Oracle® Essbase

Using the Essbase Web Interface

F17137-26
August 2026

Oracle Essbase Using the Essbase Web Interface,

F17137-26

Copyright © 2019, 2026, Oracle and/or its affiliates.

Primary Author: Essbase 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

1 Access Oracle Essbase

New Features in the Essbase Web Interface 1
- Quick Access to Open a Cube, Open an Outline, or Analyze Data 2
- Bookmark an Active Browser Tab for Quick Access 3
- Open Artifacts in a New Browser Tab or in an Editor 3

Essbase, REST, and Smart View Client URLs 4
Set Up Your Client 5
Ask Essbase 6
- Overview of Ask Essbase 6
- Configure Ask Essbase 7

AI Query 9

2 Manage Essbase Applications

Create an Application and a Database using the Essbase web interface 1
Set Application-Level Configuration Properties 2

3 Manage Essbase Cubes

Create an Essbase Cube 1
Write and Edit Scripts 2
Create Calculation Scripts 2
Report on Data 4
- Create Report Scripts 4
- Execute Report Scripts 5

4 Understand Your Access Permissions in Essbase

User Role 2
- Database Access Permission 3
- Database Update Permission 3
- Database Manager Permission 3
- Application Manager Permission 4

Power User Role 5

Copyright © 2019, 2026, Oracle and/or its affiliates.

Service Administrator Role 5
Filter Access to Data 5
Create a Filter using the Essbase web interface 6
Create Efficient Dynamic Filters 7
Dynamic Filter Syntax 7
Workflow to Create Dynamic Filters 9
Example of a Dynamic Filter 9

5 Manage Essbase Files and Artifacts

Explore the Essbase Files Catalog 1
Explore the Gallery Templates 1
Applications Templates 2
Technical Templates 3
System Performance Templates 3
Access Files and Artifacts 3
Explore the Application Directories 4
Work with Files and Artifacts 5
Specify Files in a Catalog Path 6

6 Create and Manage Cube Outlines Using the Web Interface

View and Edit Outline Properties for a Newly Created Cube 1
Work with General and Attribute-related Outline Properties 2
Understand and Create Alias Tables 5
Understand and Work with Dynamic Time Series Outline Properties 6
Understand and Create Textual Measures 7
Add Dimensions and Members to Outlines 7
Add Dimensions to Outlines Manually 8
Add Members to Outlines Manually 9
Name Generations and Levels 10
Restructure Cubes 11
Create Attribute Dimensions and Members 12
About Duplicate Member Names 13
Set Dimension and Member Properties 14
Open the Outline in Edit Mode 14
Set Member Properties while in Edit Mode 15
Set Properties in the Member Inspector 15
Set General Properties 16
Create Aliases 19
Create Member Formulas 20
Set Attribute Associations 22

Copyright © 2019, 2026, Oracle and/or its affiliates.

Create User-Defined Attributes 23
Select the Member Properties to Display in the Outline 24
Compare Outlines 25
Copy and Paste Members Within and Between Outlines 29

7 Manage Application and Cube Artifacts and Settings

Set Calculation-Related Cube Properties 1
Unlock Objects 2
Remove Data Locks 2
Set Buffer Sizes to Optimize Reports 3
Understand Transaction Semantics in Essbase 3
Ask Essbase 5

8 Run and Manage Jobs Using the Web Interface

View Job Status and Details 1
Execute Jobs 1
Build Aggregations 2
Clear Aggregations 4
Export to Table Format 4
Run Calculation 4
Build Dimension 5
Clear Data 6
Export Data 6
Export Excel 7
Export LCM 8
Import LCM 10
Load Data 11
Run MDX 14
Vectorize Outline 15

9 Work with Connections and Datasources

Create an Application-Level Connection and Datasource 2
Create a Global Connection and Datasource 4
Create a Connection and Datasource for a File 6
Create a Connection and Datasource to Access Another Cube 9
Create a Connection and Datasource to Access Oracle Database 12
Create a Connection and Datasource for Autonomous Data Warehouse 15
Create Connections and Datasources for Generic JDBC Drivers 19
More Connection Examples for Generic JDBC Drivers 22

Copyright © 2019, 2026, Oracle and/or its affiliates.

Implement Parameters for Datasources 25
Set a Default Parameter in a Datasource 26
Use Substitution Variables in a Datasource 29
Build Dimensions and Load Data 33
Create a GenAI Connection 33

10 Analyze Data in the Web Interface

Perform Ad Hoc Analysis in the Web Interface 1
Work with Layouts 2
Access to Layouts 3
Analyze and Manage Data with MDX 3
Analyze Data with MDX Reports 3
Access to MDX Reports 4
Examples of MDX Reports 4
Insert and Export Data with MDX 6
Run MDX Scripts 6
Write, Upload, and Run an MDX Script 7
Write an MDX Script in the Script Editor and Run It 7
Create an MDX Script in Cube Designer and Run it 8
Guidelines for MDX Scripts 8
Examples of MDX Scripts 8
Access External Data with Drill Through Reports 10
Introduction to Essbase Drill Through 10
Drill Through Terminology 12
Workflow for Drill Through Report Design 13
How Drill Through Works 14
Drill Through Report Definition 15
Drill Through Use Case Example 19
Access to Drill Through Reports 32
Design Drill Through Reports 32
General Considerations for Designing Drill Through Reports 33
Define Column Mappings for Drill Through Reports 33
Define Drillable Regions for Drill Through Reports 35
Implement Parameters for Drill Through Reports 38
Test Drill Through Reports 40
Drill Through to a URL 42
Drill Through from Multiple Cells 45
Debug Drill Through using Essbase Server Platform Log 49

Copyright © 2019, 2026, Oracle and/or its affiliates.

