, the connection uses the `LOW` consumer group.

The basic characteristics of these consumer groups are:

- HIGH: Highest resources, lowest concurrency. Queries run in parallel.
- MEDIUM: Less resources, higher concurrency. Queries run in parallel.

  You can modify the MEDIUM service concurrency limit. See Change MEDIUM Service Concurrency Limit for more information.
- LOW: Least resources, highest concurrency. Queries run serially.

14. In **Fetch Size**, specify the number of rows fetched (that the driver processes) with each database round trip for a query.
15. In **Wallet Location**, specify the location of the directory where your wallet has been downloaded to and unzipped in the EPM Integration Agent server.



16. Click **Save**.
17. Finish integrating the Oracle Autonomous Database - Agent data source with the Cloud EPM by completing the following:

Task	Description	More Information
General	Add or edit an integration for file-based and direct integration sources.	Defining a Data Integration
Map Dimensions	Map the columns in the data source to dimensions in the target application.	Mapping Dimensions
Map Members	Map dimensions to identify how source field values translate to target dimension members.	Mapping Members
Options	Define options for importing and exporting data. Also, define any source filters.	Setting Data Integration Options

> **Note**
>
> When editing an Oracle Autonomous Database integration where the same location is used but the target application has been changed from an on-premises source to an Oracle Autonomous Database source, the integration will fail at runtime.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Task	Description	More Information
Run the Integration	When the integration is executed, the EPM Integration Agent initiates the export process in the Cloud EPM. The agent downloads the export data. Based on the selected method, the agent also constructs the appropriate INSERT statement and loads the data to the target table. You can review the exported data by downloading the output file from the Process Details. The database column names for the write-back are specified as the column headers in the generated file as shown below.	For more information, see Running an Integration

Writing Back Data from the Cloud EPM to the Oracle Autonomous Database

Describes how to write back data to the Oracle Autonomous Database.

The EPM Integration Agent supports write-back by enabling you to move data from your Oracle Fusion Cloud Enterprise Performance Management applications to the Oracle Autonomous Database on a local network. To use this feature, register a data export application to export data to a selected reporting data warehouse in the Oracle Autonomous Database.

Additionally, the agent can execute two event scripts during the write-back execution: `BefExport` and `AftExport`. Using a `BefExport` event, you can perform any action before inserting data to the table or you can override the default insert processing. Using an `AftExport` event, you can do any post processing cleanup after inserting data to the table.

The write-back feature is supported for agents running in both SYNC and ASYNC modes.

To write back data to the Oracle Autonomous Database:

1. In Data Integration, create a file that contains one header row from the source data that you want to write back.

   The header row must exactly match the dimension name in the target dimension. You can use an alias for a column name in the SQL query to the dimension name.

	A	B	C	D	E
1	Entity	Account	Product	Amount	
2	580	DPO	P_200	1233.98	
3	580	7300	P_200	45100.56	
4	580	7850	P_200	4000.33	
5	580	Capital Eq	P_200	-11900.7	
6	511	NI	P_200	17744.81	
7	580	NI	P_200	100	
8	580	NI	P_200	700	

2. Save the file as a CSV format file with a header row.

   Upload the file using the file browser when registering the application.

3. Register a data export application with the type **Data Export to Oracle Autonomous Database**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

a. From the **Data Integration** home page, and then **Actions**, select **Applications**.

b. On the **Applications** page, click .

c. From **Create Application**, then **Category**, select **Data Export**.

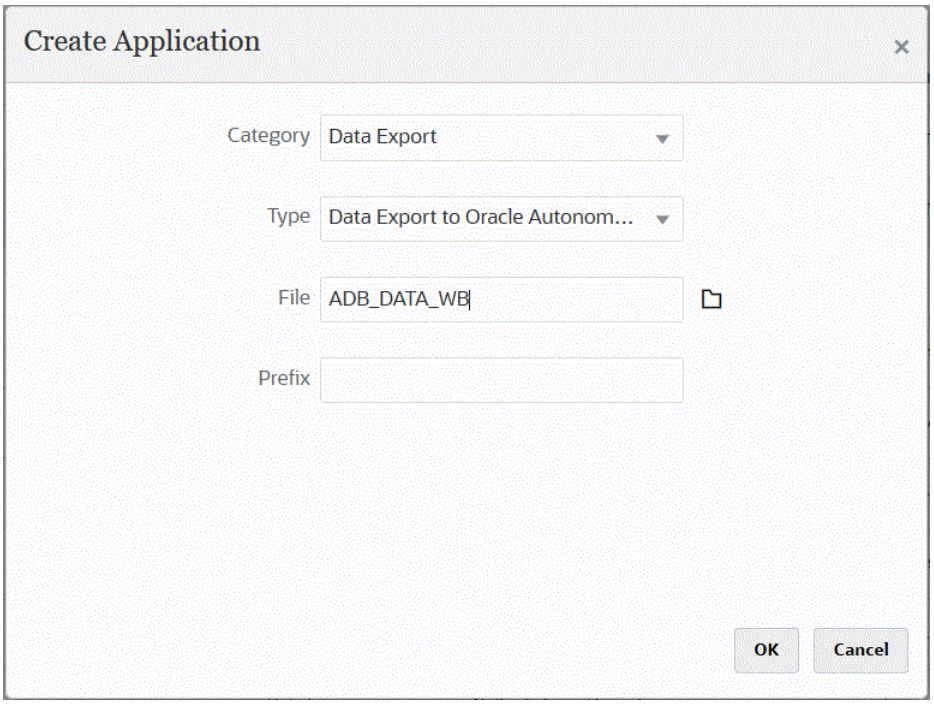
d. From **Type**, select **Data Export to Oracle Autonomous Database**.

e. From **File**, select the file that you created in step 2.

Click to browse for the file on the File Browser page.

f. In **Prefix**, specify a prefix to make the application name unique.

The prefix is concatenated with the application name to form a unique application name. For example, if you want to name an application with the same name as an existing one, you can assign your initials as the prefix.



g. Click **OK** and then click **Save**.

4. On the **Application** page, click next to the Data Export to Oracle Autonomous Database source application and then select **Application Details**.

5. Go to **Application Details Dimensions** tab for the Data Export to Oracle Autonomous Database source application and verify that the column names in the header record match exactly the column name or column alias in the table to which you are loading data.

Names are case-sensitive.

For more information, see Setting the Application Detail Dimensions for the Write-Back.

6. On the **Application Details** page, click the **Options** tab.

7. In **Batch Size**, specify the number of rows read at a time from the file to memory. This parameter is mainly used for performance. When data is loaded, this setting determines how many records are stored in the cache. For example, when **1000** is

Copyright © 2017, 2026, Oracle and/or its affiliates.

specified; 1,000 records are stored in cache. Similarly, when **5000** is specified, 5,000 records are stored in cache and commit.

Determine this setting by Server Memory and adjust as needed.

8. In **Table Name**, specify the table name to which to load the data.

   The name of the dimensions in the data export to on-premises application must match the names of the columns in the table. The system generates the INSERT statement to load the data automatically and inserts text for all fields.

   If you specify a table name in this field, do not specify an INSERT statement in the **Insert Query** field.

9. In **Insert Query**, specify the name of the SQL query associated with the custom INSERT statement created on the Query Definition page.

   INSERT query must contain the table and columns for values and the dimension name of the application. Using this method, you can provide a user-friendly dimension name and also use SQL functions like TO_DATE, TO_NUMBER and other functions to perform data type conversions and other string operations. For more information, see Writing Back with a Custom INSERT Query.

   If you specify an INSERT query, do not specify a table name in the **Table Name** field.

10. In **Credential Store**, specify the type of credential store used by the EPM Integration Agent.

    Available types of credential stores include the following:

    - Cloud
    - File

    For the **Cloud** credential store type, store the *username/password/connect* string in the application.

    For a **File** credential store type, create a file that stores the JDBC URL, user name and password for the Oracle Autonomous Database connection. The URL format must include the JDBC URL, path to wallet folder, user name, and password. The file name must be named *appname*`.cred` and stored in the `config` directory.

    The file must contain the following lines:

    ```
    jdbcurl=jdbc:oracle:thin:@<tns_name>?TNS_ADMIN=<path_to_wallet_folder>
    ```

    An example of an entry in the `.cred` file might look like this:

    ```
    jdbcurl=jdbc:oracle:thin:@epmdevdb0_low?TNS_ADMIN=C:\ATP
    Connection\epmdevdb0_wallet
    username=apps
    password=w+Sz+WjKpL8[
    ```


Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> The password used for both credential store types must be encrypted.
>
> When the type is a "Cloud" store, type the password in the user interface in the usual way. The system encrypts and stores the password.
>
> When the type is a "File" store, you must encrypt the password using the `encryptpassword` utility and store the password in the file. For more information about encrypting your password, see Encrypting the Cloud EPM User Password.

11. In **Workflow Mode**, specify the method for how data is loaded.

    By default, the data load process in Data Management is designed with a well-defined process flow that provides a full audit of the data load process and the ability to drill down and view data in Workbench. However, a full data flow may contribute to lengthy processing times due to archiving of data for audit purposes. Workflow mode options provide scalable solutions when processing large volumes of data, or when an audit is not required, and performance is a key requirement.

    The three workflow mode options are:

    - Full
    - Full (no archive)
    - Simple

    The Full option is the default flow for loading data. Data is loaded in the standard way between the staging tables, data can be viewed in Workbench, and drill down is supported.

    The Full (no archive) option loads data in the same manner as the full mode, but data is deleted from the staging tables at the end of the data load process. Data can be viewed in the Workbench only after the import step. No drill down is available with the full (no archive) mode. This method is useful when you want to review and validate the data during load process but auditing or drill down is not a requirement.

    This method does not provide performance improvement but limits space usage since data is not persisted for future reference.

    The Simple option limits data movement between the staging tables. No drill down is available, and the data cannot be viewed in the Workbench. This method provides performance improvement and is useful when you do not require auditing or drill down.

    The Simple option limits data movement between the staging tables. No drill down is available, and the data cannot be viewed in the Workbench. This method provides performance improvement and is useful when you do not require auditing or drill down.

12. In **Username**, specify the user name that you used to connect to the Oracle Autonomous Database.

    If you connect to the Oracle Autonomous Database using SQL Developer, specify the SQL Developer username and password.

13. In **Password**, specify the password that you used to connect to the Oracle Autonomous Database.

14. In **Service Name**, specify the service name.

    You can find the service name by opening the `tnsnames.ora` file and selecting a service name associated with the connection. A service name is an alias mapped to a database network address contained in a connect descriptor.

Copyright © 2017, 2026, Oracle and/or its affiliates.

You are required to select a service name when you connect to the database. The service names for Autonomous Data Warehouse connections are in the format:

- *databasename*_high
- *databasename*_medium
- *databasename*_low

These services map to the `LOW`, `MEDIUM`, and `HIGH` consumer groups.

For example, if you create an Autonomous Database with a Data Warehouse workload type and specify the database name as `DB2020`, your service names are:

- db2020_high
- db2020_medium
- db2020_low

If you connect using the `db2020_low` service, the connection uses the `LOW` consumer group.

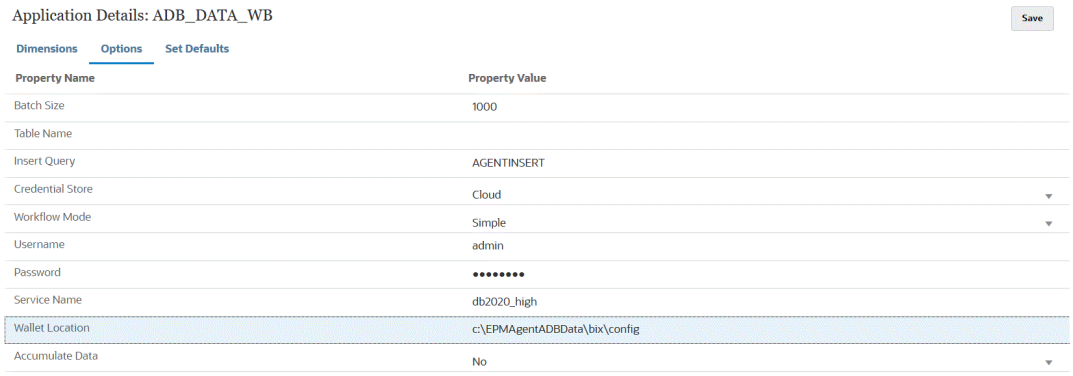
The basic characteristics of these consumer groups are:

- HIGH: Highest resources, lowest concurrency. Queries run in parallel.
- MEDIUM: Less resources, higher concurrency. Queries run in parallel.

  You can modify the MEDIUM service concurrency limit. See Change MEDIUM Service Concurrency Limit for more information.
- LOW: Least resources, highest concurrency. Queries run serially.

15. In **Wallet Location**, specify the location of the directory where your wallet has been downloaded to and unzipped in the EPM Integration Agent server.
16. In **Accumulate Data**, enter **yes** to summarize Account data before export and group the results by one or more columns.

    Enter **no** not to summarize the Account data before export and not group the results by one or more column.



17. Click **Save**.
18. Integrate the Cloud EPM application with the data export to Oracle Autonomous Database application by completing the following tasks.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Task	Description	More Information
General	Add or edit an integration for file-based and direct integration sources.	Defining a Data Integration
Map Dimensions	Map the columns in the data source to dimensions in the target application.	Mapping Dimensions
Map Members	Map dimensions to identify how source field values translate to target dimension members.	Mapping Members
Options	Define options for importing and exporting data. Also, define any source filters.	Setting Data Integration Options
Run the Integration	When the integration is executed, the EPM Integration Agent initiates the export process in the Cloud EPM. The agent downloads the export data. Based on the selected method, the agent also constructs the appropriate INSERT statement and loads the data to the target table. You can review the exported data by downloading the output file from the Process Details. The database column names for the write-back are specified as the column headers in the generated file as shown below.	Run the Integration

EPM Integration Agent Context Functions

Describes the context function available for important contextual information for script writers.

EPM Integration Agent API, context functions are available that provide important contextual information for script writers.

The contextual functions are as follows:

Copyright © 2017, 2026, Oracle and/or its affiliates.

Function	Description
`agentContext["JOBTYPE"]`	Provides the job type being executed by the agent. At this time, the `JOBTYPE` processes are "`EXTRACT`" and "`DRILL`". When performing a drill-through operation, the agent must execute the query instead of using code to run a custom query. When running an extract, it is possible to use a script to directly execute a query, create an output file, and then save that file to the data directory that will ultimately get picked up by the agent for upload to the Oracle Fusion Cloud Enterprise Performance Management. (This also requires the use of the `skipAction()` api.) The drill operation requires that the agent run the query using the defined connection, and does not pickup a file from the data directory. Because of this, if you want to use a different query, you need to use the `agentAPI.UpdateQuery()` method to update the query, and use the `setCustomConnection()` api, if needed, to update the agent connection information. The agent then passes the query output to the cloud in a JSON payload instead of a file.
`agentContext["EPM_APP_DATA_HOME"]`	Provides the data home specified in the INI file used when starting the EPM Integration Agent.
`agentContext["DELIMITER"]`	Provides the file delimiter specified as part of the data source entry from the Cloud EPM.
`agentContext["DATAFILENAME"]`	Provides the path and name of the file that is uploaded to the Cloud EPM. Use this function instead of building file names manually.
`agentContext["JOBID"]`	Provides the job ID from the integration submitted in the Cloud EPM.
`agentContext["INTEGRATION"]`	Provides the name of the INTEGRATION being executed.
`agentContext["WRITEBACK_DATA_FILE"]`	Provides the path and name of the write-back data export file that is that is downloaded from the Cloud EPM. Use this function instead of building file names manually.
`agentContext["LOCATION"]`	Provides the location from the integration submitted in the Cloud EPM.
`agentContext["SOURCE_APPLICATION"]`	Provides the source application from the integration submitted in the Cloud EPM.
`agentContext["TARGET_APPLICATION"]`	Provides the target application from the integration submitted in the Cloud EPM.

EPM Integration Agent Script Example

Describes how to shows how to call an external API that provides exchange rates, and then prepares the data for upload in a format that can be processed by way of an integration defined in the Data Exchange section of the Oracle Fusion Cloud Enterprise Performance Management.

The following script example shows how to call an external API that provides exchange rates, and then prepares the data for upload in a format that can be processed by way of an integration defined in the Data Exchange section of the Cloud EPM. The setup steps in the Cloud EPM use an agent instance as the data source for an integration with an EPM

Copyright © 2017, 2026, Oracle and/or its affiliates.

Application as a target. Note that this script is provided only as an example, and it is not warranted against defects, and users may not file a service request with Oracle support in regards to any questions or issues related to the script.

```
''' This jython script calls an external API to get exchange rates, and then
generates a file which is picked up by the EPM Integration Agent '''

import json
import urllib2

''' Turn off SQL processing by AGENT '''

agentAPI.skipAction('true')

''' Set Proxy for HTTP call. Needed when connected via VPN '''

proxy = urllib2.ProxyHandler({'http': 'www-proxy.example.com:80' 'https':
'www-proxy.example.com:80'})
opener = urllib2.build_opener(proxy) urllib2.install_opener(opener)

''' Set up URL for rates download.  Please see the URL for additional
information in regards to options. '''

currency = 'USD'
ratesurl = 'https://api.exchangeratesapi.io/latest?base=' + currency
fxrates = urllib2.urlopen(ratesurl)
text = json.loads(fxrates.read())
allrates = text['rates']

agentAPI.logInfo("Jython Script - RateExtract: URL - " + str(ratesurl))

''' Generate file for loading into the EPM Cloud '''

outfilename = agentContext["DATAFILENAME"]
outfile = open(outfilename, "w")

''' Generate header row '''

outfile.write("Account,Currency,Entity,From Currency,Scenario,View,Rate" +
chr(10))

''' Generate a row for each rate '''

for toCur,toRate in allrates.iteritems():
    mystr = "Ending Rate" + "," + str(toCur) + "," + "FCCS_Global
Assumptions" + "," + "FROM_" + str(currency) + "," + "Actual" + "," +
"FCCS_Periodic" + "," + str(toRate) + chr(10)
    outfile.write(mystr)

outfile.close()

agentAPI.logInfo("Jython Script - RateExtract: Output File Name - " +
str(outfilename))
```


Copyright © 2017, 2026, Oracle and/or its affiliates.

19

Reports Execution

Data Integration provides prebuilt reports that capture business-critical operations and revenue-generating activities within your organization. These reports provide key information on how data is integrated from the Enterprise Resource Planning (ERP) source system into the Oracle Fusion Cloud Enterprise Performance Management target application.

Accessing Reports

To access reports that can be executed:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Other** actions drop-down, select **Report Executions**.
2. From **Report Group**, select the type of report to execute:

   Available reports:

   - All
   - Process Monitor Reports
   - Check Reports
   - Variance Reports
   - Base Trial Balance Reports
   - Listing Reports
   - Audit Reports
   - Location Analysis Reports
3. Click
   ▶
   to the right of the report to execute.
4. On the **Generate Report** page, select any report-specific parameters.

   Parameters common to all reports include:

Table 19-1 Report Execution Parameter/Command Descriptions

Parameter/Command	Description
Report Format	Select the output format of the report to be executed. Available report format option: • PDF • HTML • Excel • XLSX
Execution Mode	Select how the report is to be executed. Available options: • online—Processes the report immediately. • offline—Runs the report in the background. (recommended if there are a lot of rows).

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table 19-1 (Cont.) Report Execution Parameter/Command Descriptions

Parameter/Command	Description
OK	Run the report to be executed.
Create Report Script	Creates the Data Integration report and stores it in the `outbox/reports` folder. The generated report is named based on the ID of the Data Integration job that generates the report and the report format. For example, if the report job ID is 2112 and the report output format that you specify is PDF, the report name is `2112.pdf`. The report name is displayed in the console after the report is generated. You can also identify the report name from the Process Details tab in Data Integration or by using the EPM Automate listFiles command. You can also use the EPM Automate downloadFile command to download the report to a local computer.

Audit Reports

An audit report displays all transactions for all locations that compose the balance of a target account. The data returned in this report depends on the location security assigned to the user.

Account Chase - Freeform (Target Account,Period,Category)

Shows one imported account for all Data Integration locations; subtotaled by the Data Integration location.

Runs for

All Data Integration locations

Parameters

Target account, Period, Category,Location

Query

Account Chase Freeform

Template

Account Chase - Free Form.rtf

Common Report Execution Parameters/Commands

For a description of parameters and commands common to all reports, see: Reports Execution.

Account Chase - Wildcard (TargAcct, Per, Cat)

Shows imported accounts for all Data Integration locations, subtotaled by Data Integration location, based on an account selection that enables use of wildcards.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Runs for

All Data Integration locations

Parameters

Target account, Period, Category, Location

Query

Account Chase Wildcard

Template

Account Chase WildCard.rtf

Common Report Execution Parameters/Commands

For a description of parameters and commands common to all reports, see: Reports Execution.

Intersection Drill Down (Per,Cat)

Shows target accounts and amounts; and includes drill-down list of source accounts and amounts that are mapped to target accounts.

Runs for

Current Data Integration location

Parameters

Period, Category, Location

Query

Intersection Drill Down Query

Template

Intersection Drill Down.rtf

Common Report Execution Parameters/Commands

For a description of parameters and commands common to all reports, see: Reports Execution.

Map Monitor for Location

Displays a list of mapping changes, additions, and deletions made for a location based on a date range. Shows the user name, and the time and date of the change.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> The Map Monitor reports do not capture historical data earlier than release 11.1.2.4.100.
> Map Monitor reports are enabled only if the Enable Map Audit is set to "Yes" in System Settings.

Map Monitor reports includes integration mappings from:

- Integration mapping option
- Text file imports
- Migration imports
- Oracle Data Relationship Management

Runs for

All Data Integration locations

Parameters

Location, Action Type (Add, All, Change, Delete), Start Date, End Date

Query

Dimension Map Query

Template

Dimension Map for POV.rtf

Common Report Execution Parameters/Commands

For a description of parameters and commands common to all reports, see: Reports Execution.

Map Monitor for User

Displays a list of mapping changes, additions, and deletions made by a user based on a date range. Report shows the user name, and the time and date of the change.

> **Note**
>
> The Map Monitor reports do not capture historical data earlier than release 11.1.2.4.100.
> Map Monitor reports are enabled only if the Enable Map Audit is set to "Yes" in System Settings.

Map Monitor reports includes data load mappings from:

- Integration mapping option
- Text file imports

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Migration imports
- Oracle Data Relationship Management

Runs for

All Data Integration locations

Parameters

User name, Action Type (Add, All, Change, Delete), Start Date, End Date

Query

Dimension Map for POV

Template

Dimension Map for POV.rtf

Common Report Execution Parameters/Commands

For a description of parameters and commands common to all reports, see: Reports Execution.

Base Trial Balance Reports

The Trial Balance reports provide detail on how source data is processed in Data Integration. Typically, the Trial Balance is used to display account balances in the General ledger system. As data is loaded from the source General Ledger system to the target Oracle Fusion Cloud EPM application, you can validate and compare the balances loaded with the source Trial Balance amounts.

TB Current Location, by Target Acct (Cat, Per)

Shows imported accounts subtotaled by target accounts.

Runs for

Current Data Integration location

Parameters

Category, Location, Integration Name, Period

Query

Trial Balance Current Location Sorted By Target Account

Template

TB With Target Account.rtf

Common Report Execution Parameters/Commands

For a description of parameters and commands common to all reports, see: Reports Execution.

Copyright © 2017, 2026, Oracle and/or its affiliates.

TB Current Location, By Target Entity-Acct (Cat, Per)

Shows all imported records with all dimensions and their respective targets; grouped by target entity and account.

Runs for

Current Data Integration location

Parameters

Category, Period, Location, Integration Name

Query

Trial Balance Base Transaction Currency

Template

Base Trial Balance (All Dimensions with Targets).rtf

Common Report Execution Parameters/Commands

For a description of parameters and commands common to all reports, see: Reports Execution.