11 Calculate Cubes

Access to Calculations 1
Create Calculation Scripts 3
Execute Calculations 4
Use Substitution Variables 5
Set Two-Pass Calculation Properties 8
Trace Calculations 9
Calculate Selected Tuples 12
- Use Case for Tuple Calculation 12
- Understand Tuple-Based Calculation 14
- Select Tuples for Point of View Calculation 14
- Examples of Tuple Selection to Reduce Calculation Scope 15
  - No Tuple Selection 16
  - Selection of Named Sparse Dimensions 16
  - Selection of Contextual Sparse Dimensions 17

12 Model Data in Private Scenarios

Understand Scenarios 1
- View and Work with Scenario Data 2
  - View and Work With Scenario Data From the Essbase Web Interface 2
  - View and Work With Scenario Data From a Smart View Private Connection 3
- About Scenario Calculations 4
- About Data Loads to Scenario-enabled Cubes 4
- About Data Exports from Scenario-enabled Cubes 5
- About Transparent and Replicated Partitions in Scenario-enabled Cubes 5
- About @XREF/@XWRITE in Scenario-enabled Cubes 5
- About Audit Trail in Scenario-enabled Cubes 6
- About Scenario Limitations 7

Enable Scenario Modeling 7
- Create a Scenario-Enabled Cube 8
- Create a Scenario-Enabled Sample Cube 8
- Enable an Existing Cube for Scenario Management 8
- Create Additional Sandbox Members 9

Scenario Workflow 9
- Enable Email Notifications for Scenario Status Changes 10
- Create a Scenario 11
- Model Data 12
- Submit a Scenario for Approval 13
- Approve or Reject Scenario Changes 13
- Apply Data Changes 14

Copyright © 2019, 2026, Oracle and/or its affiliates.

Copy a Scenario 14
Delete the Scenario 15
Understand Scenario User Roles and Workflow 15
Work with Scenarios 16
View Base Member Data 16
Compare Scenario Values to Base Values 17
Set Scenario Cells to #Missing 18
Revert Scenario Values Back to Base Values 19
Understand When to Aggregate Sandbox Dimensions 20
Example: Calculate Scenarios with Dynamic Upper Level Members 20
Example: Calculate Scenarios with Stored Upper Level Members 21

13 Audit Data, Security, Artifact Changes, and LCM Events

Track Data Changes 1
Turn on Data Audit Trail and View the Data Audit Trail 1
Link a Reporting Object to a Cell 2
Export Logs to a Sheet 3
Refresh the Audit Log 3
View and Manage Audit Trail Data in the Essbase Web Interface 3
Audit Security, Artifact Changes, and LCM Events 4
Workflow to Enable Security Auditing for Essbase Server 4
About the Auditing Policy File 6
Security Auditing Events 8

14 Link Cubes Using Partitions or @XREF/@XWRITE

Define a Reusable Connection for Partitions or Location Aliases 1
Understand Transparent and Replicated Partitions 2
Create a Transparent Partition 2
Create a Replicated Partition 4
Refresh a Replicated Partition 5
Export and Import Transparent and Replicated Partitions 5
Understand @XREF/@XWRITE 6
Create a Location Alias 7

15 Federated Cubes: Integrate Essbase with Autonomous AI Database

Prerequisites for Federated Cubes 2
Federated Cube Deployment Workflow 7
Provision Autonomous AI Lakehouse for Federated Cubes 8
Deploy Essbase from Marketplace for Federated Cubes 9

Copyright © 2019, 2026, Oracle and/or its affiliates.

Create a Schema for Federated Cubes 11
Identify the Pivot Dimension and Set up the Fact Table 12
Identify the Pivot Dimension 12
Create the Fact Table 13
Create a Connection for Federated Cubes 15
Set up Credentials for Federated Cubes 18
Create a Federated Partition 21
Data Load Options for Federated Cubes 26
Calculate and Query Federated Cubes 28
Federated Cube Maintenance and Troubleshooting 31
Model and Test Federated Cubes 32
Metadata Precautions for Federated Cubes 33
Cancel Long Running SQL on Federated Cubes 34
What to Do if the Database Connection Details Changed 37
Back up and Restore a Federated Cube 41
Remove a Federated Partition 42
Differences between Aggregate Storage, Block Storage, and Federated Cubes 43
Restrictions for Federated Cubes 46

16 Configure Oracle Essbase

Set Provider Services Configuration Properties 1

17 Download Utilities and Perform Tasks in the Console

Access Tools and Tasks from the Console 1
Enable Antivirus Scanning in Essbase 1

18 Use Logs and View Notifications

Download Application Logs 1
About Performance Analyzer 1
Enable Performance Analyzer and Choose Settings 2
Understand and Work With Performance Analyzer Data 2
View and Manage Notifications in the Essbase Web Interface 3

A Centralized Smart View URL, and Read-Only Clusters

Access Multiple Essbase Servers Using a Centralized Smart View URL A-2
Configure and Manage Active-Active (Read-Only) Essbase Clusters A-4
Access Multiple Essbase Servers in EPM Shared Services A-6

Copyright © 2019, 2026, Oracle and/or its affiliates.