TB Current Location with Integration (Cat, Per)

Shows imported source accounts (departments) and the mapping entity rule (map wildcard) in which the accounts are included.

Runs for

Current Data Integration location

Parameters

Category, Period, Location, Integration Name

Query

TB Location With Query

Template

TB Location with Rules.rtf

Common Report Execution Parameters/Commands

For a description of parameters and commands common to all reports, see: Reports Execution.

TB Current Location, with Targets (Cat, Per)

Shows imported target accounts (departments) and their corresponding accounts (entities).

Copyright © 2017, 2026, Oracle and/or its affiliates.

Runs for

Current Data Integration location

Parameters

Category, Accounting Period, Location, Integration Name

Query

Current Trial Balance With Location with Targets

Template

TB Location With Targets.rtf

Common Report Execution Parameters/Commands

For a description of parameters and commands common to all reports, see: Reports Execution.

TB All Columns (Per,Cat,Loc)

Shows all imported records with all source and target columns.

Runs for

Period, Category, Location, Currency

Parameters

Period, Category, Location, Integration Name

Query

TB All Columns

Template

TB All Columns.rtf

Common Report Execution Parameters/Commands

For a description of parameters and commands common to all reports, see: Reports Execution.

TB All Source Columns (Per,Cat)

Shows all imported records with all source columns.

Runs for

Period, Category, Location, Currency for all source columns.

Parameters

Period, Category, Location, Integration Name

Copyright © 2017, 2026, Oracle and/or its affiliates.

Query

All Source Columns

Template

All Source Columns.rtf

Common Report Execution Parameters/Commands

For a description of parameters and commands common to all reports, see: Reports Execution.

TB All Target Columns (Per,Cat)

Shows all imported records with all target columns.

Runs for

Period, Category, Location, Integration Name for all target columns

Parameters

Period, Category, Location, Integration Name

Query

Period, Category, Location, Integration Name

Template

All Target Columns.rtf

Common Report Execution Parameters/Commands

For a description of parameters and commands common to all reports, see: Reports Execution.

TB Converted Current Loc by Target Entity/Account

Shows imported accounts and entities in addition to original and converted accounts: subtotaled by target entity.

Runs for

Current Data Integration location

Parameters

Category, Accounting Period, Location, Integration Name

Query

Trial Balance Converted by Target Entity/Account Query

Template

TB Converted Current Location by Target Entity Account.rtf

Copyright © 2017, 2026, Oracle and/or its affiliates.

Common Report Execution Parameters/Commands

For a description of parameters and commands common to all reports, see: Reports Execution.

TB Current Loc, All Dimensions-Targets, by Target Entity-Account (Cat, Per)

Shows all imported records with all dimensions and their respective targets: grouped by target entity and account.

Runs for

Current Data Integration location

Parameters

Category, Period, Location, Integration Name

Query

Trial Balance Current Location with Targets

Template

TB/(All Dimensions with Targets) by Target Entity Account.rtf

Common Report Execution Parameters/Commands

For a description of parameters and commands common to all reports, see: Reports Execution.

TB Current Loc, All Dimensions-Targets (Cat, Per)

Shows all imported records with all dimensions and their respective targets.

Runs for

Current Data Integration location

Parameters

Category, Period, Location, Integration name

Query

Trial Balance Location All Dimension

Template

TB with Transaction Currency.rtf

Common Report Execution Parameters/Commands

For a description of parameters and commands common to all reports, see: Reports Execution.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Common Report Execution Parameters/Commands

For a description of parameters and commands common to all reports, see: Reports Execution.

Process Monitor Reports

The Process Monitor Reports shows locations and their positions within the data conversion process. You can use the process monitor report to monitor the status of the closing processes. The report is time-stamped. Therefore, it can be used to determine to which locations at which time data was loaded.

Process Monitor (Cat, Per)

Shows all locations and their current status (import, validate, export, load, or check). (Locations are displayed alphabetically.)

Runs for

All Data Integration locations

Parameters

Category, Accounting Period, Location

Query

Process Monitor

Template

Process Monitor.rtf

Common Report Execution Parameters/Commands

For a description of parameters and commands common to all reports, see: Reports Execution.

Process Monitor All Categories (Per)

Shows a list of all locations for every category based on an Accounting Period.

Runs for

All Data Integration categories and locations

Parameters

Accounting Period

Query

Process Monitor All Categories

Template

Process Monitor All Category.rtf

Copyright © 2017, 2026, Oracle and/or its affiliates.

Common Report Execution Parameters/Commands

For a description of parameters and commands common to all reports, see: Reports Execution.

Process Status Period Range (Cat, Start Per, End Per)

Shows a list of all locations and the current load status based on start and end periods for every category.

Runs for

All Data Integration categories and locations

Parameters

Category, Start Period, End Period, Location

Query

Process Status Period Range

Template

Process Status Period Range.rtf

Common Report Execution Parameters/Commands

For a description of parameters and commands common to all reports, see: Reports Execution.

Check Reports

Check reports provide information on the issues encountered when data load rules are run. Note that Check reports return target system values that include aggregation or calculations from the target system.

Note the following when using check reports:

- When the check report is run and opened from the Workbench, it is not saved to the Data Integration folder on the server.
- When you run an integration, a check rule report is not generated automatically. In this case, run the integration before executing the check report.
- If you run the report in offline mode, the report is saved to the `outbox` on the Data Integration server.

Check Report

Shows the results of the validation rules for the current location (indicates pass or fail status).

Runs for

Current Data Integration location

Copyright © 2017, 2026, Oracle and/or its affiliates.

Parameters

Period, Location and Category

Query

Check Report

Template

Check Report.rtf

Common Report Execution Parameters/Commands

For a description of parameters and commands common to all reports, see: Reports Execution.

Check Report By Validation Entity Seq.

Shows the results of the validation rules for the current location (indicates pass or fail status); sorted by the sequence defined in the validation entity group.

Runs for

Current Data Integration location

Parameters

Category, Period, Location

Query

Check Report By Validation Entity

Template

Check Report By Validation Entity Sequence.rtf

Common Report Execution Parameters/Commands

For a description of parameters and commands common to all reports, see: Reports Execution.

Check Report Period Range (Cat, Start Per, End Per)

Shows the results of the validation rules for selected periods.

Runs for

Current Data Integration location

Parameters

Category, Start Period, End Period, Location

Query

Check Report Within Period Query

Copyright © 2017, 2026, Oracle and/or its affiliates.

Template

Check Report With Period Range.rtf

Common Report Execution Parameters/Commands

For a description of parameters and commands common to all reports, see: Reports Execution.

Check Report With Warnings

Shows the results of the validation rules for the current location (warnings are recorded in validation rules).

Runs for

Current Data Integration location

Parameters

Category, Period, Location

Query

Check Report With Warning

Template

Check Report With Warning.rtf

Common Report Execution Parameters/Commands

For a description of parameters and commands common to all reports, see: Reports Execution.

Listing Reports

Listing reports summarize data and settings (such as the import format, or check rule) by the current location.

Import Formats by Location

Displays a list of all import formats; sorted by Data Integration location.

Runs for

N/A

Parameters

None

Query

Import Format By Location

Copyright © 2017, 2026, Oracle and/or its affiliates.

Template

Import Format by Location.rtf

Common Report Execution Parameters/Commands

For a description of parameters and commands common to all reports, see: Reports Execution.

Location Listing

Shows a list of all mapping rules for a selected period, category, or dimension.

Runs for

Current Data Integration location

Parameters

None

Query

Location Listing Query

Template

Location Listing.rtf

Common Report Execution Parameters/Commands

For a description of parameters and commands common to all reports, see: Reports Execution.

Location Analysis

Location Analysis reports provide dimension mapping by the current location.

Dimension Map (Dimension)

Displays a list of all mapping rules for a selected dimension by location.

Runs for

Current Data Integration location

Parameters

Current Data Integration dimension, location

Query

Dimension Map

Template

Dimension Map.rtf

Copyright © 2017, 2026, Oracle and/or its affiliates.

Common Report Execution Parameters/Commands

For a description of parameters and commands common to all reports, see: Reports Execution.

Dimension Map For POV (Dimension, Cat, Per)

Shows a list of all mapping rules for a selected period, category, or dimension.

Runs for

Current Data Integration location

Parameters

Any Data Integration Dimension, Period, Category, Location

Query

Dimension Map for POV

Template

Dimension Map.rtf

Common Report Execution Parameters/Commands

For a description of parameters and commands common to all reports, see: Reports Execution.

Variance Reports

The Variance reports display source and trial balance accounts for one target account, showing data over two periods or categories.

Account Chase Variance (TargAcc,Cat1,Per1,Cat2,Per2)

Displays source input accounts for one target input account, showing variances over two periods or categories.

Runs for

All Data Integration locations

Parameters

Reporting Category, Reporting Period, Comparison Category, Comparison Period, Location, Target Account

Query

Account Chase Variance

Template

Account Chase Variance.rtf

Copyright © 2017, 2026, Oracle and/or its affiliates.

Common Report Execution Parameters/Commands

For a description of parameters and commands common to all reports, see: Reports Execution.

Trial Balance Variance (Cat1,Per1,Cat2,Per2)

Shows source input accounts, subtotaled by target accounts, showing variances over two periods or categories.

Runs for

Current Data Integration location

Parameters

Reporting Category, Reporting Period, Comparison Category, Comparison Period, Location, Target Account

Query

Trial Balance Variance

Template

TB Variance.rtf

Common Report Execution Parameters/Commands

For a description of parameters and commands common to all reports, see: Reports Execution.

Copyright © 2017, 2026, Oracle and/or its affiliates.

20

System Maintenance Tasks

You can run system processes to maintain and clean up all runtime artifacts, such as the Process tables, Staging tables or `inbox`/`outbox` folders. Often the tables and folders contain vast amounts of data, which you may no longer need. With the System Maintenance Tasks feature, you can purge standard tables and folder by scheduling system processes and executing them in the EPM Platform Job Scheduler.

System Maintenance Tasks include:

- Partition Data Load Staging Tables
- Delete Integration
- Export Setup Table Data
- List Table Rowcount
- Maintain Application Folder
- Maintain Data Table by Application
- Maintain Data Table by Location
- Purge All Imported Data
- Maintain Process Tables
- Reallocate Tablespace
- Maintain Setup Data
- Reset Environment
- Upgrade Custom Applications
- Migrate Batches to Pipeline
- Executing a Snapshot Export Job
- Executing a Snapshot Import Job

System Maintenance Tasks can be accessed directly from Data Integration to perform system maintenance locally.

You can also select System Maintenance Task parameters using the Job Scheduler. For more information, see Scheduling Maintenance Scripts in the Job Scheduler.

Partition Data Load Staging Tables

Select a table partitioning option when loading large volumes of data.

Use the Partition Data Load Staging Table script to select a table partitioning option when loading large volumes of data. Table partitioning addresses inconsistent data load times, and enables table partitioning to isolate each data load process in a table partition.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> Run the **Partition Data Load Staging Tables** script only when no other data load processes are running.

To partition data load staging tables:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Other** actions drop-down, select **System Maintenance Tasks**.
2. From **System Maintenance** page, click ▶ to the right of the **Partition Data Load Staging Tables** task to schedule.
3. From the **Run Script: Partition Data Load Staging Tables** page, and then from **Mode**, select the partition table option.

   Available options:

   - **Enable**—Classifies a table in the cloud version of the Oracle Autonomous Database as "STAGING" and enables partitioning.
   - **Revert**—Turns off table partitioning.
4. Click **OK**.

Delete Integration

Delete an integration without having to delete individual components.

You can delete an integration including the name, import format, location, mappings and any integration definitions created in Data Integration. This option enables you to delete an entire integration without having to delete individual components.

To run the Delete Integration script:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Other** actions drop-down, select **System Maintenance Tasks**.
2. From **System Maintenance** page, click ▶ to the right of the **Delete Integration** task.
3. From the **Run Script: Delete Script** page, **Integration Name**, specify the name of the integration to delete
4. From the **Delete Smart Split Integrations Only** drop-down, select **Yes** to delete integrations created with the Smart Split option.

   A Smart Split job type enables you to define smaller data slices without having to create multiple integrations.

   For more information, see Using an Integration with Smart Split Job Type.
5. Click **OK**.

Export Setup Table Data

Export data from the Data Integration setup tables.

Use the Export Setup Table Data option to export the data from the following tables for debugging purposes:

Copyright © 2017, 2026, Oracle and/or its affiliates.

- AIF_BALANCE_RULES
- AIF_BAL_RULE_PARAMS
- AIF_LCM_ARTIFACTS_V
- AIF_SOURCE_SYSTEMS
- AIF_TARGET_APPLICATIONS
- AIF_TARGET_APPL_DIMENSIONS
- AIF_TARGET_APPL_PROPERTIES
- TPOVPARTITION
- TBHVIMPGROUP
- TBHVIMPITEMFILE
- TPOVPERIOD
- TPOVPERIODADAPTOR
- TPOVPERIODSOURCE
- TPOVCATEGORY
- TPOVCATEGORYADAPTOR

To run the Export Setup Table Data script:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Other** actions drop-down, select **System Maintenance Tasks**.
2. From **System Maintenance** page, click ▶ to the right of the **Export Setup Table Data** task.
3. From the **Export Setup Table Data** page, review the job parameters and click **OK**.

List Table Rowcount

Show the number of rows in a Data Integration data table.

Use the List Table Rowcount option to show the number of rows in a data table.

This option lists table row counts from the following tables:

- TDATASEG
- TDATASEG_T
- TDATAMAP
- TDATAMAPSEG
- TLOGPROCESS
- TPOVPARTITION

To run the List Table Rowcount script:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Other** actions drop-down, select **System Maintenance Tasks**.
2. From **System Maintenance** page, click ▶ to the right of the **List Table Rowcount** task.
3. From the **List Table Rowcount** page, review the job parameters and click **OK**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Maintain Application Folder

Purge files from the `inbox`, `outbox`, and `data` folder directories enabling you to specify a separate Days to Keep parameter for each of the folders.

The Maintain Application Folder process purges files from the `inbox`, `outbox`, and `data` folder directories. Data Integration accepts a separate Days to Keep parameter for each of the folders. If the value is not specified for a specific folder, Data Integration skips the folder.

By default, the system purges all files after 60 days. Logs are purged after 7 days.

In addition, Data Integration checks the `inbox` and `outbox` subdirectories under the respective folders and deletes any files. In the `data` subdirectory, Data Integration skips the `scripts` directory because it holds customer scripts.

To run the Maintain Application folder script:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Other** actions drop-down, select **System Maintenance Tasks**.
2. From **System Maintenance** page, click ▶ to the right of the **Maintain Application Folder** task.
3. From the **Run Script: Maintain Application Folder** page, and then from the **Target Application** drop-down, select the target application to which to purge files.
4. In **Days to keep Inbox directory**, specify the number of days to keep files in the Inbox directory.
5. In **Days to keep Outbox directory**, specify the number of days to keep files in the Outbox directory
6. In **Days to keep Data directory**, specify the number of days to keep files in the Data directory.
7. Click **OK**.

Maintain Data Table by Application

Purge archives of exported data for audit and drill by deleting the following tables:TDATAMAPSEG, TDATASEG, TPROCESSLOG.

The Data Integration data tables store archives of exported data for audit and drill down purposes. These tables can grow over a period of time but you can purge these tables as needed. The maintenance process deletes the following tables:

- TDATAMAPSEG
- TDATASEG
- TPROCESSLOG

The parameters are:

- Target Application
- Category
- Start Period
- End Period

Copyright © 2017, 2026, Oracle and/or its affiliates.

To run the Data Table by Application script:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Other** actions drop-down, select **System Maintenance Tasks**.
2. From **System Maintenance** page, click ▶ to the right of the **Maintain Data Table by Application** task.
3. From the **Run Script: Maintain Data Table by Application** page, and then from the **Target Application** drop-down, select the target application to which to purge files.
4. From **Start Period**, select the starting period from which to delete data.
5. From **End Period**, select the ending period from which to delete data.
6. From **Category**, select the category data to delete.

   To delete all category data, leave blank.
7. Click **OK**.

Maintain Data Table by Location

Delete data for a specific location by period or category.

Use Maintain Data Table by Location to delete data for a specific location by period or category.

> **Note**
>
> To delete all locations for an application, use the Maintain Data Tables by Application option. This option enables you to delete data across all locations associated with a selected target application. For more information, see Maintain Data Table by Application.

Maintain Setup Data uses a Mode parameter, which enables you to either preview or delete invalid data.

The parameters are:

- Location
- Start Period
- End Period
- Category

> **Note**
>
> If you need to delete multiple period mapping but you are unsure which integrations include period mappings, leave Location and Category blank if you want to delete all the period mappings.

To run the Maintain Data Table by Location task:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Other** actions drop-down, select **System Maintenance Tasks**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

2. From **System Maintenance** page, click ▶ to the right of the **Maintain Data Table by Location** task.
3. From the **Run Script: Maintain Data Table by Location** page, and then from the **Location** drop-down, select the location from which to delete data.

   To delete data from all locations, leave the **Location** field blank.
4. From **Start Period**, select the starting period from which to delete data.
5. From **End Period**, select the ending period from which to delete data.
6. From **Category**, select the category data to delete.

   To delete all category data, leave this field blank.
7. Click **OK**.

Purge All Imported Data

Clear all imported data from your data tables.

Use Purge All Imported Data to clear all imported data from your data tables. This option enables you to clear data from the system so that you can start over when loading data.

When executed, the Purge All Imported Data deletes the following:

- Mapping audit used for loading
- Audit information about the data load
- Process load status
- Process parameters

> **Note**
>
> There is **no backup** to recover any purged data. It is recommended that you use extreme caution before executing this process.

After the purge, note the following:

- You cannot view any data in Workbench for any POV.
- You cannot drill down from the Oracle Fusion Cloud Enterprise Performance Management to Data Integration.

  > **Note**
  >
  > Drill regions are not deleted as part of the purge process. If you need to delete a drill region, then delete it manually.
- You cannot restore maps for the given POV.

> **Note**
>
> All setup data for example application registration, import format, and mapping are retained and not impacted by the purge process.

Copyright © 2017, 2026, Oracle and/or its affiliates.

To run the Purge All Imported Data script:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Other** actions drop-down, select **System Maintenance Tasks**.
2. From **System Maintenance** page, click ▶ to the right of the **Purge All Imported Data** task.
3. From the **Run Script: Purge All Imported Data** page, and then from **Confirm Delete All Imported Data**, select **Y** (yes) to launch a confirmation message before executing a purge.

   Otherwise, enter **N** (No).
4. In **Reason**, specify the reason for executing the purge.
5. Click **OK**.

   The message: **Custom script executed initiated with Process ID: XXX** is displayed (where XXX is the process id number generated for job). You can access the log from the Process Details page.

Maintain Process Tables

Maintain Process tables.

This process maintains the following execution tables:

- AIF_PROCESSES
- AIF_PROCESS_DETAILS
- AIF_PROCESS_LOGS
- AIF_PROCESS_PARAMETERS
- AIF_PROCESS_PERIODS
- AIF_PROCESS_STEPS
- AIF_BAL_RULE_LOADS
- AIF_BAL_RULE_LOAD_PARAMS
- AIF_BATCH_JOBS
- AIF_BATCH_LOAD_AUDIT
- AIF_TEMP

To run the Maintain Process Tables task:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Other** actions drop-down, select **System Maintenance Tasks**.
2. From **System Maintenance** page, click ▶ to the right of the **Maintain Process Table** task.
3. From the **Run Script: Maintain Process Tables** page, and then from **Days to keep records**, specify the number of days to keep records in the execution tables.
4. Click **OK**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Maintain Setup Data

Use Maintain Setup Data to view and delete an orphan row or invalid data that cannot be accessed or deleted from the user interface. This data might include an orphan row from an import format, location, or data load rule. It can also include a duplicate row, or a Cloud user id or password defined for a local target application

Maintain Setup Data uses a Mode parameter, which enables you to either preview or delete invalid data..

To run the Maintain Setup Data task:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Other** actions drop-down, select **System Maintenance Tasks**.
2. From **System Maintenance** page, click ▶ to the right of the **Maintain Process Table** task.
3. From the **Maintain Setup Data** page, and then from **Mode**, select either **Preview** or **Delete.**

   The Preview mode enables you to print the data to be deleted. The Delete mode enables you to delete the data.
4. Click **OK**.

Reallocate Tablespace

Free unused and space in your tables enabling you to free up space available for the storage of data.

Use the Reallocate Tablespace maintenance script to free unused and space in your tables. Reallocating tablespace enables you not to exceed the space quota defined for a specific table and free up space available for the storage of data.

> **Note**
>
> Do not load data when executing this script.

To run the Reallocate Tablespace script:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Other** actions drop-down, select **System Maintenance Tasks**.
2. From **System Maintenance** page, click ▶ to the right of the **Reallocate Tablespace** task.
3. Click **OK**.

Reset Environment

Clear all Data Integration setup and data resetting the entire business process environment.

The Reset Environment script enables you to clear all Data Integration setup and data resetting the entire business process environment. You can then reconfigure the integration

Copyright © 2017, 2026, Oracle and/or its affiliates.

process as needed. This process accepts a **Yes** or **No** confirmation parameter and allows you to specify a reason for the reset.

> **Note**
>
> When the Reset Environment process is executed, data cannot be recovered.

To run the Reset Environment task:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Other** actions drop-down, select **System Maintenance Tasks**.
2. From **System Maintenance** page, click ▶ to the right of the **Reset Environment** task.
3. From the **Run Script: Reset Environment** page, and then from **Confirm Delete**, select **Y** (yes) to confirm the reset of the environment.

   Otherwise, enter **N** (No).
4. In **Reason for Delete**, specify the reason for resetting the environment.
5. Click **OK**.

Upgrade Custom Applications

Migrate your existing custom target application to data export to file applications.

Use the Upgrade Custom Applications option to migrate your existing custom target application to data export to file applications. The migration converts existing file format from the custom target application to the file formats used in the data export to file option and retains all the existing setup. When the custom target application has been converted, you can run same data rule as before. You can run the migration for one custom application or all.

Before using this option, reconcile the differences in the file formats. For example, the header row in the Data Export to File contains the name of the dimension and not UD1, UD2 etc.

For more information about the Data Export to File option, see Registering a Data Export File Application.

To run the Upgrade Custom Applications task:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Other** actions drop-down, select **System Maintenance Tasks**.
2. From **System Maintenance** page, click ▶ to the right of the **Upgrade Custom Applications** task.
3. From the **Run Script: Upgrade Custom Applications** page, and then from the **Custom Application** drop-down, select the name of the custom target application to migrate from the LOV.
   To migrate all custom applications, select **All Custom Applications**.
4. Click **OK**.

Migrate Batches to Pipeline

Migrate a batch created in Data Management to the Data Integration Pipeline.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Use the Migrate Batches to Pipeline task to migrate a batch created in Data Management to the Data Integration Pipeline feature.

> **Note**
>
> For more information about the Pipeline Open Batch Job types, see Using Open Batch Job Types.

To run the migrate batches to the Pipeline:

1. From the **Data Integration** home page, and then from the **Actions** menu, then from the **Other** actions drop-down, select **System Maintenance Tasks**.
2. From **System Maintenance** page, click ▶ to the right of the **Migrate Batches to Pipe** task.
3. From the **Run Script: Migrate Batches to Pipe** page, and then from **Batch Name**, specify the name of the batch to migrate.

Executing a Snapshot Export Job

Export all artifacts from setup and data folders including the ID columns to a ZIP file.

The data export process exports all artifacts from setup and data folders including the ID columns to a ZIP file. Artifacts are exported by POV and retain the ID of the setup artifact when data is exported and then imported to the target environment.

The snapshot export options supports: All, All Incremental, Incremental, and Setup export types.