Accessibility and Support

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 © 2019, 2026, Oracle and/or its affiliates.

1
Access Oracle Essbase

Oracle Essbase is a business analytics solution that uses a proven, flexible, best-in-class architecture for analysis, reporting, and collaboration. Essbase delivers instant value and greater productivity for your business users, analysts, modelers, and decision-makers, across all lines of business within your organization.

Access Essbase using credentials supplied by your Service Administrator.

To access Essbase, you must have the following information:

- URL to access the Essbase web interface
- User name
- Password
- Identity domain to which you belong

After you log in to the Essbase web interface, the Applications page is displayed.

New Features in the Essbase Web Interface

The following new features add to the usability of the Essbase web interface.

> **Note**
>
> Redwood, the default Essbase web interface is Oracle's standard interface with a modern look and feel. Starting in 21.8, the older Classic interface is unavailable.

The updated interface features a new Redwood look and feel, with enhanced functionality in many areas:

- You work in the context of the application or cube instead of working in the application and cube inspectors.
- Objects such as scripts or the outline are opened in tabs, so you can easily toggle between them.
- Functionality for adding members is enhanced. You can provide, in one dialog, all the general properties like storage type, dimension type, solve order, formulas, attributes, and UDAs. You can also add the multiple members consecutively using the shortcut 'Ctrl + Enter' or by clicking the Add button without closing the dialog.
- Search capability is enhanced, with improved performance and access to the following operations in the Search dialog: Locate, Member inspector, and Hierarchy (Shows the hierarchy of the member in a tree view, where you can locate, launch the member inspector, or add a sibling or a child, if the outline is in Edit mode).
- Compare two outlines side-by-side. See Compare Outlines.

Copyright © 2019, 2026, Oracle and/or its affiliates.

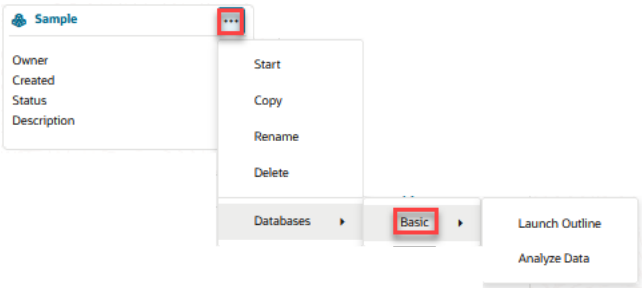
- Copy and Paste Members Within and Between Outlines. This can be between two different outlines, shown on different browser tabs, or from a source to a target outline when using outline compare.
- View notifications from your active session. See View and Manage Notifications in the Essbase Web Interface.
- Write and edit Essbase scripts using enhanced script editing capabilities in the Redwood Interface. See Write and Edit Scripts.

Quick Access to Open a Cube, Open an Outline, or Analyze Data

In the Essbase web interface, you now have quick access to open a cube, open an outline, or analyze data.

You can open a cube from the Home page.

1. On the Home page, find the application.
2. Select the Actions menu for that application ••• .
3. Choose **Databases** > *database_name*.



You can open an outline or analyze data in the Essbase web interface in two ways. You can open outlines or analyze data from within the cube,

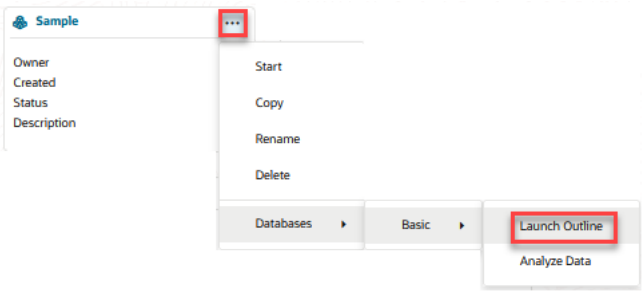
1. Open the application.
2. Open the cube.
3. Click **Launch Outline** or **Analyze Data**.

Now you can also open them from the Home page.

To open an outline from the Home page.

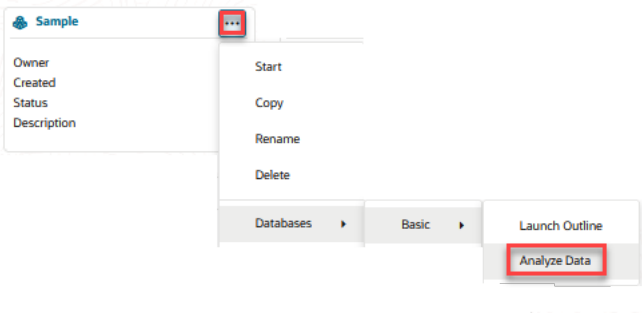
1. On the Home page, find the application.
2. Select the Actions menu for that application ••• .
3. Choose **Databases** > *database_name* > **Launch Outline**.

Copyright © 2019, 2026, Oracle and/or its affiliates.



To open analyze data from the Home page.

1. On the applications page, find the application.
2. Select the Actions menu for that application ••• .
3. Choose **Databases** > *database_name* > **Analyze Data**.



Bookmark an Active Browser Tab for Quick Access

Using your browser's bookmark feature, you can bookmark specific applications, cubes, scripts, or any active browser tab in Essbase to quickly return to that location.