To execute a snapshot export:

1. Click **Application**, then **Jobs**, and then **Schedule Jobs**.
2. From the **Schedule Jobs** page, then **What type of job is this?**, select **Integration Maintenance Script**.
3. To schedule an integration maintenance script to run now, see Scheduling Hourly Jobs .

   To schedule an integration maintenance script to run later, see Scheduling a Data Rule or Integration Definition to Run Later in the Job Scheduler.
4. From the **Integration Maintenance Script** page, click ✔ to the right of the **Execute Snapshot Export** task to schedule.
5. On the **Run Script: Execute Snapshot Export** page, from **Snapshot Type**, select the type of snapshot to export.

   Available snapshot types:

   - All—Includes all setup and historical artifacts.
   - All Incremental—Include only new or changed historical artifacts based on the POV since the last snapshot was exported and include setup artificats and all POVs (old and new) in the output file.
   - Incremental— Include only new or changed historical data based on the POV since the last snapshot was exported and include only setup artifacts and new POVs in the output file.
   - Setup—Includes only setup artifacts.

Copyright © 2017, 2026, Oracle and/or its affiliates.

6. In **Snapshot File Name**, specify the file name of the ZIP with the ZIP extension to be written to the *outbox* for the snapshot export.

   For example, you might type: **DMSnapshot_082021.zip**.

   If a file doesn't end with a ZIP extension, the system forces a ZIP extension by appending a **.zip** suffix at the end of the file name.

7. In **Overwrite Output File**, specify **Yes** to overwrite a ZIP that has the same name as the one you are creating. Otherwise, specify **No**.

   > **Tip**
   >
   > If you specify an existing output file and specify **No** to overwrite the existing output file (ZIP), the job fails and the system shows the message: "Rule Execution did not complete successfully."

8. Click **OK**.

   One Execute Snapshot Exports job can be executed at a time.

   The message: **Custom script executed initiated with Process ID: XXX** is displayed (where *XXX* is the process id number generated for the job).

   You can access the job log from the Process Details page in Data Integration.

   You can view the snapshot export ZIP after the snapshot export has completed by downloading the output ZIP in Process Details.

   You can also download the snapshot ZIP using the EPM Automate downloadFile command.

Executing a Snapshot Import Job

Truncates all the setup and data artifacts and then import them.

During the snapshot data *import* process, the system first truncates all the setup and data artifacts and then imports them. Additionally, database sequences are reset based on the imported data so that no conflicts occur after the data is restored.

To execute a snapshot import:

1. Upload the snapshot export ZIP to the *inbox* in Data Integration.

   To upload the snapshot export ZIP to the *inbox* in Data Integration, use the EPM Automate uploadFile command.

   You can also use the File Browser in Data Integration to upload a file to the Data Integration inbox.

2. Click **Application**, then **Jobs**, and then **Schedule Jobs**.

3. From the **Schedule Jobs** page, then **What type of job is this?**, select **Integration Maintenance Script**.

4. To schedule an integration maintenance script to run now, see Scheduling Hourly Jobs.

   To schedule an integration maintenance script to run later, see Scheduling a Data Rule or Integration Definition to Run Later in the Job Scheduler.

5. From the **Integration Maintenance Script** page, click to the right of the **Execute Snapshot Export** task to schedule.

Copyright © 2017, 2026, Oracle and/or its affiliates.

6. On the **Run Script: Execute Snapshot Import** page, from **Snapshot File Name**, specify the path and file name of the ZIP to be executed.

   For example, you might type: **inbox/DMSnapshot_082021.zip**.

7. Click **OK**.

   Only one Execute Snapshot Import job can be executed at a time.

   The message: **Custom script executed initiated with Process ID: 0**.

   You can access the job log from the Process Details page.

> **Note**
>
> The Process ID for the snapshot import is always " 0" only when the file is exported in ALL, INCREMENTAL or ALL_INCREMENTAL modes because each of these three exports contain data artifacts and process details are part of data tables.
>
> The Process ID is not "0" when only setup data is exported. This is because setup data doesn't contain data tables (Process details). In this case, the next available sequence in the Process ID is used.

Copyright © 2017, 2026, Oracle and/or its affiliates.

21

Predictive Cash Forecasting and Data Integration

Predictive Cash Forecasting is a Planning application type designed to help treasurers and cash managers perform short-term tactical (rolling ~10 days) or mid-term operational (~3-6 months /~12-26 weeks) cash forecasting. It allows for daily, weekly, or monthly rolling forecasts that can be generated for operational, financial, and investing cash flow line items. It is built using a direct cash flow method and enables decision-making and actions for cash optimization across multiple legal entities within the business.

Pipelines in Data Integration are used to manage the metadata and data needed to support predictive cash forecasting in EPM.

For information about using Predictive Cash Forecasting with Data Integration, see the following sections in the *Administering Predictive Cash Forecasting for Early Adopters* guide available on docs.oracle.com:

- Predictive Cash Forecasting Overview section
- Configuring Custom Dimensions in Data Integration

Below are some useful Data Integration links when using Predictive Cash Forecasting:

- Mapping Dimensions
- Refreshing Metadata
- Using Target Expressions
- Pipeline Process Description
- Using an Import Metadata Job Type

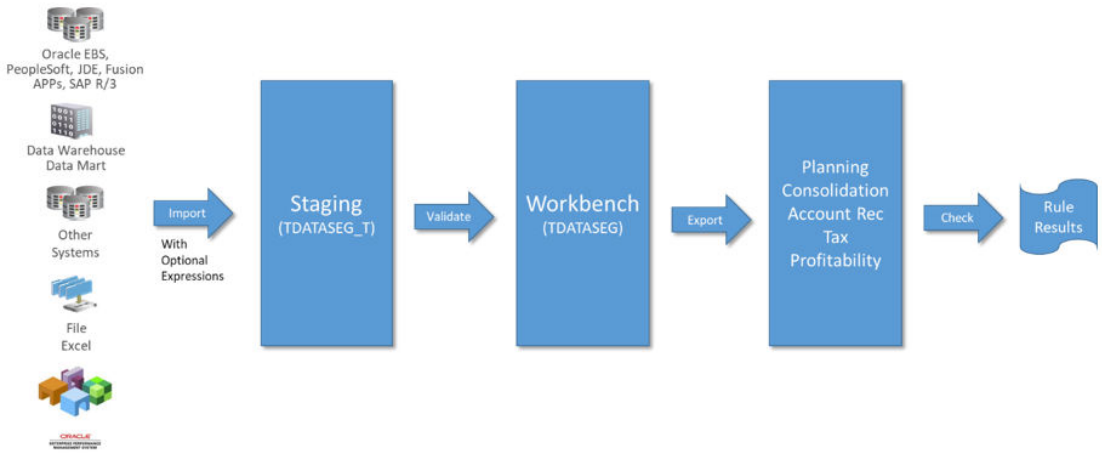
Copyright © 2017, 2026, Oracle and/or its affiliates.

A

Performance Tuning in Data Integration

Data Integration uses a multi-step workflow to transform and load data into Oracle Fusion Cloud Enterprise Performance Management business processes. Each step in the workflow is used to control and manage this process, and each step contributes to the overall performance of the entire process. Within Data Integration, user-selectable options are provided to adjust the workflow and transformation steps to meet customer performance requirements.

At a high level the workflow process is as follows:



Tuning the full process requires the analysis of each step to determine the required level of detail, data volumes, data transformations, and drill-through requirements. These factors drive the options selected and configured in Data Integration to deliver the desired result. The examples in the steps below use the full workflow mode as the workflow type and the load method of "Numeric Data Only."

Job Log Analysis

The starting point for any tuning exercise is the job log. When the integration finishes, the job log may be downloaded from the execution dialog, or from the Process Details page. To effectively analyze the job log, it's important to understand each of the Data Integration process steps including:

- Step 1: Stage and load file to the TDATASEG_T table
- Step 2: Clean up prior data loads
- Step 3: Process Mapping Rules
- Step 4: Clean up prior integrations in TDATASEG
- Step 5: Copy mapping results from TDATASEG_T to TDATASEG
- Step 6: Validate Mapping Results
- Step 7: Generate File and Load to Target

  Additional job log information is available at: Job Log Additional Considerations

Copyright © 2017, 2026, Oracle and/or its affiliates.

Step 1: Stage and Load File to the TDATASEG_T table

This step is used to initialize the system for processing, and to load the source data into the temporary table used for mapping. Sections of a sample log file for a million row data file are provided below:

```
2021-07-10 22:33:16,523 INFO  [AIF]: FDMEE Process: 51, Log Level: 5
2021-07-10 22:33:16,523 INFO  [AIF]: Location     : PartnerTraining01-MM (Partitionkey:14)
2021-07-10 22:33:16,524 INFO  [AIF]: Period Name  : Jan-21 (Period Key:1/31/21 12:00 AM)
2021-07-10 22:33:16,524 INFO  [AIF]: Category Name: MillionRow (Category key:25)
2021-07-10 22:33:16,524 INFO  [AIF]: Rule Name    : PartnerTraining01-MM (Rule ID:13)
2021-07-10 22:33:18,710 DEBUG [AIF]: FDMEE Version: 21.07.04
2021-07-10 22:33:18,710 INFO  [AIF]: Log File Encoding: UTF-8
2021-07-10 22:33:19,273 INFO  [AIF]: -------START IMPORT STEP-------
2021-07-10 22:33:19,395 DEBUG [AIF]: FileData.extractDataFromSource - START
2021-07-10 22:33:19,396 DEBUG [AIF]: CommData.getRuleInfo - START
2021-07-10 22:33:19,402 DEBUG [AIF]:
```

In this example, the process started as 22:33:16,523 in log level 5. When tuning and debugging, it is recommended to use log level 5 to see full details of the processing run. The next figure shows the completion of the load to the TDATASEG_T table:

```
2021-07-10 22:33:21,025 INFO  [AIF]: EPMFDM-140274:Message - File Name Sample_file_1048576_Rows.dat
periodKey2021-01-31
GL PeriodName0
2021-07-10 22:33:21,025 INFO  [AIF]: EPMFDM-140274:Message - Instantiating ExpressionInterpreter
2021-07-10 22:33:21,386 INFO  [AIF]: EPMFDM-140274:Message -  PROCESS ID: 51
 PARTITIONKEY: 14
 IMPORT GROUP: PartnerTraining01-MM
 FILE TYPE: DELIMITED
DELIMITER: ,
SOURCE FILE: Sample_file_1048576_Rows.dat
PROCESSING CODES:
--------------------------------------------------------------------------------
BLANK.............. Line is blank or empty.
NN................. Non-Numeric, Amount field contains non numeric characters.
TC................. Type Conversion, Amount field could not be converted to a number.
ZP................. Zero Suppress, Amount field contains a 0 value and zero suppress is ON.
SKIP FIELD.............. SKIP field value was found
NULL ACCOUNT VALUE.............. Account Field is null
SKIP FROM SCRIPT.............. Skipped through Script
HEADER ROW.............. Header Row
ERROR_INVALID_PERIOD.............. Invalid period specified in the file.



2021-07-10 22:37:32,637 INFO  [AIF]: EPMFDM-140274:Message - Rows Loaded: 1048576
Rows Rejected: 0
```

The log shows that 1,048,576 rows were loaded and 0 rows were rejected. The approximate time for staging and file loading before mapping was 4 minutes and 16 seconds. There isn't anything that you can tune in regard to this step with a file-based load, so this is a fixed component of the overall process. Note that users may specify business rules to run during selected events during the load process, and if you have included business rules, make sure that these are also performing as required. The performance of business rules is not controlled by Data Integration, and these should be tuned in the target application when they are causing performance issues.

Step 2: Clean up Prior Data Loads

This step is used to clean up data in the TDATASEG_T and AIF_PROCESS_PERIODS tables as shown here:

Copyright © 2017, 2026, Oracle and/or its affiliates.

```
2021-07-10 22:37:34,086 DEBUG [AIF]: intermediateCommit: True, skipRegularSql: False
2021-07-10 22:37:34,088 INFO  [AIF]: Executing delete statement:
2021-07-10 22:37:34,088 DEBUG [AIF]:
        DELETE FROM TDATASEG_T
        WHERE LOADID < 51
        AND EXISTS ( SELECT 1 FROM AIF_PROCESSES p WHERE p.RULE_ID = 13 AND p.PROCESS_ID = TDATASEG_T.LOADID )

2021-07-10 22:37:34,092 INFO  [AIF]: Successfully executed delete statement
2021-07-10 22:37:34,095 DEBUG [AIF]:
        DELETE FROM AIF_PROCESS_PERIODS
        WHERE PROCESS_ID < 51
        AND EXISTS ( SELECT 1 FROM AIF_PROCESSES p WHERE p.RULE_ID = 13 AND p.PROCESS_ID = AIF_PROCESS_PERIODS.PROCESS_ID )
```

The data in TDATASEG_T is cleared during daily maintenance window. If you are running the process after a restart, this step completes very quickly, but subsequent loads may be slower. In full workflow mode, the data in TDATASEG_T is deleted at the end of the run. In a simple workflow, it is not deleted at the end of the run. If there are more than a million rows from a previous run, then the system performs a batch delete, which is slower than a single deletion.

Step 3: Process Mapping Rules

The mapping process is where tuning can provide improved results, and where users have control over the speed of the process. The log file shows the mapping rules processed, along with the SQL used to process the mapping rule.

In these examples are some of the wild card * to * mapping rules were used for each dimension, and these are the most time intensive rules that a user can implement. For each * to * rule, the system does a full pass of the data set, so if there is a million row data set, with 5 * to * mapping rules, which turns into a database read of 5 million rows. As row numbers increase, this significantly increases the mapping time. An additional consideration is in regard to the number of periods being processed in a single run, and the mapping process is executed for each period in the data set. This too leads to an increase in the time required to process the full data set.

For this example, the following shows the first mapping rule, which is a * to * rule for the Account dimension:

```
Processing Mappings for Column 'ACCOUNT'
2021-07-10 22:37:34,775 DEBUG [AIF]:
               UPDATE TDATASEG_T
               SET ACCOUNTX = ACCOUNT
                  ,ACCOUNTR = 201
                  ,ACCOUNTF = 3
               WHERE LOADID = 51
               AND PARTITIONKEY = 14
               AND CATKEY = 25
               AND (ACCOUNTX IS NULL OR ACCOUNTX = '')
               AND (1=1)
               AND PERIODKEY = '2021-01-31'
2021-07-10 22:38:07,915 INFO  [AIF]: Data Rows Updated by Location Mapping '10' (LIKE): 1048576
```

This mapping rule completed in approximately 33 seconds, and as the process continues, the like rules process between 35 and 53 seconds each. As you can see, adding additional like mapping rules continues to consume processing time during mapping.

Step 4: Clean up Prior Integrations in TDATASEG

Upon the successful completion of mapping, the system deletes the prior run from the TDATASEG table. (The TDATASEG table is where the data for the workbench is stored.) If this is the first execution of the integration, this step completes quickly because there isn't any prior period data. If there is a large amount of prior data, then this step can potentially take a long time.

Copyright © 2017, 2026, Oracle and/or its affiliates.

As you see below, the batch size parameter is used to delete chunks of data from the TDATASEG table so that the delete process does not hit the processing limit for an individual SQL statement. It's possible to tune this step by adjusting the batch size so that you limit the number of delete statements. For this execution, the batch size was set to 1000, and it could be set higher. On a subsequent execution, this step took 6 minutes to delete the prior load of approximately a million records.

> **Note**
>
> Do not set the batch size to be very large as the data is loaded to memory for processing based on the batch size, and can use up memory. The system memory is common for the Oracle Fusion Cloud Enterprise Performance Management and can impact other areas of the instance.

```
2021-07-10 22:40:22,760 INFO  [AIF]: Executing delete statement using batch method
2021-07-10 22:40:22,763 DEBUG [AIF]: batchSize: 1000
2021-07-10 22:40:22,763 DEBUG [AIF]: SELECT rowid FROM TDATASEG WHERE LOADID < 51 AND PARTITIONKEY = 14 AND CATKEY = 25 AND RULE_ID = 13 AND PERIODKE

2021-07-10 22:40:22,775 INFO  [AIF]: Successfully executed delete statement using batch method
2021-07-10 22:40:22,775 INFO  [AIF]: Number of Rows deleted from TDATASEG: 0
```

Step 5: Copy Mapping Results from TDATASEG_T to TDATASEG

After the prior results have been deleted from the TDATASEG table, the new data that has been mapped is moved into the TDATASEG table for access via the workbench. This is another time consuming step in the data load process, and it too uses the batch size parameter to "chunk" the insert of the mapping results into the TDATASEG table so that the processing limit for a single SQL is not reached.

In this example, the batch size is set to 1000, which resulted in 1049 batches required to move the data. The total time to move the data from the TDATASEG_T table to the TDATASEG table was 5 minutes and 26 seconds.

```
2021-07-10 22:40:23,434 DEBUG [AIF]:

        INSERT INTO TDATASEG (
          DATAKEY, PARTITIONKEY, CATKEY, PERIODKEY,CURKEY, DATAVIEW, CALCACCTTYPE, CHANGESIGN, JOURNALID
          ,AMOUNT, AMOUNTX, AMOUNT_PTD, AMOUNT_YTD, DATA, DATAX, DESC1, DESC2
          ,ACCOUNT, ACCOUNTX, ACCOUNTR, ACCOUNTF, ENTITY, ENTITYX, ENTITYR, ENTITYF, ICP, ICPX, ICPR, ICPF
          ,UD1, UD1X, UD1R, UD1F, UD2, UD2X, UD2R, UD2F, UD3, UD3X, UD3R, UD3F, UD4, UD4X, UD4R, UD4F
          ,UD5, UD5X, UD5R, UD5F, UD6, UD6X, UD6R, UD6F, UD7, UD7X, UD7R, UD7F, UD8, UD8X, UD8R, UD8F
          ,UD9, UD9X, UD9R, UD9F, UD10, UD10X, UD10R, UD10F, UD11, UD11X, UD11R, UD11F, UD12, UD12X, UD12R, UD12F
          ,UD13, UD13X, UD13R, UD13F, UD14, UD14X, UD14R, UD14F, UD15, UD15X, UD15R, UD15F, UD16, UD16X, UD16R, UD16F
          ,UD17, UD17X, UD17R, UD17F, UD18, UD18X, UD18R, UD18F, UD19, UD19X, UD19R, UD19F, UD20, UD20X, UD20R, UD20F
          ,ATTR1, ATTR2, ATTR3, ATTR4, ATTR5, ATTR6, ATTR7, ATTR8, ATTR9, ATTR10, ATTR11, ATTR12, ATTR13, ATTR14
          ,ATTR15, ATTR16, ATTR17, ATTR18, ATTR19, ATTR20, ATTR21, ATTR22, ATTR23, ATTR24, ATTR25, ATTR26, ATTR27
          ,ATTR28, ATTR29, ATTR30, ATTR31, ATTR32, ATTR33, ATTR34, ATTR35, ATTR36, ATTR37, ATTR38, ATTR39, ATTR40
          ,ARCHIVEID, HASMEMOITEM, STATICDATAKEY, LOADID, RULE_ID, CODE_COMBINATION_ID,STAT_BALANCE_FLAG, VALID_FLAG )
         VALUES (1130025,14,25,'2021-01-31 00:00:00.0','[NONE]','YTD',9,0,NULL,11111,11111,NULL,NULL,NULL,NULL,NULL,NULL,'ACCOUNT1','ACCOUNT1',201
2021-07-10 22:40:23,725 DEBUG [AIF]: Executing batch number: 1
2021-07-10 22:40:24,120 DEBUG [AIF]: Executing batch number: 2
2021-07-10 22:40:24,699 DEBUG [AIF]: Executing batch number: 3
2021-07-10 22:40:25,063 DEBUG [AIF]: Executing batch number: 4
...
2021-07-10 22:46:58,103 DEBUG [AIF]: Executing batch number: 1047
2021-07-10 22:46:58,468 DEBUG [AIF]: Executing batch number: 1048
2021-07-10 22:46:58,736 DEBUG [AIF]: Executing partial batch number: 1049
2021-07-10 22:46:58,834 INFO  [AIF]: Successfully executed insert statement using batch method
2021-07-10 22:46:58,834 INFO  [AIF]: Number of Rows inserted into TDATASEG: 1048576
```

After the data is moved to TDATASEG, the data in TDATASEG_T is then deleted. This process also uses the batch size parameter, and in this example, the process to delete the temporary data took 5 minutes and 5 seconds.

Step 6: Validate Mapping Results

The next step is to check the mapping results for missing mappings. This is a one step process and is relatively quick. The system counts the rows where the VALID_FLAG = "Y" and compares that to the total number of rows in the TDATASEG table for the process run, and if

Copyright © 2017, 2026, Oracle and/or its affiliates.

the numbers are the same then it indicates a successful mapping process. Any invalid rows are updated to VALID_FLAG = "N."

```
Validate Data Mappings for Period 'Jan-21'
2021-07-10 22:52:04,389 DEBUG [AIF]:
      SELECT COUNT(*) ROW_COUNT
      FROM TDATASEG
      WHERE LOADID = 51
      AND (PARTITIONKEY = 14 AND CATKEY = 25 AND PERIODKEY = '2021-01-31' AND RULE_ID = 13 AND VALID_FLAG = 'Y'
```

Step 7: Generate File and Load to Target

The last step is to generate the file that is loaded to the target business process. For this process, the time to generate the file, the drill region and then load to the Oracle Fusion Cloud Enterprise Performance Management takes 1 minute and 45 seconds. This step also may include business rules or other calculations in the target application, and once the file leaves Data Integration, any tuning needs to be addressed in the target application.

Job Log Additional Considerations

Note that a selected load method also impacts the performance during the last step of the process when the data is loaded. The "Numeric Data Only" option loads directly to Oracle Essbase and bypasses all validation and security checks and is the quickest way to load data when using Planning.

If users select "All Data Types with Security", this uses the Outline Load Utility (OLU), or importDataSlice API, depending on the user role. The OLU is used when the user is an administrator, and the "Validate Data for Admin " setting is set to "No." The Validate Data option uses the platform API when loading the data to ensure that the data load is valid. The importDataSlice API is used for non-administrator users, or when an administrator is loading, and the "Validate Data for Admin" is set to "Yes." The importDataSlice API performs a complete check of data, and honors all validations and security settings, and is the slowest method.

Tuning Considerations

There are various strategies available to tune integrations, and include:

- SQL Mapping
- Expressions
- Simple Workflow Mode
- Quick Mode

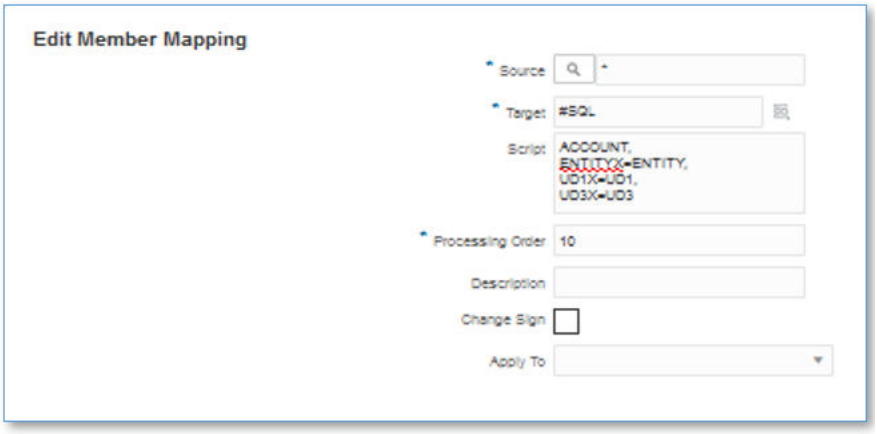
  Additional turning performance information is available at: Additional Considerations

SQL Mapping

The SQL Mapping feature is available to use for complex mapping requirements, and also may be used to replace multiple wildcard * to * mapping rules with a single pass of the database.