Simply bookmark the browser tab you want to save, and use the bookmark to return there. This allows you to return to the specific application, database, script, or any active browser tab without going through all of the steps to navigate there in the Essbase web interface.

Open Artifacts in a New Browser Tab or in an Editor

You can open certain Essbase files in a new browser tab or in an editor. This works for files that have dedicated editors in the Essbase web interface.

To open the following file types in a new browser tab or in an editor

- `.csc` – calculation scripts
- `.rul` – rule files
- `.rep` – report scripts
- `.msh` – MaxL scripts
- `.mdx` – MDX scripts
- `.dtr`– drill-through reports

1. On the Home page, select the application.

Copyright © 2019, 2026, Oracle and/or its affiliates.

2. Select the cube.
3. Click **Files**.
4. Go to the Actions menu for a file ***.
5. Select **Open in New Browser** or **Open in Editor**.

> **Note**
>
> Some file types, such as .txt and .xml cannot be opened in this way. You can, however, download .txt and .xml files using the Actions menu to the right of the file.

To open a file in an editor, from within the application,

1. From the Home page, click **Files**.
2. Navigate to the cube directory, **Applications** > *application name* > *cube name*.
3. Click the name of the file you want to open in an editor.

> **Note**
>
> Some file types, such as .txt and .xml cannot be opened in this way. You can, however, download .txt and .xml files using the Actions menu to the right of the file.

Essbase, REST, and Smart View Client URLs

Get the Essbase web interface URL for the Oracle Essbase instance you are using from your Service Administrator. You access other Essbase components, such as the Smart View client and the REST API, using slightly different URLs.

The basic format of the Essbase web interface URL is:

```
https://Host:port/essbase/jet
```

The default secured port is 9001, unless it was changed during deployment.

For example:

```
https://myserver.example.com:9001/essbase/jet
```

The Smart View client and the REST API have similar URLs, with different endings.

Sample Smart View client URL:

```
https://myserver.example.com:9001/essbase/smartview
```

You can access Smart View if you have valid credentials. You can also configure the Smart View URL. See Connect to Essbase.

If you have multiple Essbase instances to connect to from Smart View, see Centralized Smart View URL, and Read-Only Clusters.

Copyright © 2019, 2026, Oracle and/or its affiliates.

A Provider Services URL has `/japi` appended to the end. You can use it to register multiple Essbase instances for centralized URL access. Example:

```
https://myserver.example.com:9001/essbase/japi
```

A discovery URL has `/agent` appended to the end. You can use it to log in to the MaxL Client. Example:

```
https://myserver.example.com:9001/essbase/agent
```

The following is an example of a REST API URL:

```
https://myserver.example.com:9001/essbase/rest/v1
```

Set Up Your Client

From the Console of the Essbase web interface, you can download desktop tools to use for administration, import, and export. Set up your local client computer using these tools. Many of your interactions with Essbase originate from your local machine.

You must install the version of client that matches the Essbase Server version.

> **Note**
>
> Be sure you're using the latest versions of tools provided in the Console, as older, previously downloaded versions may not work correctly.

- **Command Line Tools**
  Manage, migrate, backup, and recover Essbase applications.
  - **11g Excel Export Utility**—Exports Essbase 11g applications to application workbooks. You can use the application workbooks to re-create the applications on the current Essbase version.

    Download `dbxtool.zip`, and see Export Essbase 11g On-Premises Cubes and About Application Workbooks for details.
  - **11g LCM Export Utility**—Exports artifacts from Essbase 11g On-Premise as a `.zip` file, which you can import in to Essbase 12c or higher. This Life Cycle Management (LCM) utility can also be used to export from, and import to, 11g releases of Essbase. This utility packages into a zip everything you need to support migration to the current version. Download `EssbaseLCMUtility.zip`, and see the enclosed README for usage details.

    See also Migrate an Essbase 11g On-Premises Application.
  - **Command Line Interface (CLI)**—A scripting interface that uses REST APIs to perform most common Essbase administrative actions. The CLI includes an LcmImport command you can use for migrating 11g LCM Export Utility `.zip` files exported from Essbase 11g On-Premise. The LcmExport and LCMImport commands also facilitate migrating applications between instances, on versions 12c or higher.

    Download `cli.zip`, and see Download and Use the Command-Line Interface .

Copyright © 2019, 2026, Oracle and/or its affiliates.

  - **Migration Utility**—Utility to manage migration of an entire Essbase instance, for Essbase 12c or higher. In addition to migrating application artifacts, this utility also helps you migrate user role assignments and users/groups from supported identity providers. Download `migrationTools.zip`, and see the enclosed README for usage details.

    See also Migrate Using Migration Utility.

- **Smart View**
  - **Smart View for Essbase**—Provides a Microsoft Office interface for data analysis. It is the out-of-box query interface for Essbase.
  - **Cube Designer Extension**—Deploys Essbase cubes from formatted application workbooks. Cube Designer is an add-in to Smart View that enables desktop design of Essbase cubes. It can also be used to deploy cubes from tabular data in an Excel worksheet.

    See Set up Cube Designer.

- **Essbase Administration Services Lite**—Optionally manage applications using Essbase Administration Services (EAS) Lite. Although the Essbase web interface is the modern administration interface supporting all of the current platform features, a light version of Essbase Administration Services is a limited-support option for continued management of your applications, in case your company is not ready to adopt the new interface.