In this example, mapping takes approximately 3 minutes, and with a single SQL mapping rule is should only take about 30 seconds. A single SQL mapping rule can be used to replace all of the "like" rules, and would look like the following:

Copyright © 2017, 2026, Oracle and/or its affiliates.



The actual SQL that is generated and executed is as follows:

```
Processing Mappings for Column 'ACCOUNT'
2021-07-15 14:54:34,333 DEBUG [AIF]:
              UPDATE TDATASEG_T
              SET ACCOUNTX = ACCOUNT,
ENTITYX=ENTITY,
UD1X=UD1,
UD3X=UD3
                  ,ACCOUNTR = 201
                  ,ACCOUNTF = 3
              WHERE LOADID = 75
              AND PARTITIONKEY = 14
              AND CATKEY = 25
              AND (ACCOUNTX IS NULL OR ACCOUNTX = '')
              AND (1=1)
              AND PERIODKEY = '2021-01-31'
2021-07-15 14:54:59,646 INFO  [AIF]: Data Rows Updated by Location Mapping '10' (LIKE): 1048576
```

In this case the SQL mapping was defined on the ACCOUNT dimension, and the other * to * mapping rules were deleted. The total time for this one SQL mapping rule was 29 seconds, and no other mapping rules were required.

The ACCOUNT and ENTITY dimension may be referenced by those names, but the other dimensions are mapped to UD dimensions. To find the set of dimensions you need for the SQL mapping, you need to either look at the Application definition, or the log file to see which dimensions to use. In this example Product and Scenario are mapped to UD1 and UD3. The source dimension members use the column without the "X," and the mapped values are in the column with an "X" as a suffix. For the ACCOUNT dimension, the value from the source file is in the column named ACCOUNT, and the mapped value is stored in the ACCOUNTX column. The SQL mapping is used to set the "X" column for each dimension.

This same type of mappings may be used in the Account Reconciliation, but note that the Profile dimension is classified as ACCOUNT, so any SQL mappings for the Profile dimension should be specified on the ACCOUNT dimension. Other dimensions in Account Reconciliation should be referenced based on the mapping defined in the application definition.

Each type of mapping uses resources differently, and the mapping performance is in the following order, where Explicit is the fastest, and Multi-Dim is the slowest:

1. EXPLICIT
2. IN
3. BETWEEN and LIKE
4. MULTI-DIM

Copyright © 2017, 2026, Oracle and/or its affiliates.

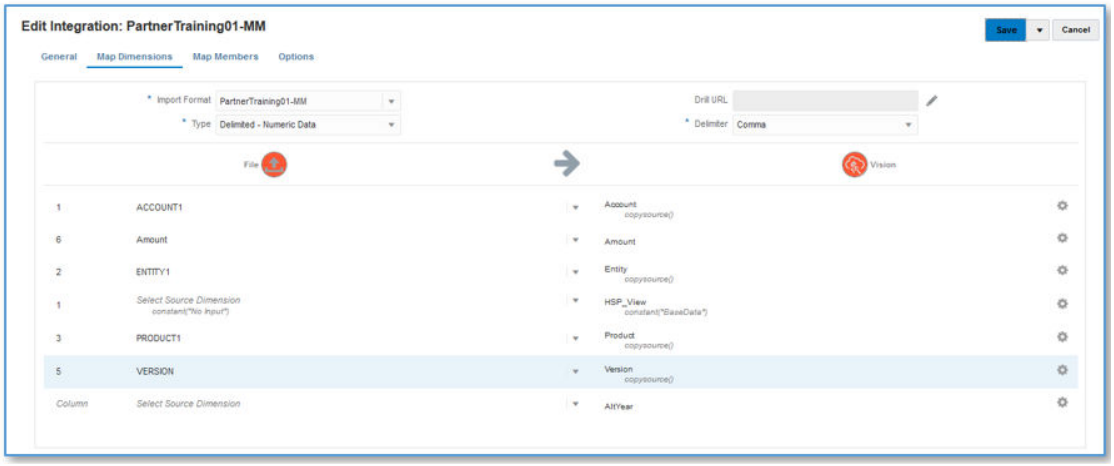
Multi-dim mappings are the slowest mapping, and try to limit multi-dim rules for complex use cases where you need to use a combination of EXPLICIT and LIKE mapping. For example, ENTITY = 100 AND ACCOUNT LIKE 4*.

As an additional tuning strategy, you may be able to replace multi-dim mappings with explicit mappings by combining source dimensions. For example if ENTITY=100 AND ACCOUNT=4100 you can concatenate ENTITY and ACCOUNT as the source, and define an EXPLICIT mapping for 100-4000.

> **Note**
>
> Although the performance is similar when the data volume is very large (greater than 3 million rows), SQL mappings can fail due to database governor limits. Import expressions are processed when the data is imported and does not involve a SQL operation so the import does not fail. For this reason for very large data volume it is recommended to use import expressions instead of SQL mapping.

Expressions

Expressions may also be used instead of mapping rules, and this technique also helps improve performance. To replace the * to * "like" mapping rules, the CopySource expression may be used, and looks like the following:

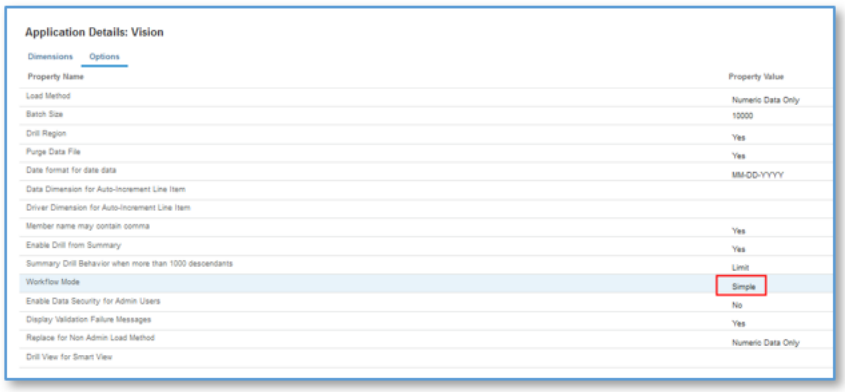


This expression does the same thing as the * to * mapping, and it is applied during import, rather than via a scan of the table with a SQL statement. The performance of expressions is roughly the same as using a single SQL mapping rule, but it is recommended to use expressions when data volume is large so mapping does not fail because of database governor limits. (Expressions are processed during the import step of the load process.)

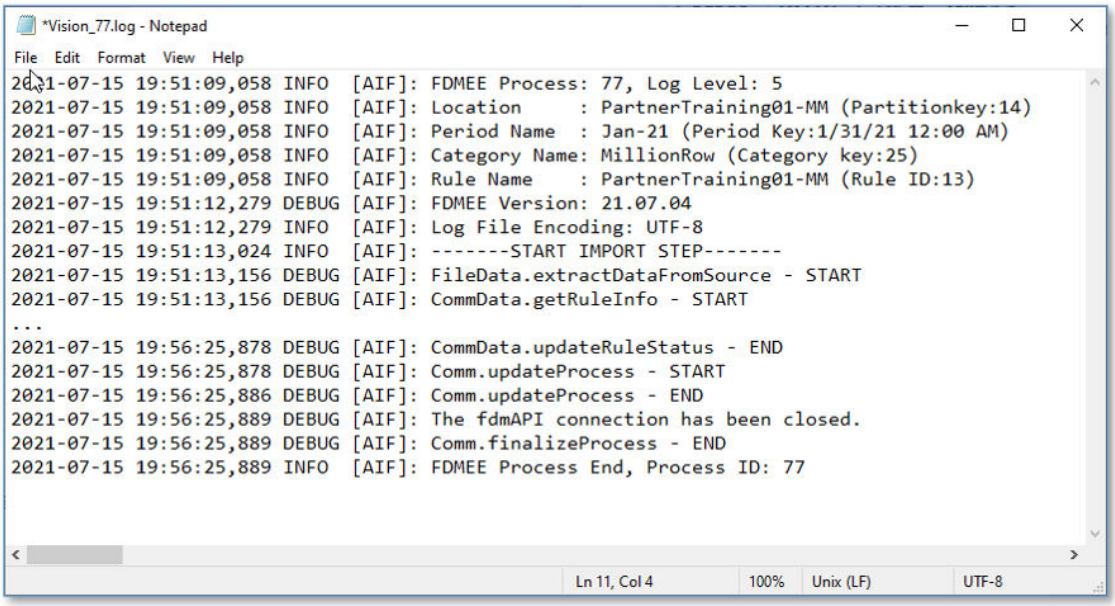
Simple Workflow Mode

With the Simple Workflow mode, the TDATASEG table is bypassed, and data is loaded directly to the target. This technique eliminates the copy of data to TDATASEG, and also the delete from TDATASEG. The only caveat is that drill-through to the Data Integration landing page is unavailable. (Drill-through using direct drill is available.)

Copyright © 2017, 2026, Oracle and/or its affiliates.



Using this simple workflow mode with expressions, the entire load process took 5 minutes and 16 seconds:



Quick Mode

Quick mode should be considered for high volume data loads that do not require complex transformations. Quick mode by-passes most of the steps and database tables in the workflow process, but does support expressions for simple transformations. Users are able to use the direct-drill feature even in Quick mode, and by-pass the Data Integration landing page when drilling.

> **Note**
>
> Quick mode is only available when loading to Oracle Essbase, and is not supported when loading to Account Reconciliation or Transaction Matching.

Partitioning Tables

A table partitioning option is available when loading large volumes of data. This feature is first used if a sudden performance degradation of your integrations has occurred. Table partitioning

Copyright © 2017, 2026, Oracle and/or its affiliates.

provides enhanced support for large data loads to the TDATASEG_T and TDATASEG tables by letting you decompose them into smaller, independent and more manageable pieces. Each data load works on a separate petition on which supports parallel processing. Additionally, a partition can be truncated instead of deleted from the from the TDATASEG_T table, which enhances performance.

In general, table partitioning is a technique used by database administrators to help manage performance of customer datasets. For Oracle Fusion Cloud Enterprise Performance Management, customers don't have direct access to the database, this feature has been enabled within Cloud EPM to allow users to directly perform this operation, but not as a direct user command against the database. Using this feature improves overall data load performance when a large number of mapping rules are used to transform the data prior to loading to the downstream business processes.

Run the **Partition Data Load Staging Tables** process only when no data load processes are running.

To enable table partitioning, see Partition Data Load Staging Tables.

Additional Considerations

When defining integrations, the Workflow Mode, and load method directly impact the performance of the load based on the specific data volume. When loading up to about 500,000 source records/rows, any workflow mode is recommended when using the load method of "Numeric Data Only."

When using the load method of "All Data with Security," expect that the data load takes longer because each row is validated against the target application in regard to any user defined security.

When loading files over 1,000,000 rows, the system performs batch updates and deletions from the TDATASEG_T and TDATASEG tables based on the "Batch Size" setting in the Target Options (see Defining Target Options). In some cases, files over 1,000,000 rows may be split into files each with less than 1,000,000 rows, and this usually results in a performance improvement. Users may then create multiple integrations, one for each file, and then combine these integrations into a batch, running the batch in parallel mode to maintain the performance achieved by splitting the file. This provides a single execution point, that kicks off multiple rules for the split file.

The following table provides recommendations in regard to workflow mode, load method and data volume.

Table A-1 Recommend Workflow Mode, Load Method, and Data Volume

Workflow Mode	Load Method	Row Count
Full Workflow	Numeric Data Only	Up to about 3 million rows
Simple Workflow	Numeric Data Only	Up to about 4-5 million rows
Full Workflow	Admin User All Data with Security Validate Data for Admin = Yes	Less than 500,000 rows
Full Workflow	Admin User All Data with Security Validate Data for Admin = No (This loads to the target using the Outline Load Utility)	Up to about 3 million rows

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table A-1 (Cont.) Recommend Workflow Mode, Load Method, and Data Volume

Workflow Mode	Load Method	Row Count
Quick Mode	Numeric Data Only	Any row count
Quick Mode Validate Data for Admin = Yes is not supported.	Admin User All Data with Security Validate Data for Admin = No (This loads to the target using the Outline Load Utility)	Any row count

> **Note**
>
> Tuning integrations is somewhat of an art, and the same techniques may not be applicable in all cases. Tuning usually requires multiple iterations to get to a final solution, and time should be included in all implementations to address tuning.

Copyright © 2017, 2026, Oracle and/or its affiliates.

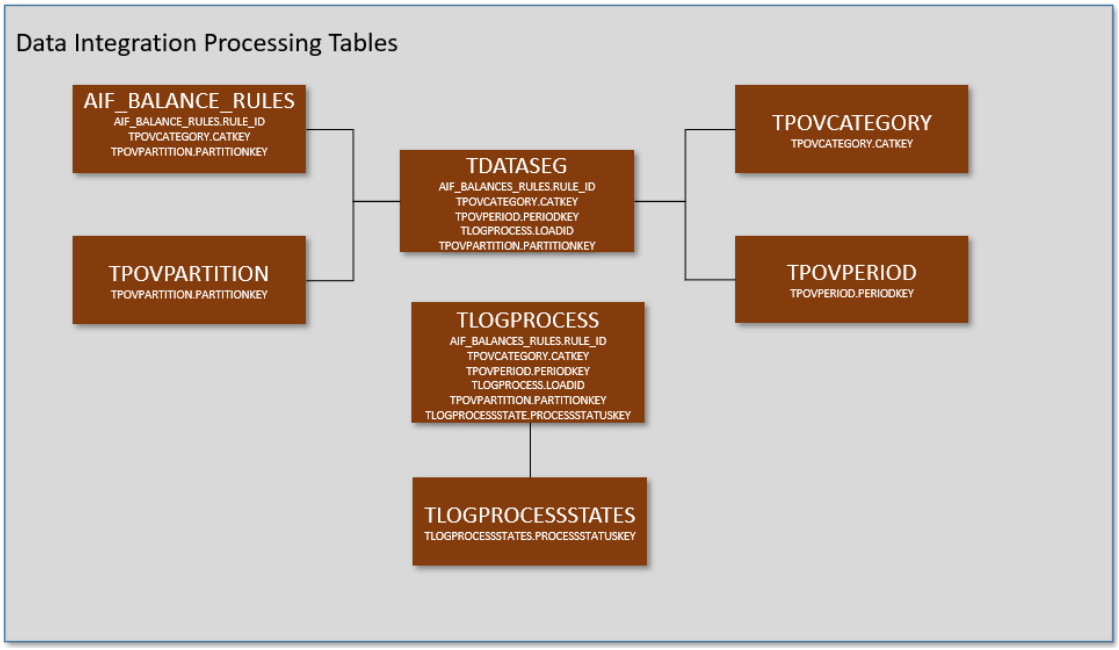
B

Building Reports Against the Data Integration Schema

To assist customers who wish to build reports in Data Integration, the following illustration shows a high level description of the processing tables in Data Integration.

Data Integration Processing Tables

The following illustrations shows the Data Integration processing tables and their relationships with each other.



Data Integration Table Definitions

Table definitions are shown for the following Data Integration tables:

- AIF_BALANCE_RULES Table Definition Reference
- TDATASEG Table Definition Reference
- TLOGPROCESS Table Definition Reference
- TLOGPROCESSSTATES Table Definition Reference
- TPOVCATEGORY Table Definition Reference
- TPOVPARTITION Table Definition Reference
- TPOVPERIOD Table Definition Reference

Copyright © 2017, 2026, Oracle and/or its affiliates.

AIF_BALANCE_RULES Table Definition Reference

Table B-1 AIF_BALANCE_RULES Definitions

AIF_BALANCE_RULES Column	AIF_BALANCE_RULES Definition
RULE_ID	NUMBER(15,0) NOT NULL ENABLE
RULE_NAME	VARCHAR2(80 CHAR) NOT NULL ENABLE
PARTITIONKEY	NUMBER(10,0)
CATKEY	NUMBER(10,0)
APPLICATION_ID	NUMBER(15,0) NOT NULL ENABLE
SOURCE_SYSTEM_ID	NUMBER(15,0) NOT NULL ENABLE
SOURCE_LEDGER_ID	NUMBER(15,0)
IMPGROUPKEY	VARCHAR2(20 CHAR)
PLAN_TYPE	VARCHAR2(30 CHAR)
SOURCE_APP_PLAN_TYPE	VARCHAR2(30 CHAR)
MULTI_PERIOD_FILE_FLAG	VARCHAR2(1 CHAR)
RULE_DESCRIPTION	VARCHAR2(400 CHAR)
STATUS	VARCHAR2(30 CHAR)
FILE_NAME_STATIC	VARCHAR2(256 CHAR)
FILE_PATH	VARCHAR2(256 CHAR)
FILE_NAME_DATE_FORMAT	VARCHAR2(30 CHAR)
FILE_NAME_SUFFIX_TYPE	VARCHAR2(30 CHAR)
PERIOD_MAPPING_TYPE	VARCHAR2(30 CHAR)
CALENDAR_ID	VARCHAR2(100 CHAR)
RULE_ATTR1	VARCHAR2(100 CHAR)
RULE_ATTR2	VARCHAR2(100 CHAR)
RULE_ATTR3	VARCHAR2(100 CHAR)
RULE_ATTR4	VARCHAR2(100 CHAR)
DIRECT_FILE_LOAD_FLAG	VARCHAR2(1 CHAR)
LEDGER_GROUP	VARCHAR2(10 CHAR)
INCL_ZERO_BALANCE_FLAG	VARCHAR2(1 CHAR)
BALANCE_SELECTION	VARCHAR2(30 CHAR)
AMOUNT_TYPE	VARCHAR2(30 CHAR)
BALANCE_METHOD_CODE	VARCHAR2(30 CHAR)
BALANCE_TYPE	VARCHAR2(1 CHAR)
BAL_SEG_VALUE_OPTION_CODE	VARCHAR2(30 CHAR)
EXCHANGE_RATE_OPTION_CODE	VARCHAR2(30 CHAR)
EXCHANGE_BEGIN_RATE_TYPE	VARCHAR2(30 CHAR)
EXCHANGE_END_RATE_TYPE	VARCHAR2(30 CHAR)
EXCHANGE_AVERAGE_RATE_TYPE	VARCHAR2(30 CHAR)
DATA_SYNC_OBJECT	VARCHAR2(255 CHAR)
DATA_SYNC_OBJECT_ID	VARCHAR2(50 CHAR)
INCLUDE_ADJ_PERIODS_FLAG	VARCHAR2(1 CHAR)
BALANCE_AMOUNT_BS	VARCHAR2(10 CHAR)
BALANCE_AMOUNT_IS	VARCHAR2(10 CHAR)

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table B-1 (Cont.) AIF_BALANCE_RULES Definitions

AIF_BALANCE_RULES Column	AIF_BALANCE_RULES Definition
AS_OF_DATE	DATE
BLANK_PERIODKEY	DATE
BR_MEMBER_NAME	VARCHAR2(80 CHAR)
BR_MEMBER_DISP_NAME	VARCHAR2(80 CHAR)
CURRENCY_CODE	VARCHAR2(10 CHAR)
DP_MEMBER_NAME	VARCHAR2(80 CHAR)
LEDGER_GROUP_ID	NUMBER(15,0)
VERSION	VARCHAR2(80 CHAR)
SIGNAGE_METHOD	VARCHAR2(30 CHAR)
LOAD_OPTIONS	VARCHAR2(2000 CHAR)
ICP_LOAD	VARCHAR2(1 CHAR)

TDATASEG Table Definition Reference

Table B-2 TDATASEG Table Reference

TDATASEG Column Name	TDATASEG Definition	TDATASEG Description
DATAKEY	NUMBER(31,0) NOT NULL ENABLE	System generated unique key for each row of data
PARTIONNKEY	NUMBER(10,0) NOT NULL ENABLE	Location key. Join to TPOVPARTITION to retrieve location information.
CATKEY	NUMBER(10,0) NOT NULL ENABLE	Category Key. Join to TPOVCATEGORY to retrieve category information.
PERIODKEY	DATE NOT NULL ENABLE	Period Key. Join to TPOVPERIOD to retrieve Data Management to EPM period mapping details.
DATAVIEW	VARCHAR2(20 CHAR) DEFAULT 'YTD' NOT NULL ENABLE	Hard coded to YTD for file, and set to YTD for balance sheet and PTD for income statement when pulling data from an Enterprise Resource Planning (ERP) system.
CURKEY	VARCHAR2(25 CHAR) DEFAULT	Currency code of the data.
CALCACCTTYPE	NUMBER(6,0) DEFAULT 9 NOT NULL ENABLE	Indicates if row was imported from source or computed by Logic Group: • 9=Imported • 5=Calculated and Exported • 1=Calculated, and Not Exported
CHANGESIGN	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE	Indicates that the sign of the imported amount should be reversed: • 0=No Change • 1=Reverse Sign
JOURNALID	VARCHAR2(80 CHAR) DEFAULT	ID for the Journal. User provided value
AMOUNT	NUMBER(38,12) DEFAULT 0 NOT NULL ENABLE	Amount loaded from source
AMOUNTX	NUMBER(38, 12) DEFAULT 0 NOT NULL ENABLE	Amount after any transformation rules. This value is loaded to the target application.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table B-2 (Cont.) TDATASEG Table Reference

TDATASEG Column Name	TDATASEG Definition	TDATASEG Description
DESC1	VARCHAR2(300 CHAR) DEFAULT	Description can be imported from file
DESC2	VARCHAR2(300 CHAR) DEFAULT	Description can be imported from file
ACCOUNT	VARCHAR2(300 CHAR) NOT NULL ENABLE	Account member from source
ACCOUNTX	VARCHAR2(2000 CHAR) DEFAULT	Account member after mapping rules processed
ACCOUNTR	NUMBER(10,0) DEFAULT 0 NOT NULL ENABLE	Key to Mapping used for this dimension. Refers to DATAKEY in TDATAMAPSEG.
ACCOUNTF	NUMBER(6,0) DEFAULT 0 NOT NULL ENABLE	Map types: • 1=Exception • 3=Between • 4=Range
ENTITY	VARCHAR2(300 CHAR) DEFAULT	Entity member from source
ENTITYX	VARCHAR2(300 CHAR) DEFAULT	Entity member after mapping rules processed. This value is exported.
ENTITYR	NUMBER(10,0) DEFAULT 0 NOT NULL ENABLE	Key to Mapping used for this dimension. Refers to DATAKEY in TDATAMAPSEG
ENTITYF	NUMBER(6,0) DEFAULT 0 NOT NULL ENABLE	Map types: • 1=Exception • 3=Between • 4=Range
ICP	VARCHAR2(300 CHAR) DEFAULT	ICP from source
ICPX	VARCHAR2(300 CHAR) DEFAULT	ICP after mapping rules processed. This value is exported.
ICPR	NUMBER(10,0) DEFAULT 0 NOT NULL ENABLE	Key to mapping used for this dimension. Refers to DATAKEY in TDATAMAPSEG.
ICPF	NUMBER(6,0) DEFAULT 0 NOT NULL ENABLE	Map type: • 1=Exception • 3=Between • 4=Range
UD1	VARCHAR2(280 CHAR) DEFAULT	UD1 from source
UD1X	VARCHAR2(280 CHAR) DEFAULT	UD1 after mapping rules processed. This value is exported.
UD1R	VARCHAR2(10, 0 CHAR) DEFAULT	Key to mapping used for this dimension. Refers to DATAKEY in TDATAMAPSEG
UD1F	VARCHAR2(6, 0 CHAR) DEFAULT 0 NOT NULL ENABLE	Map type: • 1=Exception • 3=Between • 4=Range
UD2	VARCHAR2(300 CHAR) DEFAULT	UD2 from source
UD2X	VARCHAR2(300 CHAR) DEFAULT	UD2X from source
UD2R	VARCHAR2(10, 0 CHAR) DEFAULT	UD2R from source
UD2F	VARCHAR2(6, 0 CHAR) DEFAULT 0 NOT NULL ENABLE	UD2F from source
UD3	VARCHAR2(300 CHAR) DEFAULT	UD3 from source
UD3X	VARCHAR2(300 CHAR) DEFAULT	UD3X from source
UD3R	VARCHAR2(10, 0 CHAR) DEFAULT	UD3R from source

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table B-2 (Cont.) TDATASEG Table Reference

TDATASEG Column Name	TDATASEG Definition	TDATASEG Description
UD3F	VARCHAR2(6, 0 CHAR) DEFAULT 0 NOT NULL ENABLE	UD3F from source
UD4	VARCHAR2(300 CHAR) DEFAULT	UD4 from source
UD4X	VARCHAR2(300 CHAR) DEFAULT	UD4X from source
UD4R	VARCHAR2(10, 0 CHAR) DEFAULT	UD4R from source
UD4F	VARCHAR2(6, 0 CHAR) DEFAULT 0 NOT NULL ENABLE	UD4F from source
UD5	VARCHAR2(300 CHAR) DEFAULT	UD5 from source
UD5X	VARCHAR2(300 CHAR) DEFAULT	UD5X from source
UD5R	VARCHAR2(10, 0 CHAR) DEFAULT	UD5R from source
UD5F	VARCHAR2(6, 0 CHAR) DEFAULT 0 NOT NULL ENABLE	UD5F from source
UD6	VARCHAR2(300 CHAR) DEFAULT	UD6 from source
UD6X	VARCHAR2(300 CHAR) DEFAULT	UD6X from source
UD6R	VARCHAR2(10, 0 CHAR) DEFAULT	UD6R from source
UD6F	VARCHAR2(6, 0 CHAR) DEFAULT 0 NOT NULL ENABLE	UD6F from source
UD7	VARCHAR2(300 CHAR) DEFAULT	UD7 from source
UD7X	VARCHAR2(300 CHAR) DEFAULT	UD7X from source
UD7R	VARCHAR2(10, 0 CHAR) DEFAULT	UD7R from source
UD7F	VARCHAR2(6, 0 CHAR) DEFAULT 0 NOT NULL ENABLE	UD7F from source
UD8	VARCHAR2(300 CHAR) DEFAULT	UD8 from source
UD8X	VARCHAR2(300 CHAR) DEFAULT	UD8X from source
UD8R	VARCHAR2(10, 0 CHAR) DEFAULT	UD8R from source
UD8F	VARCHAR2(6, 0 CHAR) DEFAULT 0 NOT NULL ENABLE	UD8F from source
ARCHIVEID	NUMBER(31,0) DEFAULT 0 NOT NULL ENABLE	Future use
HASMEMOITEM	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE	Future use
STATICDATAKEY	NUMBER(31,0) DEFAULT 0 NOT NULL ENABLE	Future use
UD9	VARCHAR2(300 CHAR) DEFAULT	UD9 from source
UD9X	VARCHAR2(300 CHAR) DEFAULT	UD9X from source
UD9R	VARCHAR2(10, 0 CHAR) DEFAULT	UD9R from source
UD9F	VARCHAR2(6, 0 CHAR) DEFAULT 0 NOT NULL ENABLE	UD9F from source
UD10	VARCHAR2(300 CHAR) DEFAULT	UD10 from source
UD10X	VARCHAR2(300 CHAR) DEFAULT	UD10X from source
UD10R	VARCHAR2(10, 0 CHAR) DEFAULT	UD10R from source
UD10F	VARCHAR2(6, 0 CHAR) DEFAULT 0 NOT NULL ENABLE	UD10F from source
UD11	VARCHAR2(300 CHAR) DEFAULT	UD11 from source
UD11X	VARCHAR2(300 CHAR) DEFAULT	UD11X from source
UD11R	VARCHAR2(10, 0 CHAR) DEFAULT	UD11R from source

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table B-2 (Cont.) TDATASEG Table Reference

TDATASEG Column Name	TDATASEG Definition	TDATASEG Description
UD11F	VARCHAR2(6, 0 CHAR) DEFAULT 0 NOT NULL ENABLE	UD11F from source
UD12	VARCHAR2(300 CHAR) DEFAULT	UD12 from source
UD12X	VARCHAR2(300 CHAR) DEFAULT	UD12X from source
UD12R	VARCHAR2(10, 0 CHAR) DEFAULT	UD12R from source
UD12F	VARCHAR2(6, 0 CHAR) DEFAULT 0 NOT NULL ENABLE	UD12F from source
UD13	VARCHAR2(300 CHAR) DEFAULT	UD13 from source
UD13X	VARCHAR2(300 CHAR) DEFAULT	UD13X from source
UD13R	VARCHAR2(10, 0 CHAR) DEFAULT	UD13R from source
UD13F	VARCHAR2(6, 0 CHAR) DEFAULT 0 NOT NULL ENABLE	UD13F from source
UD14	VARCHAR2(300 CHAR) DEFAULT	UD14 from source
UD14X	VARCHAR2(300 CHAR) DEFAULT	UD14X from source
UD14R	VARCHAR2(10, 0 CHAR) DEFAULT	UD14R from source
UD14F	VARCHAR2(6, 0 CHAR) DEFAULT 0 NOT NULL ENABLE	UD14F from source
UD15	VARCHAR2(80 CHAR) DEFAULT	UD15 from source
UD15X	VARCHAR2(300 CHAR) DEFAULT	UD15X from source
UD15R	VARCHAR2(10, 0 CHAR) DEFAULT	UD15R from source
UD15F	VARCHAR2(6, 0 CHAR) DEFAULT 0 NOT NULL ENABLE	UD15F from source
UD16	VARCHAR2(300 CHAR) DEFAULT	UD16 from source
UD16X	VARCHAR2(300 CHAR) DEFAULT	UD16X from source
UD16R	VARCHAR2(10, 0 CHAR) DEFAULT	UD16R from source
UD16F	VARCHAR2(6, 0 CHAR) DEFAULT 0 NOT NULL ENABLE	UD16F from source
UD17	VARCHAR2(300 CHAR) DEFAULT	UD17 from source
UD17X	VARCHAR2(300 CHAR) DEFAULT	UD1X from source
UD17R	VARCHAR2(10, 0 CHAR) DEFAULT	UD17R from source
UD17F	VARCHAR2(6, 0 CHAR) DEFAULT 0 NOT NULL ENABLE	UD17F from source
UD18	VARCHAR2(300 CHAR) DEFAULT	UD18 from source
UD18X	VARCHAR2(300 CHAR) DEFAULT	UD18X from source
UD18R	VARCHAR2(10, 0 CHAR) DEFAULT	UD18R from source
UD18F	VARCHAR2(6, 0 CHAR) DEFAULT 0 NOT NULL ENABLE	UD18F from source
UD19	VARCHAR2(300 CHAR) DEFAULT	UD19 from source
UD19X	VARCHAR2(300 CHAR) DEFAULT	UD19X from source
UD19R	VARCHAR2(10, 0 CHAR) DEFAULT	UD19R from source
UD19F	VARCHAR2(6, 0 CHAR) DEFAULT 0 NOT NULL ENABLE	UD19F from source
UD20	VARCHAR2(300 CHAR) DEFAULT	UD20 from source
UD20X	VARCHAR2(300 CHAR) DEFAULT	UD20X from source
UD20R	VARCHAR2(10, 0 CHAR) DEFAULT	UD20R from source
UD20F	VARCHAR2(6, 0 CHAR) DEFAULT 0 NOT NULL ENABLE	UD20F from source

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table B-2 (Cont.) TDATASEG Table Reference

TDATASEG Column Name	TDATASEG Definition	TDATASEG Description
ATTR1	VARCHAR2(300 CHAR) DEFAULT	User defined attribute - used as needed for mapping or drill-through
ATTR2	VARCHAR2(300 CHAR) DEFAULT	User defined attribute - used as needed for mapping or drill-through **Note** When importing data from Financial Consolidation and Close, attribute columns ATTR2 and ATTR3 should not be used for any other dimension mappings.
ATTR3	VARCHAR2(300 CHAR) DEFAULT	User defined attribute - used as needed for mapping or drill-through **Note** When importing data from Financial Consolidation and Close, attribute columns ATTR2 and ATTR3 should not be used for any other dimension mappings.
ATTR4	VARCHAR2(300 CHAR) DEFAULT	User defined attribute - used as needed for mapping or drill-through
ATTR5	VARCHAR2(300 CHAR) DEFAULT	User defined attribute - used as needed for mapping or drill-through
ATTR6	VARCHAR2(300 CHAR) DEFAULT	User defined attribute - used as needed for mapping or drill-through
ATTR7	VARCHAR2(300 CHAR) DEFAULT	User defined attribute - used as needed for mapping or drill-through
ATTR8	VARCHAR2(300 CHAR) DEFAULT	User defined attribute - used as needed for mapping or drill-through

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table B-2 (Cont.) TDATASEG Table Reference

TDATASEG Column Name	TDATASEG Definition	TDATASEG Description
ATTR9	VARCHAR2(300 CHAR) DEFAULT	User defined attribute - used as needed for mapping or drill-through
ATTR10	VARCHAR2(300 CHAR) DEFAULT	User defined attribute - used as needed for mapping or drill-through
ATTR11	VARCHAR2(300 CHAR) DEFAULT	User defined attribute - used as needed for mapping or drill-through
ATTR12	VARCHAR2(300 CHAR) DEFAULT	User defined attribute - used as needed for mapping or drill-through
ATTR13	VARCHAR2(300 CHAR) DEFAULT	Restricted. Used to store the "Logic Item" if a logic group is assigned to a location.
ATTR14	VARCHAR2(300 CHAR) DEFAULT	User defined attribute - used as needed for mapping or drill-through
ATTR15	VARCHAR2(300 CHAR)	User defined attribute - used as needed for mapping or drill-through
ATTR16	VARCHAR2(300 CHAR)	User defined attribute - used as needed for mapping or drill-through
ATTR17	VARCHAR2(300 CHAR)	User defined attribute - used as needed for mapping or drill-through
ATTR18	VARCHAR2(300 CHAR)	User defined attribute - used as needed for mapping or drill-through
ATTR19	VARCHAR2(300 CHAR)	User defined attribute - used as needed for mapping or drill-through
ATTR20	VARCHAR2(300 CHAR)	User defined attribute - used as needed for mapping or drill-through
ATTR21	VARCHAR2(300 CHAR)	User defined attribute - used as needed for mapping or drill-through
ATTR22	VARCHAR2(300 CHAR)	User defined attribute - used as needed for mapping or drill-through
ATTR23	VARCHAR2(300 CHAR)	User defined attribute - used as needed for mapping or drill-through
ATTR24	VARCHAR2(300 CHAR)	User defined attribute - used as needed for mapping or drill-through
ATTR25	VARCHAR2(300 CHAR)	User defined attribute - used as needed for mapping or drill-through
ATTR26	VARCHAR2(300 CHAR)	User defined attribute - used as needed for mapping or drill-through
ATTR27	VARCHAR2(300 CHAR)	User defined attribute - used as needed for mapping or drill-through
ATTR28	VARCHAR2(300 CHAR)	User defined attribute - used as needed for mapping or drill-through
ATTR29	VARCHAR2(300 CHAR)	User defined attribute - used as needed for mapping or drill-through
ATTR30	VARCHAR2(300 CHAR)	User defined attribute - used as needed for mapping or drill-through
ATTR31	VARCHAR2(300 CHAR)	User defined attribute - used as needed for mapping or drill-through
ATTR32	VARCHAR2(300 CHAR)	User defined attribute - used as needed for mapping or drill-through
ATTR33	VARCHAR2(300 CHAR)	User defined attribute - used as needed for mapping or drill-through

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table B-2 (Cont.) TDATASEG Table Reference

TDATASEG Column Name	TDATASEG Definition	TDATASEG Description
ATTR34	VARCHAR2(300 CHAR)	User defined attribute - used as needed for mapping or drill-through
ATTR35	VARCHAR2(300 CHAR)	User defined attribute - used as needed for mapping or drill-through
ATTR36	VARCHAR2(300 CHAR)	User defined attribute - used as needed for mapping or drill-through
ATTR37	VARCHAR2(300 CHAR)	User defined attribute - used as needed for mapping or drill-through
ATTR38	VARCHAR2(300 CHAR)	User defined attribute - used as needed for mapping or drill-through
ATTR39	VARCHAR2(300 CHAR)	User defined attribute - used as needed for mapping or drill-through
ATTR40	VARCHAR2(300 CHAR)	User defined attribute - used as needed for mapping or drill-through
DATA	VARCHAR2(2000 CHAR)	
DATAX	VARCHAR2(2000 CHAR)	
DATAR	NUMBER(10,0) DEFAULT 0 NOT NULL ENABLE	
DATAF	NUMBER(6,0) DEFAULT 0 NOT NULL ENABLE	
CODE_COMBINATION_ID	VARCHAR2(500 CHAR)	Used for integration with Oracle E-Business Suite.
AMOUNT_YTD	NUMBER(29,12)	YTD Amount. Used for E-Business Suite, Peoplesoft, Fusion data sources
AMOUNT_PTD	NUMBER(29,12)	PTD Amount. Used for E-Business Suite, Peoplesoft, Fusion data sources
LOADID	NUMBER(15,0)	Process ID that created or updated this row.
RULE_ID	NUMBER(15,0)	Data Rule ID used to create this row. Join to AIF_BALANCE_RULES for details.
STAT_BALANCE_FLAG	VARCHAR2(1 CHAR)	Indicates if balance is a statistic: • Y=Stat • N=Balance
VALID_FLAG	VARCHAR2(1 CHAR)	Indicates if row has valid mappings: • Y=Valid • N=Not Valid • I=Ignore

TLOGPROCESS Table Definition Reference

Table B-3 TLOGPROCESS Table Definition Reference

TLOGPROCESS Column	TLOGPROCESS Definition	Description
PARTITIONKEY	NUMBER(10,0) NOT NULL ENABLE	Location key. Join to TPOVPARTITION to retrieve location information.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table B-3 (Cont.) TLOGPROCESS Table Definition Reference

TLOGPROCESS Column	TLOGPROCESS Definition	Description
CATKEY	NUMBER(10,0) NOT NULL ENABLE	Category Key. Join to TPOVCATEGORY to retrieve category information.
PERIODKEY	DATE NOT NULL ENABLE	Period Key. Join to TPOVPERIOD to retrieve FDMEE to EPM period mapping details.
RULE_ID	NUMBER(15,0) NOT NULL ENABLE	Data Rule ID. Join to AIF_BALANCE_RULES for details.
PROCESSIMP	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE	Status for Import step: • 0=Not started or failed • 1=Successful
PROCESSIMPNOTE	VARCHAR2(50 CHAR) DEFAULT NULL,	Textual note on Validate status: • Import Successful • Recalculated OK • Import Error • Recalculate Failed • MultiLoad • BypassDataLoad • Logic Calc Err • Map Calc Err
PROCESSVAL	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE	Status for Validate step: • 0=Not started or failed • 1=Successful
PROCESSVALNOTE	VARCHAR2(50 CHAR) DEFAULT NULL	Textual note on Validate step: • Validate Successful • Error= x records (Where X = how many members did not have map rules) • BypassDataLoad
PROCESSEXP	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE	Status for Export step: • 0=Not started or failed • 1=Successful
PROCESSEXPNOTE	VARCHAR2(50 CHAR) DEFAULT NULL	Textual note on Export step: • Last successful export • Export -B Successful • Export Successful • BypassDataLoad
PROCESSENTLOAD	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE	Status for Load step: • 0=Not started or failed • 1=Successful
PROCESSENTLOADNOTE	VARCHAR2(50 CHAR) DEFAULT NULL	Textual note on Load status: • Time Date stamp for success • Load Error • BypassDataLoad

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table B-3 (Cont.) TLOGPROCESS Table Definition Reference

TLOGPROCESS Column	TLOGPROCESS Definition	Description
PROCESSENTVAL	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE	Status for Check step: • 0=Not started or failed • 1=Successful
PROCESSENTVALNOTE	VARCHAR2(50 CHAR) DEFAULT NULL	Textual note on Check step: • Check Successful • Check Error • BypassDataLoad
PROCESSCERT	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE	Status for Certification step: • 0=Not started or unsubmitted • 1=Submitted
PROCESSCERTNOTE	VARCHAR2(50 CHAR) DEFAULT NULL	Textual note on Load status: • Controls Submitted • Controls Cleared • Not Submitted • No Controls Found for Profile • No Controls Group Assigned • Error Setting Status
PROCESSASSES	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE	Status for Assessment (process explorer) step: • 0=Not started or unsubmitted • 1=Submitted
PROCESSASSESNOTE	VARCHAR2(50 CHAR) DEFAULT NULL	Textual note on Load status: • Controls Submitted • Controls Cleared • Not Submitted • No Controls Found for Profile • No Controls Group Assigned • Error Setting Status
PROCESSCHILDDONE	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE	Status for Certification status for parent locations step: • 0=Not started or all children not complete • 1=All children complete
PROCESSCHILDDONENOTE	VARCHAR2(50 CHAR) DEFAULT NULL	Textual note on Certification status for parent location: • Children Submitted • No Children
PROCESSUD1	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE	Not Used
PROCESSUD1NOTE	VARCHAR2(50 CHAR) DEFAULT NULL	Not Used

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table B-3 (Cont.) TLOGPROCESS Table Definition Reference

TLOGPROCESS Column	TLOGPROCESS Definition	Description
PROCESSUD2	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE	Not Used
PROCESSUD2NOTE	VARCHAR2(50 CHAR) DEFAULT NULL	Not Used
PROCESSUD3	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE	Not Used
PROCESSUD3NOTE	VARCHAR2(50 CHAR) DEFAULT NULL	Not Used
PROCESSUD4	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE	Not Used
PROCESSUD4NOTE	VARCHAR2(50 CHAR) DEFAULT NULL	Not Used
PROCESSENDTIME	DATE DEFAULT TO_DATE('01/01/1900', 'MM/DD/YYYY') NOT NULL ENABLE	Last update time/date
BLNWCDIRTY	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE	Flag used to indicate that maps (WC = WildCard) must be recalculated prior to validating: • 0=OK • 1=Re-caculate location Map rules have changed after data was imported. This causes the Calculate flag to be displayed.
BLNLOGICDIRTY	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE	Flag used to indicate the LOGIC must be recalculated prior to validating.
BLNVALDIRTY	NUMBER(1,0) DEFAULT 1 NOT NULL ENABLE	Flag used to indicate when Validation workflow must be re-run: • 0=OK • 1=re-process validations Map rules have changed after data was imported. This causes the Calculate flag to be displayed.
INTLOCKSTATE	NUMBER(6,0) DEFAULT 50 NOT NULL ENABLE	Location POV lock status: • 50=open • 60=locked
PROCESSSTATUS	NUMBER(10,0) DEFAULT 0 NOT NULL ENABLE	Current state of workflow for location/category/per. Valid status from LogProcessStates
LOADID	NUMBER(15,0)	Load ID.

Copyright © 2017, 2026, Oracle and/or its affiliates.

TLOGPROCESSSTATES Table Definition Reference

Table B-4 TLOGPROCESSSTATES Table Reference

TLOGPROCESSSTATES Columns	TLOGPROCESSSTATES Descriptions
PROCESSSTATUSKEY	NUMBER(10,0) DEFAULT 0 NOT NULL ENABLE
PROCESSSTATUSDESC	VARCHAR2(100 CHAR)
PROCESSSTATUSCODE	VARCHAR2(10 CHAR)

TPOVCATEGORY Table Definition Reference

Table B-5 TPOVCATEGORY Table Definition Reference

TPOVCATEGORY Column	TPOVCATEGORY Definition
CATKEY	NUMBER(10,0) NOT NULL ENABLE
CATNAME	VARCHAR2(20 CHAR) NOT NULL ENABLE
CATDESC	VARCHAR2(50 CHAR) DEFAULT NULL
CATTARGET	VARCHAR2(75 CHAR) NOT NULL ENABLE
CATFREQ	VARCHAR2(10 CHAR) DEFAULT 'M' NOT NULL ENABLE

TPOVPARTITION Table Definition Reference

Table B-6 TPOVPARTITION Table Definitions

TPOVPARTITION Columns	TPOVPARTITION Definition
PARTITIONKEY	NUMBER(10,0) NOT NULL ENABLE
PARTNAME	VARCHAR2(20 CHAR) NOT NULL ENABLE
PARTDESC	VARCHAR2(100 CHAR) DEFAULT NULL
PARTNOTES	VARCHAR2(255 CHAR) DEFAULT NULL
PARTLASTIMPFILE	VARCHAR2(250 CHAR) DEFAULT NULL
PARTLASTEXPFILE	VARCHAR2(250 CHAR) DEFAULT NULL
PARTIMPGROUP	VARCHAR2(20 CHAR) DEFAULT '[NONE]' NOT NULL ENABLE
PARTLOGICGROUP	VARCHAR2(20 CHAR) DEFAULT '[NONE]' NOT NULL ENABLE
PARTVALGROUP	VARCHAR2(20 CHAR) DEFAULT '[NONE]' NOT NULL ENABLE
PARTVALENTGROUP	VARCHAR2(20 CHAR) DEFAULT '[NONE]' NOT NULL ENABLE
PARTCURRENCYKEY	VARCHAR2(25 CHAR) DEFAULT '[NONE]' NOT NULL ENABLE
PARTPARENT	VARCHAR2(20 CHAR) DEFAULT NULL
PARTTYPE	NUMBER(6,0) DEFAULT 2 NOT NULL ENABLE
PARTSEQMAP	NUMBER(1,0) DEFAULT 0 NOT NULL ENABLE

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table B-6 (Cont.) TPOVPARTITION Table Definitions

TPOVPARTITION Columns	TPOVPARTITION Definition
PARTDATAVALUE	VARCHAR2(100 CHAR) DEFAULT NULL,
PARTSEGMENTKEY	NUMBER(6,0) DEFAULT 1 NOT NULL ENABLE,
PARTCONTROLSTYPE	NUMBER(10,0) DEFAULT 1 NOT NULL ENABLE
PARTCONTROLSGROUP1	VARCHAR2(80 CHAR) DEFAULT '[NONE]' NOT NULL ENABLE
PARTCONTROLSGROUP2	VARCHAR2(80 CHAR) DEFAULT '[NONE]' NOT NULL ENABLE
PARTCONTROLSAPPROVER	VARCHAR2(255 CHAR) DEFAULT NULL
PARTCONTROLSAPPROVERPROXY	VARCHAR2(255 CHAR) DEFAULT NULL
PARTCONTROLSREDFLAGLEVEL	NUMBER(10,0) DEFAULT 6 NOT NULL ENABLE
PARTCLOGICGROUP	VARCHAR2(20 CHAR) DEFAULT '[NONE]' NOT NULL ENABLE
PARTINTGCONFIG1	VARCHAR2(300 CHAR) DEFAULT NULL
PARTINTGCONFIG2	VARCHAR2(300 CHAR) DEFAULT NULL
PARTINTGCONFIG3	VARCHAR2(300 CHAR) DEFAULT NULL
PARTINTGCONFIG4	VARCHAR2(300 CHAR) DEFAULT NULL
PARTADAPTOR	VARCHAR2(10 CHAR) DEFAULT '[NONE]' NOT NULL ENABLE
PARTSOURCESYSTEMID	NUMBER(15,0)
PARTSOURCELEDGERID	NUMBER(15,0)
PARTTARGETAPPLICATIONID	NUMBER(15,0)
PARTPARENTKEY	NUMBER(10,0)
PARTSOURCEAPPLICATIONID	NUMBER(15,0)
PARTTARGETSOURCESYSTEMID	NUMBER(15,0)

TPOVPERIOD Table Definition Reference

Table B-7 TPOVPERIOD Table Definition

TPOVPERIOD Column	TPOVPERIOD Definition
PERIODKEY	DATE NOT NULL ENABLE
PRIORPERIODKEY	DATE NOT NULL ENABLE
PERIODDESC	VARCHAR2(20 CHAR) NOT NULL ENABLE
PERIODTARGETM	VARCHAR2(80 CHAR) NOT NULL ENABLE
PERIODTARGETQ	VARCHAR2(80 CHAR) DEFAULT NULL
PERIODTARGETY	VARCHAR2(80 CHAR) DEFAULT NULL,
PERIODTARGETD	VARCHAR2(80 CHAR) DEFAULT NULL
YEARTARGET	VARCHAR2(80 CHAR) NOT NULL ENABLE

Copyright © 2017, 2026, Oracle and/or its affiliates.