  See Use Essbase Administration Services Lite.

- **Essbase Maxl Clients**—Provides Linux and Windows clients to enable scripting of Essbase administrative tasks. MaxL is an administrative, language-based interface for managing Essbase cubes and artifacts.

  See Manage Essbase Using the MaxL Client.

- **Essbase Clients**—Provides libraries for Essbase C API.
- **Essbase Java API**—Enables development of Essbase client tools in Java, and provides libraries, samples and documentation for the Essbase Java API.

Ask Essbase

Overview of Ask Essbase

Ask Essbase is an AI-powered Oracle Essbase feature that allows you to retrieve information about Essbase technology and usage. Using natural language questions in a chat-style interface, you can receive answers sourced from official Essbase documentation combined with your choice of available GenAI models.

Key Components

- **Generative AI Tool:** Integrates the DBMS_CLOUD_AI package for smart search and response.
- **Storage:** Uses Autonomous AI Database for vector data and Oracle Cloud Infrastructure (OCI) Object Storage for documents.
- **Embedding Model:** Converts documentation into searchable vectors using the Cohere embedding model.
- **Large Language Model:** Processes queries and generates responses using the **Meta Llama** model (recommended), or an available model of your choice.

Copyright © 2019, 2026, Oracle and/or its affiliates.

- **Response Delivery:** Provides clear answers along with direct explanations and links to the relevant Oracle documentation for reference.

How It Works: When you enter a question into Ask Essbase, available as a chat function on the Academy page of the Essbase web interface, the system automatically searches relevant content, processes your query using AI, and displays a clear, concise answer with links to the original documentation.

> **Note**
>
> Oracle Essbase includes features powered by artificial intelligence / Gen AI. While Oracle strives to provide accurate results, Gen AI outputs may be incorrect, incomplete, or inappropriate. Information provided by Gen AI features must be verified by consumers of this product before taking actions or making decisions. Oracle is not liable for damages or losses arising from use of Gen AI features.

Configure Ask Essbase

You can use these steps to efficiently configure, activate, and manage Ask Essbase for seamless AI-driven querying within your Oracle Essbase environment.

Prerequisites

Before configuring Ask Essbase, complete the steps in Prerequisites for AI Features.

- **Marketplace Deployment:** Make sure **Catalog Storage Type** is set to **Object Storage Bucket**.

> **Note**
>
> Ask Essbase is enabled by default in Marketplace deployments. All users can submit queries, but only admin users can configure its settings.

> **Note**
>
> For optimal performance, use a Meta Llama chat model when creating an AI connection. Refer to Pretrained foundational models to view the available models.

Set Up Database Connection

1. Go to **Sources**.
2. Click **Create Connection**.
3. Select **Oracle Database** as the **Connection Type**.
4. Enter your Oracle AI Database 26ai instance credentials in the **Connection Details** .
5. Click **Save**.

Create AI Connection

1. Go to **Sources**.
2. Click **Create Connection**.

Copyright © 2019, 2026, Oracle and/or its affiliates.

3. Select **GenAI** as the **Connection Type**.
4. Fill in the parameters to complete the **Vector and chat credential details** section:
   a. **User OCID**: Find this value in the **Configuration file preview**.
   b. **Tenancy OCID**: Find this value in the **Configuration file preview**.
   c. **OCI Compartment ID**: Go to **Identity** in the Oracle Cloud Infrastructure console, then select **Compartments**. You can find the compartment ID there.
   d. **Private Key**: Refer to the Required Keys and OCIDs for instructions to create your private key.
   e. **Fingerprint**: Find this value in the **Configuration file preview**.

> **Note**
>
> **To get your OCI credentials:**
>
> a. Sign in to the Oracle Cloud Infrastructure console.
> b. Click **Profile** icon, then select **User settings**.
> c. Go to **Tokens and keys**.
> d. Under the **Fingerprint** list, select your API key.
> e. Click the menu (three dots).
> f. Select **View configuration file** to view the **Configuration file preview**.

> **Note**
>
> **If you haven't set up your OCI configuration:**
>
> a. Sign in to the Oracle Cloud Infrastructure console.
> b. Click **Profile** icon, then select **User settings**.
> c. Go to **Tokens and keys**.
> d. Click **Add API key**.
> e. Choose **Paste a public key**.
> f. On your VM, open your public key file and copy the public key content.
> g. Return to the Oracle Cloud Infrastructure page and paste your public key content.
> h. Click **Add**.
> i. After you add your API key, click menu (three dots) and then select **View configuration file** to view the **Configuration file preview**.

For more details, refer to the Required Keys and OCIDs.

5. Use the following details to complete the **Chat profile details** section and **Vectorization section and MDX generator details** section:
   a. **Chat model:** Select any available chat model listed under Pretrained foundational models.

Copyright © 2019, 2026, Oracle and/or its affiliates.

b. **OCI region:** Select a region where the Generative AI service and the models you want to use are available. See Generative AI Models by Region.

c. **Generative AI service URL:** Use the inference API endpoint for the selected OCI region and model. For example, for the US Midwest (Chicago) region: `https://inference.generativeai.us-chicago-1.oci.oraclecloud.com/20231130/actions/embedText`

d. **Vector embedding model:** Select an available embedding model listed under Pretrained foundational models.

e. **Transfer timeout:** 3000.

f. **Description:** (Optional).