C

Troubleshooting Data Integration Issues

For help with troubleshooting issues in Data Integration, see the following topic in the *Operations Guide*:

Table C-1 Data Integration Troubleshooting Issues

Issue in Data Integration	Tip
Data Integration Errors	Handling Data Integration Errors in *Oracle Enterprise Performance Management Cloud Operations Guide*
Oracle Fusion Cloud Enterprise Performance Management Roles in Data Integration	Required Roles for Loading Data in *Oracle Enterprise Performance Management Cloud Operations Guide*
Importing or Exporting data	Troubleshooting Data Management and Data Integration Issues in *Oracle Enterprise Performance Management Cloud Operations Guide*
Getting Help and Support	Getting Help in *Oracle Enterprise Performance Management Cloud Operations Guide*

Copyright © 2017, 2026, Oracle and/or its affiliates.

D

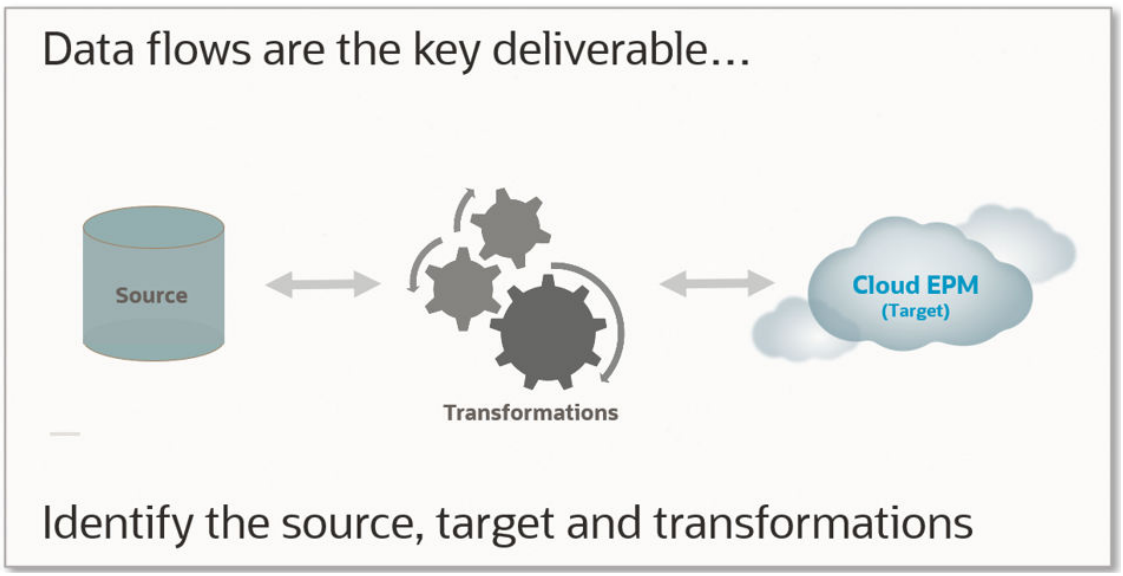
Data Integration Best Practices

Use the best practices in this section to help you improve integration processes using Data Integration.

Defining Data Flows

Data is the basis for all cloud Oracle Fusion Cloud Enterprise Performance Management business processes, and it is an important decision when it comes to selecting the tools to enable the ability to load the required data. Many times, a discussion related to integration tools precedes a discussion of the required data flows, and it is these data flows that drive the integration tools discussion.

The best practice for integrations is to first define data flows by determining the sources, transformations, and target business processes. This should be completed before any discussion of tools or integration features. For example, a data flow may use the Fusion GL as the source, Financial Consolidation and Close as the target, and then transformations to map the source chart of accounts into the dimensionality within Financial Consolidation and Close.



After the data flows are identified, it is then an exercise to determine the tools that will enable each of the data flows. For net-new cloud Cloud EPM implementations, it is recommended to first review the integration tools available in EPM before considering other tools. (For example, the Data Integration, which is available in each Cloud EPM business process can support most, if not all defined data flows.)

In other cases, a customer may have existing tools that are used for integrations, or tools which might be an IT standard. In these cases, it is easy to incorporate the Cloud EPM tools, and Cloud EPM APIs into an existing integration architecture or set of tools.

Even in the case of existing tools, it is still important to define all data flows as the starting point for the Cloud EPM integration requirements.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Copyright © 2017, 2026, Oracle and/or its affiliates.

E

Scheduling Jobs

The EPM Platform Job Scheduler console supports the execution and scheduling of Data Management/Data Integration jobs. This feature provides consistent scheduling functionality in the Oracle Fusion Cloud Enterprise Performance Management.

Jobs that can be scheduled in the EPM Platform Job Scheduler console include:

- data load rules and integrations. See Scheduling a Data Rule, Integration, or Batch Definition to Run in the Job Scheduler and Scheduling a Data Rule or Integration Definition to Run Later in the Job Scheduler.
- batches. See Scheduling a Data Rule, Integration, or Batch Definition to Run in the Job Scheduler and Scheduling a Data Rule or Integration Definition to Run Later in the Job Scheduler.
- System Maintenance scripts. See Scheduling Maintenance Scripts in the Job Scheduler.

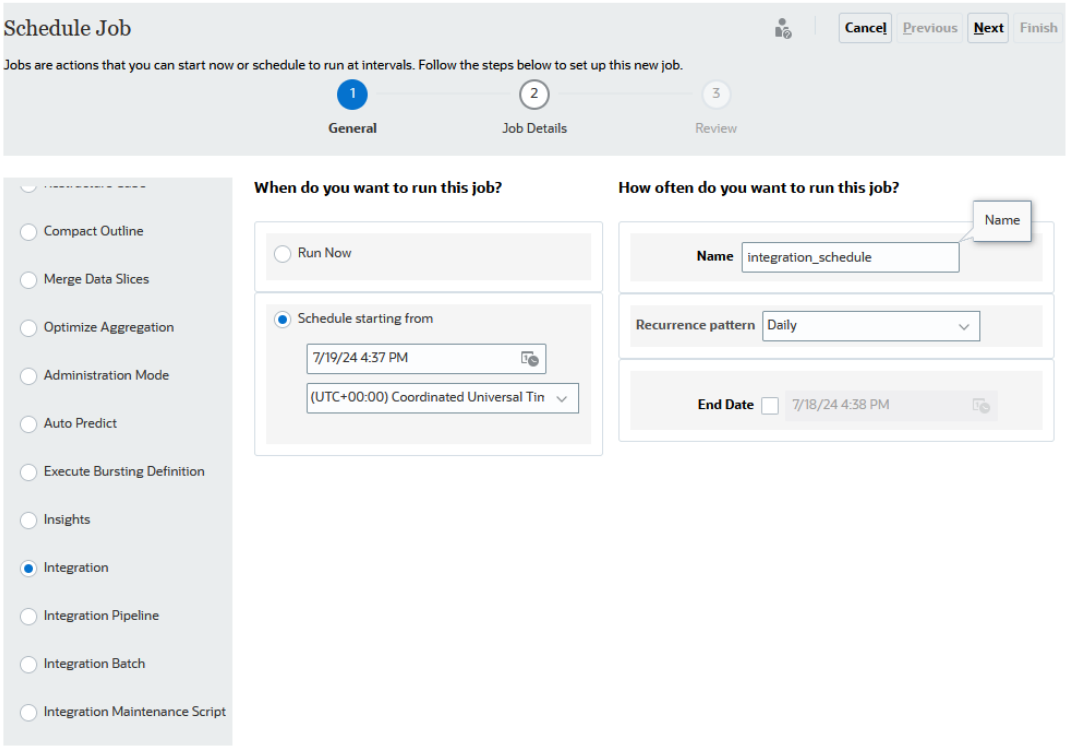
Scheduling a Data Rule, Integration, or Batch Definition to Run in the Job Scheduler

You can schedule the execution times of a data load rule or batch in Data Management or an integration in Data Integration.

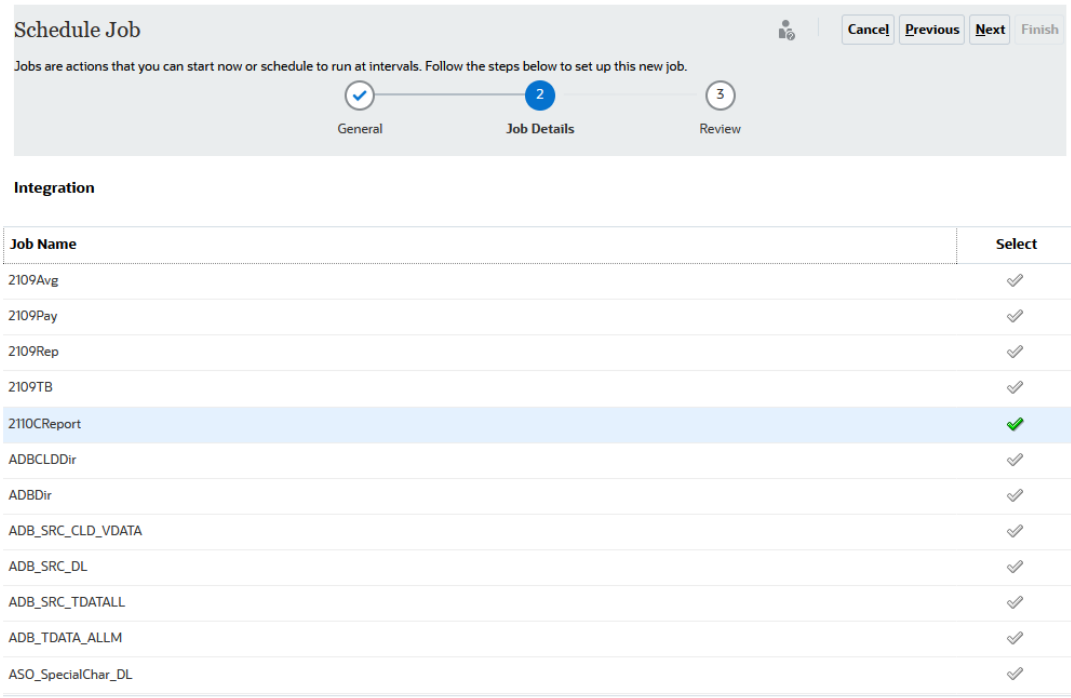
To schedule a data load rule, integration, or batch:

1. Click **Application**, then **Jobs**, and then **Schedule Jobs**.
2. From the **Schedule Jobs** page, then **What type of job is this?**, select **Integration**.

Copyright © 2017, 2026, Oracle and/or its affiliates.



3. Click **Next**.
4. From **Job Name**, select the data rule or integration to schedule and click .



5. Click **Next**.
6. From **Import Mode**, select the method for importing data.

   Options include:

Copyright © 2017, 2026, Oracle and/or its affiliates.

- **Append**—Keep existing rows for the POV but append new rows to the POV. For example, a first-time load has 100 rows and second load has 50 rows. In this case, 50 rows are appended. After this load, the row total for the POV is 150.
- **Replace**—Clears all data for the POV in the target, and then loads from the source or file. For example, a first-time load has 100 rows, and a second load has 70 rows. In this case, 100 rows are removed, and 70 rows are loaded to TDATASSEG. After this load, the row total is 70.

  For a Planning application, Replace clears data for Year, Period, Scenario, Version, and Entity that you are loading, and then loads the data from source or file. Note that when you have a year of data in the Planning application, but are only loading a single month, this option clears the entire year before performing the load.

  > **Note**
  >
  > When running an integration in Replace mode to an ASO cube, if the scenario member is a shared member, then only a Numeric data load is performed. Be sure to specify the member name with fully qualified name including complete hierarchy. The All Data Type load methods does not work when the scenario member is a shared member.

  > **Note**
  >
  > Replace Mode is not supported for the load method "All data types with auto-increment of line item."

- **Merge**—(Account Reconciliation only). Merge changed balances with existing data for the same location.

  Merge mode eliminates the need to load an entire data file when only a few balances have changed since the last time data was loaded into Account Reconciliation. If mappings change between two loads, customers must reload the full data set.

  For example, a customer might have 100 rows of existing balances for one number account IDs, each which has an amount of $100.00. If the customer runs the integration in merge mode and the source has one row for one account ID with an amount of $80, then after running integration, there are one hundred rows of balances, 99 each of which have a balance of $100.00, and 1 which has a balance of $80.00.
- **No Import**—Skip the import of data entirely.
- **Map and Validate**—Skip importing the data but reprocess the data with updated mappings.

7. From **Export Mode**, select the method for exporting data to the target application.

   Options include:

   - **Merge**—Overwrite existing date with the new data from the load file. (By default, all data load is processed in the Merge mode.) If data does not exist, create new data.
   - **Replace**—Clears all data for the POV in the target, and then loads from the source or file. For example, a first-time load has 100 rows, and a second load has 70 rows. In this case, 100 rows are removed, and 70 rows are loaded to the staging table. After this load, the row total is 70.

     For a Planning application, Replace clears data for Year, Period, Scenario, Version, and Entity that you are loading, and then loads the data from source or file. Note that

Copyright © 2017, 2026, Oracle and/or its affiliates.

when you have a year of data in the Planning application, but are only loading a single month, this option clears the entire year before performing the load.

- **Accumulate**—Accumulate the data in the application with the data in the load file. For each unique point of view in the data file, the value from the load file is added to the value in the application.
- **Subtract**—Subtract the value in the source or file from the value in the target application. For example, when you have 300 in the target, and 100 in the source, then the result is 200.
- **Dry Run**—(Financial Consolidation and Close and Tax Reporting only) Scan a data load file for invalid records without loading data it to the target application. The system validates the data load file and lists any invalid records into a log, which lists 100 errors or less. For each error, the log indicates each record in error with its corresponding error message. Log details are available in Process Details.

> **Note**
>
> Dry Run ignores the Enable Data Security for Admin Users target option and always uses the REST API for the administrator user.

- **No Export**—Skip the export of data entirely.
- **Check**—After exporting data to the target system, display the Check report for the current POV. If check report data does not exist for the current POV, a blank page is displayed.

8. From **Start Period**, select the first period for which data is to be loaded.You can filter periods by simply typing the character(s) to filter by. For example, type J to filter by months beginning J such as June or July. You can also click the drop down and specify additional filter criteria in the edit box shown below More results available, please filter further. This period name must be defined in period mapping.

   This period name must be defined in period mapping.

   > **Note**
   >
   > Use a Start Period and End Period that belong to a single Fiscal Year. If a period range crosses fiscal years, you run into the following issues:
   >
   > - When loading data in replace mode, the system clear data for both years
   > - When exporting data, you will get duplicate data
   >
   > The above is applicable for all modes of data load including standard and quick mode.

9. From **End Period**, select the last period for which data is to be loaded. This period name must be defined in period mapping.
10. Click **OK**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Scheduling a Data Rule or Integration Definition to Run Later in the Job Scheduler

Table E-1 Scheduling Jobs Options

Option	Description
Schedule starting from	Select the starting date and time, including the time zone.
Name	Specify a name for the scheduled job; for example, **MyDailyCubeRefresh**. The name you specify is displayed along with the job name (which you'll choose on the next screen); for example, **MyDailyCubeRefresh : Refresh Database**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

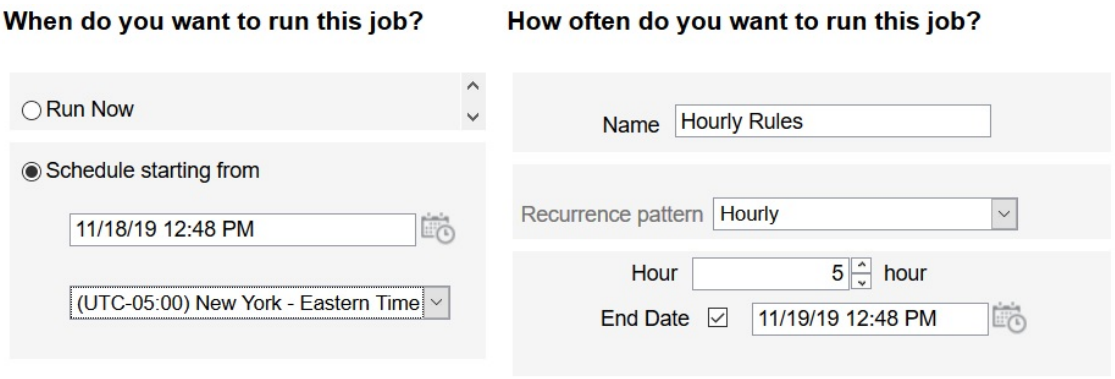
Table E-1 (Cont.) Scheduling Jobs Options

Option	Description
Recurrence Pattern	Specify the frequency with which to run the job: • **Hourly**: Hourly jobs run according to a timetable which is based on the values you set in the **Schedule starting from** and **Hour** fields. The schedule for **Hourly** jobs restarts each day during the midnight hour and the recurring job starts at the first hour, second hour, third hour (and so on up to 12 hours) after the midnight hour, depending on the value you select in the **Hour** field. So for example, if you specify a value of **5** in the **Hour** field, the possible times the job could start are during the 12:00 AM hour, the 5:00 AM hour, the 10:00 AM hour, the 3:00 PM hour, and the 8:00 PM hour each day. If the scheduled start time for a job is 12:48 PM and you set the **Hourly** recurrence for **5 Hours**, the job will start at 3:48 PM, which is the first scheduled time available after 12:48 PM in which to start a 5-hour recurring job. Then the job will run again at 8:48 PM, 12:48 AM, 5:48 AM, and 10:48 AM. For the default recurrence of **1 Hour**, the job will start running at the start time you specify and continue to run each hour until the ending date and time; for example, if the scheduled start time for a job is 12:48, the job will run at 12:48, 1:48, 2:48, 3:48 and so on. • **By Minute**: Set the **Frequency** for 15 or 30 minutes. The job will start running in 15 or 30 minutes and continue to run at the selected frequency until the ending date and time; for example, if you set the job to run every 15 minutes and the starting time is 3:15, the job will start to run at 3:30, then 3:45, and so on. Jobs cannot be scheduled to run in increments smaller than 15 minutes. • **Run Once**: The job will run once at the starting date and time. • **Yearly**: The job will run at the starting date and time and continue to run each year thereafter until the end date. • **Monthly**: The job will run at the starting date and time and continue to run each month thereafter until the end date. • **Weekly**: The job will run at the starting date and time and continue to run each week until the end date. • **Daily**: The job will run at the starting date and time and continue to run each day until the end date.
End Date	If the job is recurring, select an ending date and time.

Scheduling Hourly Jobs

Hourly jobs run according to a timetable which is based on the values you set in the **Schedule starting from** and **Hour** fields. The schedule for **Hourly** jobs restarts each day during the midnight hour and the recurring job starts at the first hour, second hour, third hour (and so on up to 12 hours) after the midnight hour, depending on the value you select in the **Hour** field.

Copyright © 2017, 2026, Oracle and/or its affiliates.



For example, if you specify a value of **5** in the **Hour** field, the hours during which the job will run each day are 12 AM, 5 AM, 10 AM, 3 PM, and 8 PM. The start time indicates the hour and minute when the system should start honoring the schedule, and the end time indicates when the system should stop. So if the start time is 12:48 PM on 11/18, then that means 12:48 AM, 5:48 AM, and 10:48 AM have already occurred on the current day (11/18) and the next available timeslot is 3:48 PM, which is when the scheduled job will run for the first time. After that it will run at 8:48 PM on 11/18. However on 11/19 it starts over again and will run at 12:48 AM, 5:48 AM, 10:48 AM, and so on until the system reaches the specified end time (in this case, 12:48 PM on 11/19).

> **Note**
>
> When a job spans two days, the job runs during the midnight hour the next day and resets the recurrence pattern.

Table E-2 Timetable for Hourly Jobs

Hourly Recurrence	Timetable
1 (default)	AM: 12:00, 1:00, 2:00, 3:00, 4:00, 5:00, 6:00, 7:00, 8:00, 9:00, 10:00, 11:00 PM: 12:00, 1:00, 2:00, 3:00, 4:00, 5:00, 6:00, 7:00, 8:00, 9:00, 10:00, 11:00 For example, if the scheduled start time is 12:48 PM and you set the hourly recurrence for 1 hour, the job will run for the first time at 12:48 PM, and then again at 1:48 PM, 2:48 PM, 3:48 PM, and so on.
2	AM: 12:00, 2:00, 4:00, 6:00, 8:00, 10:00 PM: 12:00, 2:00, 4:00, 6:00, 8:00, 10:00 For example, if the scheduled start time is 12:48 PM and you set the hourly recurrence for 2 hours, the job will run for the first time at 12:48 PM, and then again at 2:48 PM, 4:48 PM, 6:48 PM, and so on.
3	AM: 12:00, 3:00, 6:00, 9:00 PM: 12:00, 3:00, 6:00, 9:00 For example, if the scheduled start time is 12:48 PM and you set the hourly recurrence for 3 hours, the job will run for the first time at 12:48 PM, and then again at 3:48 PM, 6:48 PM, 9:48 PM, 12:48 AM, 3:48 AM, and so on.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table E-2 (Cont.) Timetable for Hourly Jobs

Hourly Recurrence	Timetable
4	AM: 12:00, 4:00, 8:00 PM: 12:00, 4:00, 8:00 For example, if the scheduled start time is 12:48 PM and you set the hourly recurrence for 4 hours, the job will run for the first time at 12:48 PM, and then again at 4:48 PM, 8:48 PM, 12:48 AM, 4:48 AM, and so on.
5	AM: 12:00, 5:00, 10:00 PM: 3:00, 8:00 For example, if the scheduled start time is 12:48 PM and you set the hourly recurrence for 5 hours, the job will run for the first time at 3:48 PM, and then again at 8:48 PM, 12:48 AM, 5:48 AM, 10:48 AM, and so on.
6	AM: 12:00, 6:00 PM: 12:00, 6:00 For example, if the scheduled start time is 12:48 PM and you set the hourly recurrence for 6 hours, the job will run for the first time at 12:48 PM, and then again at 6:48 PM, 12:48 AM, 6:48 AM, and so on.
7	AM: 12:00, 7:00 PM: 2:00, 9:00 For example, if the scheduled start time is 12:48 PM and you set the hourly recurrence for 7 hours, the job will run for the first time at 2:48 PM, and then again at 9:48 PM, 12:48 AM, 7:48 AM, and so on.
8	AM: 12:00, 8:00 PM: 4:00 For example, if the scheduled start time is 12:48 PM and you set the hourly recurrence for 8 hours, the job will run for the first time at 4:48 PM, and then again at 12:48 AM, 8:48 AM, and so on.
9	AM: 12:00, 9:00 PM: 6:00 For example, if the scheduled start time is 12:48 PM and you set the hourly recurrence for 9 hours, the job will run for the first time at 6:48 PM, and then again at 12:48 AM, 9:48 AM, 6:48 PM, and so on.
10	AM: 12:00, 10:00 PM: 8:00 For example, if the scheduled start time is 12:48 PM and you set the hourly recurrence for 10 hours, the job will run for the first time at 8:48 PM, and then again at 12:48 AM, 10:48 AM, 8:48 PM, and so on.
11	AM: 12:00, 11:00 PM: 10:00 For example, if the scheduled start time is 12:48 PM and you set the hourly recurrence for 11 hours, the job will run for the first time at 10:48 PM, and then again at 12:48 AM, 11:48 AM, 10:48 PM, and so on.
12	AM: 12:00 PM: 12:00 For example, if the scheduled start time is 12:48 PM and you set the hourly recurrence for 12 hours, the job will run for the first time at 12:48 PM, and then again at 12:48 AM, and so on.