Vectorize Documentation

1. Go to the **Academy**.
2. Select **Ask Essbase**.
3. Click **Settings** icon, and select the **Active GenAI connection** you set up.
4. Click **Vectorize**.
   a. This starts a background job that creates a vector index, uploads files to OCI storage, and prepares the system for chatbot queries.
   b. You can track the job progress in the Jobs UI or directly in the chatbot interface.
   c. You only need to perform this process once.

> **Note**
>
> If you change the vector embedding model, reconfigure Ask Essbase with the AI connection that uses the new embedding model, and then vectorize the documentation again.

Start Using Ask Essbase

1. After vectorization completes, ask questions directly in Ask Essbase.
2. The AI provides complete answers along with links to supporting documentation.

Deactivate

1. Select **Settings**, then select the **Delete** icon on the form page to deactivate Ask Essbase.

> **Note**
>
> This action initiates a background job that drops the vector index and vector table, deletes the chat profile linked to the vector index, and removes all documentation pointer files from the OCI bucket.

AI Query

To enable and use AI Query in Oracle Essbase with Oracle AI Database 26ai, you must set up connections and vectorize the outline. For AI powered data analysis, you can then submit natural language queries (NLQ), which generate valid MDX queries you can save as reports.

Copyright © 2019, 2026, Oracle and/or its affiliates.

> **Note**
>
> Oracle Essbase includes features powered by artificial intelligence (AI) and Generative AI (Gen AI). While Oracle strives to provide accurate results, Gen AI outputs may be incorrect, incomplete, or inappropriate. Information provided by Gen AI features should be verified by users before taking action or making decisions based on it. Oracle is not liable for damages or losses arising from the use of Gen AI features.

Prerequisites

Before using AI Query, complete the steps in Prerequisites for AI Features.

> **Note**
>
> Only **Service Administrators** can create AI connections.

- Deploy Essbase in Marketplace or
  Autonomous AI Database Serverless deployment mode.
- Use an Oracle AI Database 26ai with the required permissions.

1. **Create Global Connections in Essbase**

   In the Essbase web interface, under **Sources**:

   a. Create an Oracle AI Database Connection:
   Set up a connection to Oracle AI Database 26ai. For detailed steps, see Create a Connection and Datasource to Access Oracle Database.

   b. Create a GenAI Connection:
   Select **GenAI** as the connection type and complete the set up. For detailed steps, see Create a GenAI Connection.

2. **Associate an AI Connection with Your Application**

   > **Note**
   >
   > Only users with Application Manager permission can associate or dissociate an AI connection with an application.

   a. Go to the **Applications** page.

   b. Select the application.

   c. Select the link next to **AI Status**.

   d. Select **Agree** to accept the terms and conditions.

   e. Select the AI connection that you want to associate.

   f. Select **Associate**.

3. **Vectorize Outline**

   a. Run the **Vectorize Outline** job to convert your database outline to vectors. For complete instructions, see Vectorize Outline.

Copyright © 2019, 2026, Oracle and/or its affiliates.

b. Once the job is submitted, a background process performs outline vectorization. After the job completes successfully, AI Query becomes available for the selected application and database. The **Vectorize Outline** job is a one-time process.

> **Note**
>
> You can monitor the status of the **Vectorize Outline** job from the **Jobs** page in the Essbase web interface.

4. **Interact With AI Query**
   Once the vectorization job is complete:
   a. Open your **Application** and select your **Database**.
   b. Select **Analyze Data**.
   c. Select **Reports**.
   d. Select **AI Query** to open the AI Query panel.

   Additional Options in the AI Query Panel

   When you launch the AI Query panel, you can select from system-generated (auto-populated) starter questions.

   AI Query features:

   a. **Repopulate this query:** Regenerates the answer based on the same question.
   b. **Show MDX query:** Displays the generated MDX query.
   c. **Copy MDX query:** Copies the generated MDX to your clipboard.
   d. **Explain MDX:** Provides a natural language explanation of the generated MDX query, making it easier to understand the query structure and purpose.
   e. **Analyze query in Smart View:** Opens the report in Oracle Smart View for further analysis in Excel.
   f. **Clear conversation:** Resets the session or conversation history.
5. **Natural Language to Report Generation (Smart View Reports)**
   a. In the AI Query panel, select the **Create Report**.
   b. Enter a name for the report in the **Name** field and select **OK**. The report will appear in the **Reports** section in the left pane.
   c. Select the generated report to start working on it directly in the spreadsheet.
   d. To edit or delete a report, select the action menu next to the report and select **Edit** or **Delete** as needed.
6. **Natural Language to MDX Query Generation**
   a. In the left pane, select **Create**, then select **Use Natural Language Query**.
   b. Enter your query in the **Natural Language Query** field and select **Generate MDX**.
   c. Select **Validate** to review the generated MDX.
   d. Once validated, select **Save**. The report will appear in the **Reports** section in the left pane.
   e. Select the generated report to start working on it directly in the spreadsheet.
   f. To edit or delete a report, select the action menu next to the report and select **Edit** or **Delete** as needed.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Dissociate an AI Connection

AI Query can be deactivated for an application by deleting its associated AI connection.

1. Go to the **Applications** page.
2. Select the application.
3. Select **Delete** next to the AI connection name.
4. Select **Dissociate** to confirm.

> **Note**
>
> This action starts a background job that deletes all vector tables and related AI data created during outline vectorization in your Oracle AI Database schema. If AI Query does not reflect the current vectorization status or an error occurs after you dissociate and re-associate an AI connection, clear your browser cache and try again.

Copyright © 2019, 2026, Oracle and/or its affiliates.

2
Manage Essbase Applications

An Essbase application is a structure for managing one or more databases (cubes) and related artifacts/files.

Topics:

- Create an Application and a Database using the Essbase web interface
- Set Application-Level Configuration Properties
- Federated Cubes: Integrate Essbase with Autonomous AI Database

Create an Application and a Database using the Essbase web interface

An Essbase application is a structure for managing one or more databases (cubes) and related artifacts/files.

There are several ways to create Essbase applications and cubes:

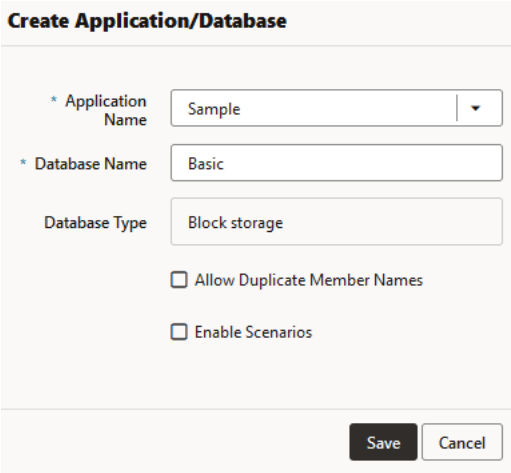
- Importing from an application workbook. See Design and Create Cubes Using Application Workbooks.
- Building from Cube Designer. See Work with Application Workbooks in Cube Designer.
- Using the Import LCM job or the lcmimport CLI command.
- Using the MaxL create application and create database statements.
- Using Essbase APIs.
- Creating from scratch in the Essbase web interface.

To create an application and database from scratch using the Essbase web interface,

1. Log in to Essbase as a power user or a service administrator. Refer to Understand Your Access Permissions in Essbase.
2. On the Home page, click **Create**.

3. Enter names for the application and the database. Refer to Naming Conventions for Applications and Databases for restrictions.
4. Select a database type, **Aggregate storage**, or **Block storage**. Refer to Comparison of Aggregate and Block Storage.
5. Select whether to allow duplicate member names and enable scenarios.

Copyright © 2019, 2026, Oracle and/or its affiliates.



6. Click **Save**.

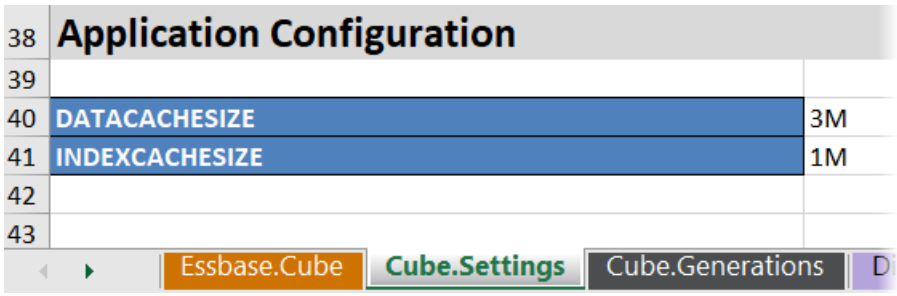
See Also

Create an Application and Database

Set Application-Level Configuration Properties

If you have the Service Administrator role, or the Power User role for applications that you created, you can customize Oracle Essbase using application-level configuration properties. Application-level configuration properties apply to all cubes in the application.

One way to specify configuration properties of an application is to do it prior to building the application and cube, using the application workbook. To see an example, go to Files in the Essbase web interface, and download the application workbook `Sample_Basic.xlsx`. It is located in the gallery, in the Demo Samples section (under Block Storage). In this application workbook, go to the Cube.Settings worksheet. Under Application Configuration, you can see that the DATACACHESIZE property is set to 3M, and the INDEXCACHESIZE property is set to 1M.



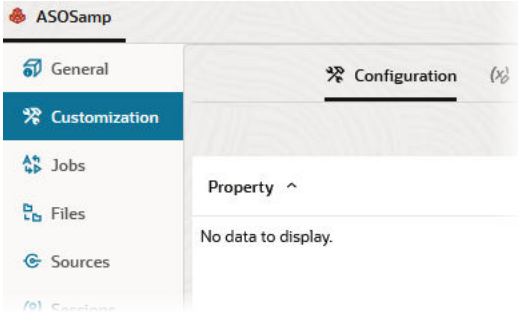
The following steps tell you how to configure an application that is already deployed, by adding properties and their corresponding values in the Essbase web interface.

- Redwood
- JET

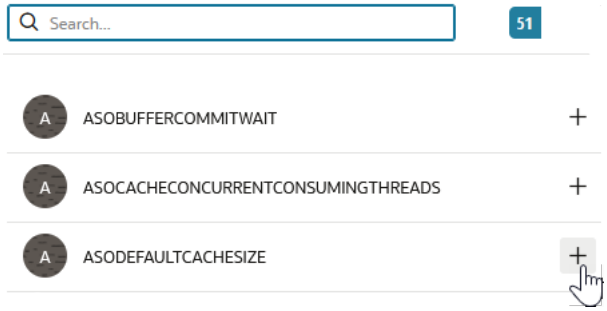
Copyright © 2019, 2026, Oracle and/or its affiliates.