Copyright © 2017, 2026, Oracle and/or its affiliates.

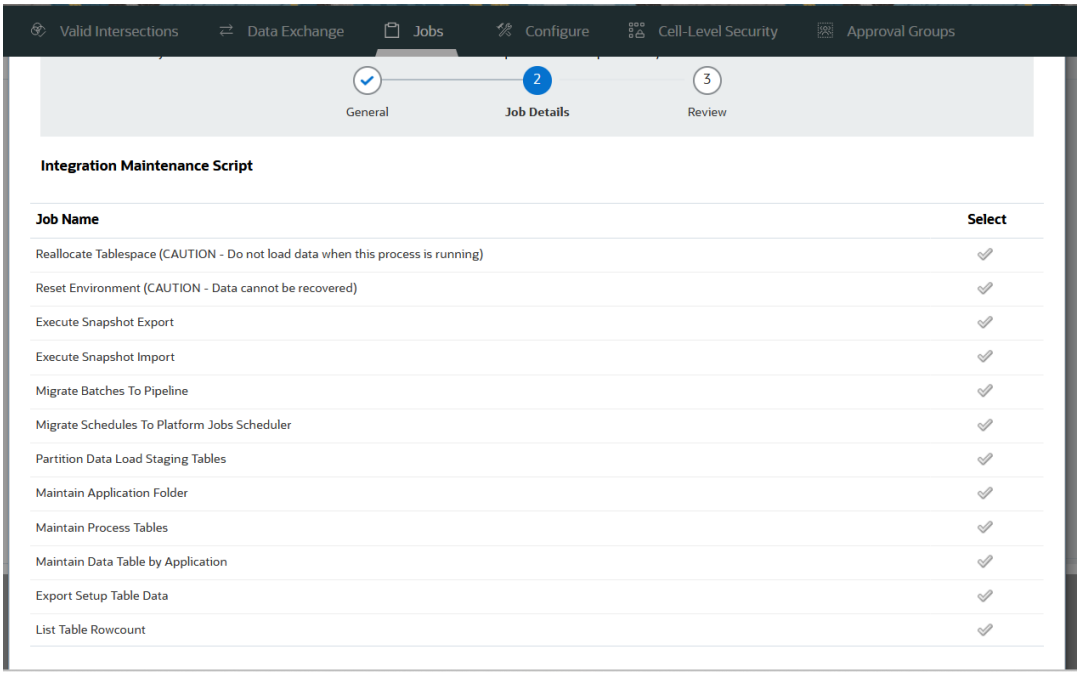
Scheduling Maintenance Scripts in the Job Scheduler

You can schedule the execution times of Data Management maintenance scripts in the Job Scheduler.

To schedule an Integration Maintenance script:

1. Click **Application**, then **Jobs**, and then **Schedule Jobs**.
2. From the **Schedule Jobs** page, then **What type of job is this?**, select **Integration Maintenance Script**.
3. To schedule an integration maintenance script to run now, see Scheduling Hourly Jobs .

   To schedule an integration maintenance script to run later, see Scheduling a Data Rule or Integration Definition to Run Later in the Job Scheduler.
4. Click **Next**.
5. From **Job Name**, select the integration maintenance script to schedule and click .



6. Complete the parameters for the System Maintenance scripts to be executed, and then click **OK**.

   For more information on the tasks and parameters used in System Maintenance scripts, see System Maintenance Tasks.

Partition Data Load Staging Tables

Select a table partitioning option when loading large volumes of data.

Use the Partition Data Load Staging Table script to select a table partitioning option when loading large volumes of data. Table partitioning addresses inconsistent data load times, and enables table partitioning to isolate each data load process in a table partition.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> Run the **Partition Data Load Staging Tables** script only when no other data load processes are running.

To partition data load staging tables:

1. Click **Application**, then **Jobs**, and then **Schedule Jobs**.
2. From the **Schedule Jobs** page, then **What type of job is this?**, select **Integration Maintenance Script**.
3. To schedule an integration maintenance script to run now, see Scheduling Hourly Jobs .

   To schedule an integration maintenance script to run later, see Scheduling a Data Rule or Integration Definition to Run Later in the Job Scheduler.
4. Click **Next**.
5. From **Integration Maintenance Script** page, click to the right of the **Partition Data Load Staging Tables** task to schedule.
6. From the **Partition Data Load Staging Tables** page, from **Mode**, select the partition table option.

   Available options:

   - **Enable**—Classifies a table in the cloud version of the Oracle Autonomous Database as "STAGING" and enables partitioning.
   - **Revert**—Turns off table partitioning.
7. Click **OK**.

Deleting an Integration

Delete an integration without having to delete individual components.

You can delete an integration including the name, import format, location, mappings and any data rules created in Data Integration. This option enables you to delete an entire integration without having to delete individual components.

To run the Delete Integration script:

1. Click **Application**, then **Jobs**, and then **Schedule Jobs**.
2. From the **Schedule Jobs** page, then **What type of job is this?**, select **Integration Maintenance Script**.
3. To schedule an integration maintenance script to run now, see Scheduling Hourly Jobs .

   To schedule an integration maintenance script to run later, see Scheduling a Data Rule or Integration Definition to Run Later in the Job Scheduler.
4. Click **Next**.
5. From **Integration Maintenance Script** page, click to the right of the **Partition Data Load Staging Tables** task to schedule.
6. From the **Delete Integration** page, and then from **Integration Name**, specify the name of the integration to delete.

Copyright © 2017, 2026, Oracle and/or its affiliates.

7. From the **Delete Smart Split Integrations Only** drop-down, select **Yes** to delete integrations created with the Smart Split option.

   A Smart Split job type enables you to define smaller data slices without having to create multiple integrations.

   For more information, see Using an Integration with Smart Split Job Type.

8. Click **OK**.

Export Setup Table Data

Export data from the Data Integration setup tables.

Use the Export Setup Table Data option to export the data from the following tables for debugging purposes:

- AIF_BALANCE_RULES
- AIF_BAL_RULE_PARAMS
- AIF_LCM_ARTIFACTS_V
- AIF_SOURCE_SYSTEMS
- AIF_TARGET_APPLICATIONS
- AIF_TARGET_APPL_DIMENSIONS
- AIF_TARGET_APPL_PROPERTIES
- TPOVPARTITION
- TBHVIMPGROUP
- TBHVIMPITEMFILE
- TPOVPERIOD
- TPOVPERIODADAPTOR
- TPOVPERIODSOURCE
- TPOVCATEGORY
- TPOVCATEGORYADAPTOR

To run the Export Setup Table Data script:

1. Click **Application**, then **Jobs**, and then **Schedule Jobs**.
2. From the **Schedule Jobs** page, then **What type of job is this?**, select **Integration Maintenance Script**.
3. To schedule an integration maintenance script to run now, see Scheduling Hourly Jobs .

   To schedule an integration maintenance script to run later, see Scheduling a Data Rule or Integration Definition to Run Later in the Job Scheduler.
4. Click **Next**.
5. From **Integration Maintenance Script** page, click to the right of the **Export Setup Table Data** task to schedule.
6. From the **Export Setup Table Data** page, review the job parameters and click **OK**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

List Table Rowcount

Show the number of rows in a Data Integration data table.

Use the List Table Rowcount option to show the number of rows in a data table.

This option lists table row counts from the following tables:

- TDATASEG
- TDATASEG_T
- TDATAMAP
- TDATAMAPSEG
- TLOGPROCESS
- TPOVPARTITION

To run the List Table Rowcount script:

1. Click **Application**, then **Jobs**, and then **Schedule Jobs**.
2. From the **Schedule Jobs** page, then **What type of job is this?**, select **Integration Maintenance Script**.
3. To schedule an integration maintenance script to run now, see Scheduling Hourly Jobs .

   To schedule an integration maintenance script to run later, see Scheduling a Data Rule or Integration Definition to Run Later in the Job Scheduler.
4. Click **Next**.
5. From **Integration Maintenance Script** page, click to the right of the **List Table Rowcount** task to schedule.
6. On the **List Table Rowcount** page, review the job parameters and click **OK**.

Maintain Application Folder

Purge files from the `inbox`, `outbox`, and `data` folder directories enabling you to specify a separate Days to Keep parameter for each of the folders.

The Maintain Application Folder process purges files from the `inbox`, `outbox`, and `data` folder directories. Data Integration accepts a separate Days to Keep parameter for each of the folders. If the value is not specified for a specific folder, Data Integration skips the folder.

By default, the system purges all files after 60 days. Logs are purged after 7 days.

In addition, Data Integration checks the `inbox` and `outbox` subdirectories under the respective folders and deletes any files. In the `data` subdirectory, Data Integration skips the `scripts` directory because it holds customer scripts.

To run the Maintain Application folder script:

1. Click **Application**, then **Jobs**, and then **Schedule Jobs**.
2. From the **Schedule Jobs** page, then **What type of job is this?**, select **Integration Maintenance Script**.
3. To schedule an integration maintenance script to run now, see Scheduling Hourly Jobs .

Copyright © 2017, 2026, Oracle and/or its affiliates.

To schedule an integration maintenance script to run later, see Scheduling a Data Rule or Integration Definition to Run Later in the Job Scheduler.

4. Click **Next**.
5. From **Integration Maintenance Script** page, click to the right of the **Maintain Application Folder** task to schedule.
6. From the **Maintain Application Folder** page, and then from the **Target Application** drop-down, select the target application to which to purge files.
7. In **Days to keep Inbox directory**, specify the number of days to keep files in the Inbox directory.
8. In **Days to keep Outbox directory**, specify the number of days to keep files in the Outbox directory
9. In **Days to keep Data directory**, specify the number of days to keep files in the Data directory.
10. Click **OK**.

Maintain Process Tables

Maintain Process tables.

This process maintains the following execution tables:

- AIF_PROCESSES
- AIF_PROCESS_DETAILS
- AIF_PROCESS_LOGS
- AIF_PROCESS_PARAMETERS
- AIF_PROCESS_PERIODS
- AIF_PROCESS_STEPS
- AIF_BAL_RULE_LOADS
- AIF_BAL_RULE_LOAD_PARAMS
- AIF_BATCH_JOBS
- AIF_BATCH_LOAD_AUDIT
- AIF_TEMP

To run the Maintain Process Tables task:

1. Click **Application**, then **Jobs**, and then **Schedule Jobs**.
2. From the **Schedule Jobs** page, then **What type of job is this?**, select **Integration Maintenance Script**.
3. To schedule an integration maintenance script to run now, see Scheduling Hourly Jobs .

   To schedule an integration maintenance script to run later, see Scheduling a Data Rule or Integration Definition to Run Later in the Job Scheduler.
4. From the **Integration Maintenance Script** page, click to the right of the **Maintain Process Table** task to schedule.
5. From the **Maintain Process Tables** page, and then from **Days to keep records**, specify the number of days to keep records in the execution tables.

Copyright © 2017, 2026, Oracle and/or its affiliates.

6. Click **OK**.

Maintain Data Table by Application

Purge archives of exported data for audit and drill by deleting the following tables:TDATAMAPSEG, TDATASEG, TPROCESSLOG.

The Data Integration data tables store archives of exported data for audit and drill down purposes. These tables can grow over a period of time but you can purge these tables as needed. The maintenance process deletes the following tables:

- TDATAMAPSEG
- TDATASEG
- TPROCESSLOG

The parameters are:

- Target Application
- Category
- Start Period
- End Period

To run the Data Table by Application script:

1. Click **Application**, then **Jobs**, and then **Schedule Jobs**.
2. From the **Schedule Jobs** page, then **What type of job is this?**, select **Integration Maintenance Script**.
3. To schedule an integration maintenance script to run now, see Scheduling Hourly Jobs .

   To schedule an integration maintenance script to run later, see Scheduling a Data Rule or Integration Definition to Run Later in the Job Scheduler.
4. Click **Next**.
5. From **Integration Maintenance Script** page, click to the right of the **Maintain Data Table by Application** task to schedule.
6. From the **Maintain Data Table by Application** page, and then from the **Target Application** drop-down, select the target application to which to purge files.
7. From **Start Period**, select the starting period from which to delete data.
8. From **End Period**, select the ending period from which to delete data.
9. From **Category**, select the category data to delete.

   To delete all category data, leave blank.
10. Click **OK**.

Maintain Data Table by Location

Delete data for a specific location by period or category.

Use Maintain Data Table by Location to delete data for a specific location by period or category.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> To delete all locations for an application, use the Maintain Data Tables by Application option. This option enables you to delete data across all locations associated with a selected target application. For more information, see Maintain Data Table by Application.

Maintain Setup Data uses a Mode parameter, which enables you to either preview or delete invalid data.

The parameters are:

- Location
- Start Period
- End Period
- Category

> **Note**
>
> If you need to delete multiple period mapping but you are unsure which integrations include period mappings, leave Location and Category blank if you want to delete all the period mappings.

To run the Maintain Data Table by Location task:

1. Click **Application**, then **Jobs**, and then **Schedule Jobs**.
2. From the **Schedule Jobs** page, then **What type of job is this?**, select **Integration Maintenance Script**.
3. To schedule an integration maintenance script to run now, see Scheduling Hourly Jobs .

   To schedule an integration maintenance script to run later, see Scheduling a Data Rule or Integration Definition to Run Later in the Job Scheduler.
4. Click **Next**.
5. From **Integration Maintenance Script** page, click to the right of the **Maintain Data Table by Location** task to schedule.
6. From the **Maintain Data Table by Location** page, and then from the **Location** drop-down, select the location from which to delete data.

   To delete data from all locations, leave the **Location** field blank.
7. From **Start Period**, select the starting period from which to delete data.
8. From **End Period**, select the ending period from which to delete data.
9. From **Category**, select the category data to delete.

   To delete all category data, leave this field blank.
10. Click **OK**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Maintain Setup Data

Use to view and delete an orphan row or invalid data that cannot be accessed or deleted from the user interface.

Use Maintain Setup Data to view and delete an orphan row or invalid data that cannot be accessed or deleted from the user interface. This data might include an orphan row from an import format, location, or data load rule. It can also include a duplicate row, or a Cloud user id or password defined for a local target application.

Maintain Setup Data uses a Mode parameter, which enables you to either preview or delete invalid data.

To run the Maintain Setup Data task:

1. Click **Application**, then **Jobs**, and then **Schedule Jobs**.
2. From the **Schedule Jobs** page, then **What type of job is this?**, select **Integration Maintenance Script**.
3. To schedule an integration maintenance script to run now, see Scheduling Hourly Jobs .

   To schedule an integration maintenance script to run later, see Scheduling a Data Rule or Integration Definition to Run Later in the Job Scheduler.
4. Click **Next**.
5. From the **Integration Maintenance Script** page, click to the right of the **Maintain Setup Data** task to schedule.
6. From the **Maintain Setup Data** page, and then from **Mode**, select either **Preview** or **Delete.**

   The Preview mode enables you to print the data to be deleted. The Delete mode enables you to delete the data.
7. Click **OK**.

Purge All Imported Data

Clear all imported data from your data tables.

Use Purge All Imported Data to clear all imported data from your data tables. This option enables you to clear data from the system so that you can start over when loading data.

When executed, the Purge All Imported Data deletes the following:

- Mapping audit used for loading
- Audit information about the data load
- Process load status
- Process parameters

> **Note**
>
> There is **no backup** to recover any purged data. It is recommended that you use extreme caution before executing this process.

Copyright © 2017, 2026, Oracle and/or its affiliates.

After the purge, note the following:

- You cannot view any data in Workbench for any POV.
- You cannot drill down from the Oracle Fusion Cloud Enterprise Performance Management to Data Integration.

> **Note**
>
> Drill regions are not deleted as part of the purge process. If you need to delete a drill region, then delete it manually.

- You cannot restore maps for the given POV.

> **Note**
>
> All setup data for example application registration, import format, and mapping are retained and not impacted by the purge process.

To run the Purge All Imported Data script:

1. Click **Application**, then **Jobs**, and then **Schedule Jobs**.
2. From the **Schedule Jobs** page, then **What type of job is this?**, select **Integration Maintenance Script**.
3. To schedule an integration maintenance script to run now, see Scheduling Hourly Jobs .

   To schedule an integration maintenance script to run later, see Scheduling a Data Rule or Integration Definition to Run Later in the Job Scheduler.
4. Click **Next**.
5. From the **Integration Maintenance Script** page, click to the right of the **Purge All Imported Data** task to schedule.
6. From the **Purge All Imported Data** page, and then from **Confirm Delete All Imported Data**, select **Y** (yes) to launch a confirmation message before executing a purge.

   Otherwise, enter **N** (No).
7. In **Reason**, specify the reason for executing the purge.
8. Click **OK**.

   The message: **Custom script executed initiated with Process ID: XXX** is displayed (where XXX is the process id number generated for job). You can access the log from the Process Details page.

Reallocate Tablespace

Free unused and space in your tables enabling you to free up space available for the storage of data.

Use the Reallocate Tablespace maintenance script to free unused and space in your tables. Reallocating tablespace enables you not to exceed the space quota defined for a specific table and free up space available for the storage of data.

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> Do not load data when executing this script.

To run the Reallocate Tablespace script:

1. Click **Application**, then **Jobs**, and then **Schedule Jobs**.
2. From the **Schedule Jobs** page, then **What type of job is this?**, select **Integration Maintenance Script**.
3. To schedule an integration maintenance script to run now, see Scheduling Hourly Jobs .

   To schedule an integration maintenance script to run later, see Scheduling a Data Rule or Integration Definition to Run Later in the Job Scheduler.
4. From the **Integration Maintenance Script** page, click to the right of the **Reallocate Tablespace** task to schedule.
5. From the **Run Script: Reallocate Table** page, click **OK**.

Reset Environment

Clear all Data Integration setup and data resetting the entire business process environment.

The Reset Environment script enables you to clear all Data Integration setup and data resetting the entire business process environment. You can then reconfigure the integration process as needed. This process accepts a **Yes** or **No** confirmation parameter and allows you to specify a reason for the reset.

> **Note**
>
> When the Reset Environment process is executed, data cannot be recovered.

To run the Reset Environment task:

1. Click **Application**, then **Jobs**, and then **Schedule Jobs**.
2. From the **Schedule Jobs** page, then **What type of job is this?**, select **Integration Maintenance Script**.
3. To schedule an integration maintenance script to run now, see Scheduling Hourly Jobs .

   To schedule an integration maintenance script to run later, see Scheduling a Data Rule or Integration Definition to Run Later in the Job Scheduler.
4. From the **Integration Maintenance Script** page, click to the right of the **Reset Environment** task to schedule.
5. From the **Reset Environment** page, and then from **Confirm Delete**, select **Y** (yes) to confirm the reset of the environment.

   Otherwise, enter **N** (No).
6. In **Reason for Delete**, specify the reason for resetting the environment.
7. Click **OK**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Upgrade Custom Applications

Migrate your existing custom target application to data export to file applications.

Use the Upgrade Custom Applications option to migrate your existing custom target application to data export to file applications. The migration converts existing file format from the custom target application to the file formats used in the data export to file option and retains all the existing setup. When the custom target application has been converted, you can run same data rule as before. You can run the migration for one custom application or all.

Before using this option, reconcile the differences in the file formats. For example, the header row in the Data Export to File contains the name of the dimension and not UD1, UD2 etc.

For more information about the Data Export to File option, see Registering a Data Export File Application.

To run the Upgrade Custom Applications script:

1. Click **Application**, then **Jobs**, and then **Schedule Jobs**.
2. From the **Schedule Jobs** page, then **What type of job is this?**, select **Integration Maintenance Script**.
3. To schedule an integration maintenance script to run now, see Scheduling Hourly Jobs .

   To schedule an integration maintenance script to run later, see Scheduling a Data Rule or Integration Definition to Run Later in the Job Scheduler.
4. From the **Integration Maintenance Script** page, click to the right of the **Upgrade Custom Applications** task to schedule.
5. From the **Upgrade Custom Applications** page, and then from the **Custom Application** drop-down, select the name of the custom target application to migrate from the LOV. To migrate all custom applications, select **All Custom Applications**.
6. Click **OK**.

Executing a Snapshot Export Job

Export all artifacts from setup and data folders including the ID columns to a ZIP file.

The data export process exports all artifacts from setup and data folders including the ID columns to a ZIP file. Artifacts are exported by POV and retain the ID of the setup artifact when data is exported and then imported to the target environment.

The snapshot export options supports: All, All Incremental, Incremental, and Setup export types.

To execute a snapshot export:

1. Click **Application**, then **Jobs**, and then **Schedule Jobs**.
2. From the **Schedule Jobs** page, then **What type of job is this?**, select **Integration Maintenance Script**.
3. To schedule an integration maintenance script to run now, see Scheduling Hourly Jobs .

   To schedule an integration maintenance script to run later, see Scheduling a Data Rule or Integration Definition to Run Later in the Job Scheduler.

Copyright © 2017, 2026, Oracle and/or its affiliates.

4. From the **Integration Maintenance Script** page, click to the right of the **Execute Snapshot Export** task to schedule.
5. On the **Execute Snapshot Export** page, from **Snapshot Type**, select the type of snapshot to export.

   Available snapshot types:

   - All—Includes all setup and historical artifacts.
   - All Incremental—Include only new or changed historical artifacts based on the POV since the last snapshot was exported and include setup artificats and all POVs (old and new) in the output file.
   - Incremental— Include only new or changed historical data based on the POV since the last snapshot was exported and include only setup artifacts and new POVs in the output file.
   - Setup—Includes only setup artifacts.
6. In **Snapshot File Name**, specify the file name of the ZIP with the ZIP extension to be written to the *outbox* for the snapshot export.

   For example, you might type: **DMSnapshot_082021.zip**.

   If a file doesn't end with a ZIP extension, the system forces a ZIP extension by appending a **.zip** suffix at the end of the file name.
7. In **Overwrite Output File**, specify **Yes** to overwrite a ZIP that has the same name as the one you are creating. Otherwise, specify **No**.

   > **Tip**
   >
   > If you specify an existing output file and specify **No** to overwrite the existing output file (ZIP), the job fails and the system shows the message: "Rule Execution did not complete successfully."
8. Click **OK**.

   One Execute Snapshot Exports job can be executed at a time.

   The message: **Custom script executed initiated with Process ID: XXX** is displayed (where *XXX* is the process id number generated for the job).

   You can access the job log from the Process Details page in Data Integration.

   You can view the snapshot export ZIP after the snapshot export has completed by downloading the output ZIP in Process Details.

   You can also download the snapshot ZIP using the EPM Automate downloadFile command.

Executing a Snapshot Import Job

Truncates all the setup and data artifacts and then import them.

During the snapshot data *import* process, the system first truncates all the setup and data artifacts and then imports them. Additionally, database sequences are reset based on the imported data so that no conflicts occur after the data is restored.

To execute a snapshot import:

1. Upload the snapshot export ZIP to the *inbox* in Data Integration.

Copyright © 2017, 2026, Oracle and/or its affiliates.

To upload the snapshot export ZIP to the *inbox* in Data Integration, use the EPM Automate uploadFile command.

You can also use the File Browser in Data Integration to upload a file to the Data Integration inbox.

2. Click **Application**, then **Jobs**, and then **Schedule Jobs**.
3. From the **Schedule Jobs** page, then **What type of job is this?**, select **Integration Maintenance Script**.
4. To schedule an integration maintenance script to run now, see Scheduling Hourly Jobs.

   To schedule an integration maintenance script to run later, see Scheduling a Data Rule or Integration Definition to Run Later in the Job Scheduler.
5. From the **Integration Maintenance Script** page, click to the right of the **Execute Snapshot Export** task to schedule.
6. On the **Execute Snapshot Import** page, from **Snapshot File Name**, specify the path and file name of the ZIP to be executed.

   For example, you might type: **inbox/DMSnapshot_082021.zip**.
7. Click **OK**.

   Only one Execute Snapshot Import job can be executed at a time.

   The message: **Custom script executed initiated with Process ID: 0**.

   You can access the job log from the Process Details page.

> **Note**
>
> The Process ID for the snapshot import is always " 0" only when the file is exported in ALL, INCREMENTAL or ALL_INCREMENTAL modes because each of these three exports contain data artifacts and process details are part of data tables.
>
> The Process ID is not "0" when only setup data is exported. This is because setup data doesn't contain data tables (Process details). In this case, the next available sequence in the Process ID is used.

Migrate Batches to Pipeline

Migrate a batch created in Data Management to the Data Integration Pipeline.

Use the Migrate Batches to Pipeline task to migrate a batch created in Data Management to the Data Integration Pipeline feature.

> **Note**
>
> For more information about the Pipeline Open Batch Job types, see Using Open Batch Job Types.

To run the migrate batches to the Pipeline:

1. Click **Application**, then **Jobs**, and then **Schedule Jobs**.

Copyright © 2017, 2026, Oracle and/or its affiliates.

2. From the **Schedule Jobs** page, then **What type of job is this?**, select **Integration Maintenance Script**.
3. To schedule an integration maintenance script to run now, see Scheduling Hourly Jobs .

   To schedule an integration maintenance script to run later, see Scheduling a Data Rule or Integration Definition to Run Later in the Job Scheduler.
4. Click **Next**.
5. From **Integration Maintenance Script** page, click to the right of the **Migrate Batches to Pipe** task to schedule.
6. From the **Migrate Batches to Pipe** page, and then from **Batch Name**, specify the name of the batch to migrate.
7. Click **OK**.

Migrating Schedules to EPM Platform Job Scheduler

To support the migration of all existing scheduled jobs in Data Management to the EPM Platform Job Scheduler console, use the "Migrate Schedules to Platform Jobs Scheduler" option.

> **Note**
>
> Scheduling functionality accessed from the user interface in Data Management has been deprecated.

Migrating Schedules to the EPM Platform Job Scheduler Important Considerations

The following are important actions and considerations when migrating scheduling jobs:

1. Before performing the Migrate Schedules to Platform Jobs Scheduler task, make a backup or Data Management Snapshot using the exportDataManagement EPMAutomate command or Executing a Snapshot Export Job in *Administering Data Management for Oracle Enterprise Performance Management Cloud* in System Maintenance tasks.

   The regular Lifecyle (LCM) snapshot backup does not include schedules. If you delete old schedules, there is no way to recover the old schedules without a Data Management snapshot.
2. Schedules are always migrated in Merge mode. Once a schedule is migrated, it is not updated again. However, you can delete the schedule and migrate it again. You can run the migration process repeatedly.
3. When you migrate the schedule, you have the option to delete the existing schedules from Data Management.

   If you choose not to delete existing schedules, then both the new and the old schedule execute at the same time and can cause data corruption issues.
4. Schedules are created in the EPM Platform Job Scheduler under the user running the migrate process. The schedules can be created only by an Application Administrator so always run the migrate process using a user with Planning Administrator or Consolidation Administrator roles.

Copyright © 2017, 2026, Oracle and/or its affiliates.

5. After schedules are migrated, a job is executed under the user who migrated the schedule. Jobs are always run as an Administrator. If the original schedule was created by a Power User or by a user with a Run Integration role, jobs are run in an elevated role. If there are requirements to enforce security or other validations applicable to non-admin users, then those validations are not enforced. Oracle will consider adding support for running scheduled jobs as non-admin user if there is demand in a future release.
6. Schedules with execution times defined as Last Day or Last Day - 1 are not supported in the EPM Platform Job Scheduler. If you have requirement to run the jobs on the last day of the month, then schedule a separate job for each month.

Migrating Schedules Best Practices

The following are best practices when migrating scheduling jobs:

1. Execute the migrate schedule job in Preview mode. Review the process log to understand how the schedules will be migrated.
2. When satisfied with the details of a migrated schedule job run in Preview, take a Data Management snapshot of the Setup Data only.

   For more information, see: Executing a Snapshot Export Job in *Administering Data Management for Oracle Enterprise Performance Management Cloud* .
3. Execute the Migrate Schedule process and set the **Delete Old Schedules** to **No**.
4. Review the newly created schedules.
5. If there are issues with the newly created schedules, delete the invalid schedules from the EPM Platform Job Scheduler console.

   When any unrecoverable errors occur, restore the Data Management snapshot and execute the migration again.
6. Fix the appropriate source schedules and rerun the process.
7. If everything looks OK, run the Migrate Schedule by setting the **Delete Old Schedule** option to **Yes**.

   Run the **Delete Old Schedule** option before the first scheduled job is supposed to execute.

Migrating Schedules to the EPM Platform Job Scheduler Process Description

The following are the steps to migrate scheduling jobs to the EPM Platform Job Scheduler. Before beginning the migration, familiarize yourself with the content at: Migrating Schedules Best Practices and Migrating Schedules to the EPM Platform Job Scheduler Important Considerations.

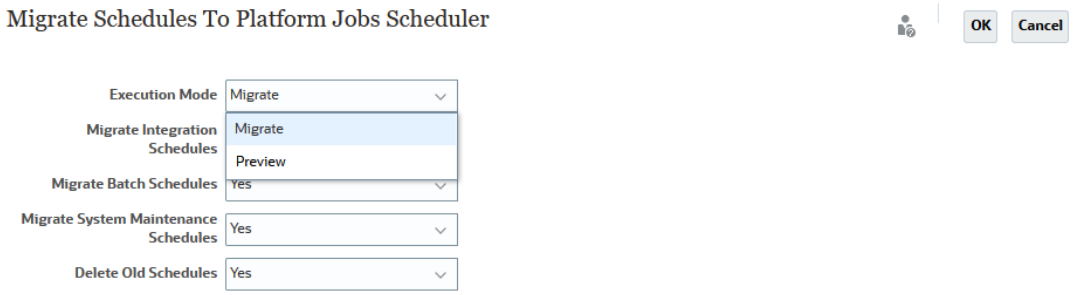
To migrate a job schedule:

1. Click **Application**, then **Jobs**, and then select **Schedule Jobs**.
2. From the **Schedule Jobs** page, then **What type of job is this?**, select **Integration Maintenance Script**.
3. Select **Migrate Schedules to Platform Jobs Scheduler** and then click **Next**.
4. Select the following parameters:

Copyright © 2017, 2026, Oracle and/or its affiliates.

Table E-3 Migrate Job Schedule Parameters

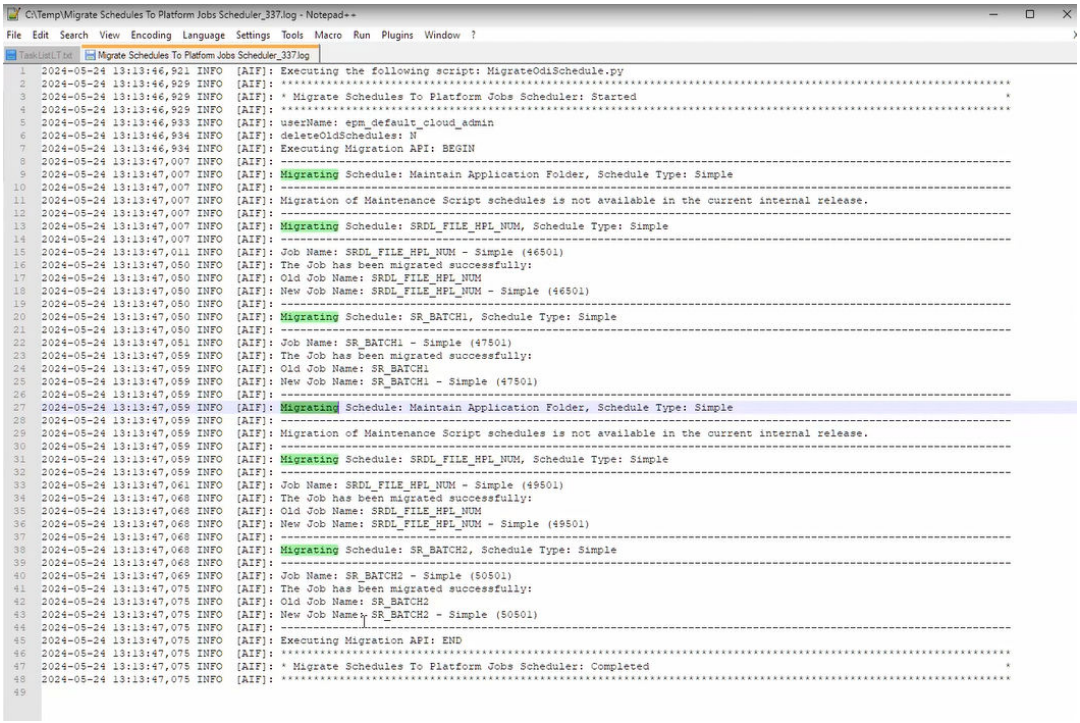
Parameter	Value	Description
Execution Mode	Options: • Preview • Migrate The default value is Preview.	In Preview mode the system provides details of all schedules that to be migrated, but will not perform the actual migration. In Migrate mode the system performs the actual migration.
Migrate Integration Schedules	Options: • Yes • No Default value is **Yes**.	Select **Yes** to migrate existing integration schedules.
Migrate Batch Schedules	Options: • Yes • No Default value is **Yes**.	Select **Yes** to migrate existing batch schedules.
Migrate System Maintenance Schedules	Options: • Yes • No Default value is **Yes**.	Select **Yes** to migrate existing System Maintenance Tasks schedules.
Delete Old Schedules	Options: • Yes • No Default value is **No**.	Select **Yes** to delete existing schedules from Data Management. If the schedules are not deleted, then both the old and new schedules are executed.



5. Click **OK**.
6. From the **Data Integration** home page, then the **Actions** drop-down, select **Process Details**.
7. To view a report of the migrated jobs, from the **Workflow** tab, under **Monitor**, select **Process Details**.
8. Click **Show** next to the migrated job schedule.

   The following image shows a migrated job schedule report link in Process Details.

Copyright © 2017, 2026, Oracle and/or its affiliates.



> **Note**
>
> To change the level of detail for the migrated jo b schedule report in Process Details, change the **Log Level** settings in **User Settings**. Setting the log level to **5** shows the most detail on the report. For more information, see System Settings.

Copyright © 2017, 2026, Oracle and/or its affiliates.

F

Frequently Asked Questions (FAQ) about Data Integration

This FAQ provides links to resources for frequently asked questions about administrative tasks in Data Integration and Data Management.

Frequently Asked Questions

- What to do if the EPM Integration Agent fails to start with a 401 Unauthorized error?
- What to do if the EPM Integration Agent does not extract data?
- EPM Integration Agent - Teradata DB - ERROR [AIF]: Unexpected Error In ImportData: EPM Agent Failed To Extract Data
- "java.lang.RuntimeException", EPM Agent Failed to Extract Data
- EPM Agent Failed to Extract Data When Using Credential Store Set to File in Datasource Target Application
- Integration with Cloud environment is Failing with a REST Connection Error
- When attempting to initialize the Fusion GL Application as a Source System in Data Management, it fails with an error.
- How to Resolve a Fetch of Driver Member Failure Error when Exporting Data to Financial Consolidation and Close?
- Fusion ERP GL Data Load Fails with Error: "Client Commands are Currently Not Being Accepted"
- How To Set Log Level To 5 In Data Management
- How to address source initialization errors when integrating with Oracle General Ledger in Data Management?
- When attempting to initialize the Fusion GL Application as a Source System in Data Management, it fails with an error.
- Data Management Load Issue Error "ERROR [AIF]: Server returned HTTP response code: 401"
- In Data Management an ERROR_INVALID_PERIOD Error is Thrown When Executing The Data Load Rule
- Unknown Member When Trying to Import the Data from Fusion Financials to Data Management
- Extracting the BI Publisher Report From Source Fusion ERP Through Data Management Fails with Error
- How do I Modify an E-Business Suite E-Business Suite General Ledger GL Preseeded Query?

Copyright © 2017, 2026, Oracle and/or its affiliates.

> **Note**
>
> For a range of Frequently Asked Questions (FAQs) along with their answers, covering various topics related to Oracle Fusion Cloud Enterprise Performance Management, as found in multiple Cloud EPM documentation resources, see: Frequently Asked Questions in *Getting Started with Oracle Enterprise Performance Management Cloud for Administrators*

What to do if the EPM Integration Agent fails to start with a 401 Unauthorized error?

The `CLOUD_DOMAIN` parameter is required except for any of the Gen 2 data centers. When using the EPM Integration Agent with Gen 2 data centers, this error can be resolved by removing the domain in the `CLOUD_DOMAIN` field. For other domains, the `CLOUD_DOMAIN` parameter must be specified in the agentparams.ini file. For details, see Configuring the EPM Integration Agent Startup Parameters.

What to do if the EPM Integration Agent does not extract data?

In some cases, this may be caused by entering the JDBC URL incorrectly. Ensure that the URL is entered in the format described in Extracting Data from On-premises Data Sources.

EPM Integration Agent - Teradata DB - ERROR [AIF]: Unexpected Error In ImportData: EPM Agent Failed To Extract Data

When trying to import data using the EPM Integration Agent connected to a Teradata database, the EPM Integration Agent fails to extract data. The error occurred due to an incorrect JDBC URL syntax.

Use the following JDBC URL for a Teradata Database: `jdbc:teradata://database=<DB_Name>`. Then update this line in the `agentparams.ini` file: `CUSTOM_CLASS_PATH=C:\Oracle\EPMAgent\lib\*`

For details, see Configuring the EPM Integration Agent Startup Parameters.

"java.lang.RuntimeException", EPM Agent Failed to Extract Data

User is not able to extract data from Profitability and Cost Management to a database. The first step Import and Validate work correctly but the, Export phase generates the following error:

```
ERROR [AIF]: Failed request or response. Request status:QUEUED
INFO [AIF]: Appending EPM agent log to process: BEGIN
DEBUG [AIF]: Updating Agent Processes: FAILED
INFO [AIF]: Appending EPM agent log to process:: END
ERROR [AIF]: Unexpected error in importData:EPM agent failed to extract data.
ERROR [AIF]: java.lang.RuntimeException: EPM agent failed to extract data.
ERROR [AIF]: The script has failed to execute:
FATAL [AIF]: Error in Comm.executeJythonScript
```

This error occurred because the file name (appname.cred) does not exactly match the data source application name.

To fix the issue:

1. Match the file name (appname.cred) exactly with the data source application name.

Copyright © 2017, 2026, Oracle and/or its affiliates.

2. The JDBC URL should be in the format: `jdbcurl=jdbc:oracle:thin:@host:port/service`.

EPM Agent Failed to Extract Data When Using Credential Store Set to File in Datasource Target Application

EPM agent failed to extract data when using a credential store set to a file in the data source target application. The system returns the error message: `Exception: java.lang.RuntimeException: EPM agent failed to extract data.`

The issue results from the format of the JDBC URL

To fix the issue:

1. Match the file name (appname.cred) exactly with the data source application name.
2. The JDBC URL should be in the format: `jdbcurl=jdbc:oracle:thin:@host:port/service`.

How to Resolve a Fetch of Driver Member Failure Error when Exporting Data to Financial Consolidation and Close?

This error may occur when the member is not valid or has no mapping. To resolve this error, define a valid member name, then define the data load mapping in the data file.

Fusion ERP GL Data Load Fails with Error: "Client Commands are Currently Not Being Accepted"

During a data load from the Fusion ERP General Ledger, the load fails with following errors:

```
INFO [AIF]: isHarnessTest():: harness flag enable ? false
INFO [AIF]: getOlapserver() :: = fa.oracleoutsourcing.com:11448
INFO [AIF]: ExtractBalanceData:Error::Cannot open cube view. Essbase
Error(1051544): message on contacting or from application:
INFO [AIF]: [Wed Mar 9 02:02:53 2022]Local////140107914331904/Error(1013204)
INFO [AIF]: Client Commands are Currently Not Being Accepted
```

or

```
INFO [AIF]: ExtractBalanceData:Error::Cannot perform cube view operation.
Essbase Error(1024015): Request [MdxReport] from user
[FUSION_APPS_GL_ESSBASE_APPID] was terminated by an admin user
```

This error occurred because a Publish Hierarchy job was scheduled to run at the same time. Publish Hierarchy locks the cube for update. This causes the issue with the balance extract ESS job.

To fix this error, change the process timings to ensure that data load from the Fusion GL to Cloud EPM and the Fusion the Publish Hierarchy job do not overlap.

Integration with Cloud environment is Failing with a REST Connection Error

Cloud to cloud integration is failing with a REST connection error. It shows in the process log like this:

```
ERROR [AIF]: https://epmserver.us2.oraclecloud.com//interop/rest/11.1.2.3.600/
applicationsnapshots/MS_IND_2192.log/contents
```


Copyright © 2017, 2026, Oracle and/or its affiliates.

or

`Error encountered while downloading file from remote server` or any error where you see double slashes in the log after the server name.

REST API no longer requires that the URL have a trailing slash. So instead of specifying: `https://epmserver.us2.oraclecloud.com/`, specify: https://epmserver.us2.oraclecloud.com.

How To Set Log Level To 5 In Data Management

To set the log level to "5" (show the most detail) in Data Management, specify **5** for the **Log Level** in System Settings. To set the log level in Data Management, see Setting System-Level Profiles in *Administering Data Management for Oracle Enterprise Performance Management Cloud* . To set the log level in Data Integration, see System Settings.

How to address source initialization errors when integrating with Oracle General Ledger in Data Management?

When integrating applications such as Oracle General Ledger created in the Oracle ERP Cloud with a Cloud EPM application, you first create and register the source system specifying the application type: Oracle ERP Cloud. For details, see Registering Oracle ERP Cloud Source Systems in *Administering Data Management for Oracle Enterprise Performance Management Cloud* .

Find additional helpful information in *Administering Data Integration*:

- Registering Applications
- Configuring a Source Connection

When attempting to initialize the Fusion GL Application as a Source System in Data Management, it fails with an error.

The required "Integration User" privileges in the Oracle ERP Cloud used for the Oracle General Ledger and Cloud EPM integration are missing.

The roles required in Oracle ERP Cloud for the "Integration User" used for the GL to Cloud EPM are the following:

- GL_RUN_TRIAL_BALANCE_REPORT_PRIV – To Import Data from GL to EPM
- GL_ENTER_BUDGET_AMOUNTS_FOR_FINANCIAL_REPORTING_PRIV – To Write-back data from EPM to GL
- FUN_FSCM_REST_SERVICE_ACCESS_INTEGRATION_PRIV – Privilege to execute REST API used to perform the integration

> **Note**
>
> If you are using a custom role or one of the roles not listed below, then you must add the FUN_FSCM_REST_SERVICE_ACCESS_INTEGRATION_PRIV to the role for the integration to work.

If you are using one of the following pre-defined roles for the "Integration User," then no further action is required.

- For Import Data
- General Accountant

Copyright © 2017, 2026, Oracle and/or its affiliates.

- Journal Management
- Period Close Management
- For Import and Write-back Data
- General Accountant
- Financial Analyst

For more information, see Security Role Requirements for Oracle ERP Cloud Integrations.

Data Management Load Issue Error "ERROR [AIF]: Server returned HTTP response code: 401"

Exporting actual data from Financial Consolidation and Close to EPBCS is failing.

This error occurs when Data Management or Data Integration is unable to consider changes in the target application.

To fix this issue, refresh members in the target application. To do this, in Data Integration, go to **Applications**, select the target application, then click ... to the right of the application, and then select **Refresh Members**. Then perform the load again.

For more information, see Refreshing Members.

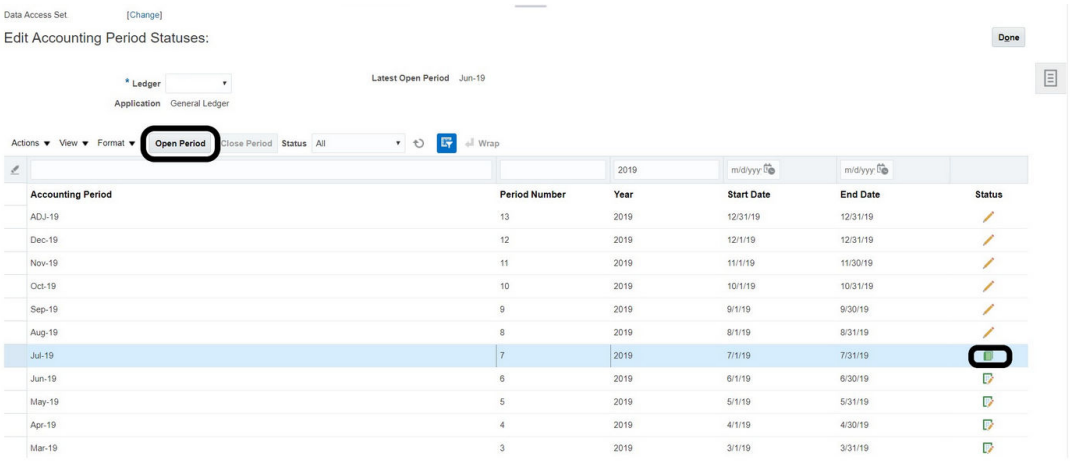
In Data Management an ERROR_INVALID_PERIOD Error is Thrown When Executing The Data Load Rule

When pulling data from the Financials General Ledger into Data Management, the "Transformation" step fails with the following error: `ERROR_INVALID_PERIOD.............. Invalid period specified in the file.` This error occurs even though the periods were mapped correctly and data was successfully loaded into the previous period.

The step failed because the Period was not open on the Financials side from which they were extracting data.

To fix this issue:

1. Log into the Financials General Ledger/data source.
2. Open the period, for example, as shown below:



3. Execute the data load rule again in Data Management.

Copyright © 2017, 2026, Oracle and/or its affiliates.

Unknown Member When Trying to Import the Data from Fusion Financials to Data Management

Users are unable to import the data from Fusion Financials to Data Management. It fails with the following error:

```
getOlapserver() :: =
bi.oracleoutsourcing.com:10215
ExtractBalanceData:Error::Cannot perform cube view operation. Essbase
Error(1260046): Unknown Member [ACC].[XXX] used in query
```

The error results when you do not enter a Fully Qualified Name (FQN) in the Source Filter option in the Data Load Rule to pull the parent child relationship members data from Fusion GL to Data Management.

When there are same parent or child members available in different hierarchy, you are required to use the Fully Qualified Name (FQN). Use the Essbase member function to select the leaf level members.

Data Management Export Error, ExtractBalanceData:Error: Cannot perform cube view operation. Essbase Error(1200766)

When a Data Management export processes with more than 10,00,000 of rows, the following error occurs:

```
ExtractBalanceData:Error::Cannot perform cube view operation. Essbase
Error(1200766): The number of query result cells exceeds the limit [1000000] set
by the QUERYRESULTLIMIT configuration setting
```

Extracting the BI Publisher Report From Source Fusion ERP Through Data Management Fails with Error

When a Oracle Business Intelligence Publisher report has been created and registered as an ESS Job and an integration is defined in Data Management, extracting a the report from a source Fusion ERP through Data Management fails with the following error: "Failure: Due to oracle.xdo.server.ReportException: oracle.xdo.servlet.CreateException: java.lang.SecurityException: Security violation."

The job fails because the customer does not have correct privilege.

To correct the issue:

1. Login to the source Fusion ERP application.
2. Add the privilege GL_RUN_TRIAL_BALANCE_REPORT_PRIV_OBI to the custom duty/ role the user has and then re-run the report.

How do I Modify an E-Business Suite E-Business Suite General Ledger GL Preseeded Query?

You can't modify the preseeded query for Oracle E-Business Suite, but you can copy the query and create a new application based on the new query.

Copyright © 2017, 2026, Oracle and/or its affiliates.