Redwood

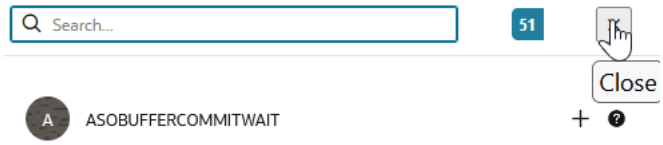
1. On the Applications page, select the application you want to configure.
2. Select the **Customization** tab, and then select the **Configuration** tab.



3. To add a property, click **Add**.
   Scroll through the list or search for a property.
4. Click + to add the property to the list.



5. Click × to close the search tool.



6. In the **Value** column, double click to enter a value.

7. When you're finished making changes, click **Apply and Restart**.
8. Wait for the confirmation message.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Configuration settings were stored successfully and will be applied when the application is restarted

JET

1. On the Applications page, select the application you want to configure.
2. From the **Actions** menu to the right of the application, click **Inspect**, then click **Configuration**.

3. To add a property, click . Scroll through the list or search for a property.
4. Click to add the property to the list.

5. Click to close the search tool.

6. In the **Value** column, double click to enter a value.

7. When you're finished making changes, click **Apply and Restart**.
8. Wait for the confirmation message.

For syntax and information about each of the application configuration properties you can use, see Config Settings List. You do not need to use the optional `[appname]` syntax when adding properties to the application configuration.

Oracle does not recommend that you modify `essbase.cfg` on the Essbase file system. This configuration is automatically set.

Copyright © 2019, 2026, Oracle and/or its affiliates.

3
Manage Essbase Cubes

The Essbase cube (database) is a data repository that contains a multidimensional data storage array. A multidimensional cube supports multiple views of data so that users can analyze the data and make meaningful business decisions.

Topics:

- Create an Essbase Cube
- Write and Edit Scripts
- Create Calculation Scripts
- Report on Data
- Link Cubes Using Partitions or @XREF/@XWRITE

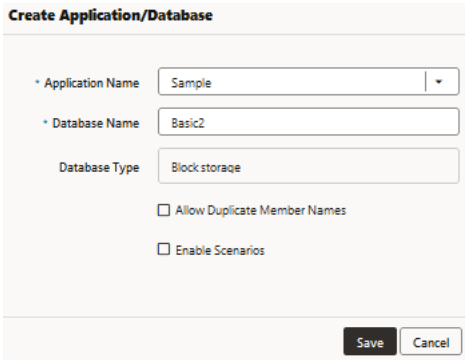
Create an Essbase Cube

You can create a cube at the same time you create the application, or you can create additional cubes later.

To create a new cube in the Essbase web interface,

1. Log in to Essbase as a power user or a service administrator.
2. On the Home page, click **Create**.

3. Select the name of an existing application.
4. Type a new cube name.
5. Select a cube type, aggregate storage or block storage.
6. Select whether to allow duplicate member names and enable scenarios.



7. Click **Save**.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Write and Edit Scripts

Write and edit Essbase scripts using enhanced script editing capabilities in the Redwood Interface.

Essbase now offers script editing capabilities in the calculation script, MDX script, and report script editors, as well as in the formula editor in the outline. You can use keyboard shortcuts to enable script editing functionality such as:

- Selecting and editing multiple lines at once
- Undo and redo
- Expand and collapse
- Find and replace
- Commenting the highlighted text

The syntax is always highlighted by default.

Essbase script editors also support code completion. Press the Tab key to select from the suggested list.

The following table lists the commonly used shortcuts for both, Mac and Windows systems:

Action	For Mac	For Windows
Go to line	Ctrl + G	Ctrl + G
Delete line	Shift + Command + K	Ctrl + Shift + K
Select all occurrences of current word	Command + F2	Ctrl + F2
Insert cursor	Opt + Click	Alt + Click
Insert cursor above	Opt + Command + ↑	Ctrl + Alt + ↑
Insert cursor below	Opt + Command + ↓	Ctrl + Alt + ↓
Undo	Command + Z	Ctrl + Z
Redo	Command + Shift + Z	Ctrl + Shift + Z
Fold region (Collapse)	Opt + Command + [	Ctrl + Shift + [
Unfold region (Expand)	Opt + Command +]	Ctrl + Shift +]
Find	Command + F	Ctrl + F
Replace	Opt + Command + F	Ctrl + H
Toggle line comment	Command + /	Ctrl + /

Create Calculation Scripts

Essbase calculation scripts specify how block storage cubes are calculated and, therefore, override outline-defined cube calculations. For example, you can calculate cube subsets or copy data values between members.

You create calculation scripts using a script editor in the Essbase web interface.

Calculation scripts do not apply to aggregate storage applications.

- Redwood

Copyright © 2019, 2026, Oracle and/or its affiliates.

- Classic

Redwood

1. On the Home page, open the application and then open the database (cube).
2. Click **Scripts**, and then click **Calculation Scripts**.
3. Click **Create** to create a new calculation script.
4. Enter a name for the new script.
5. If member names are required in your calculation script, drill into the **Member Tree** to find the members you want to add.
6. Double-click dimension or member names to insert them into the script.
7. If function names are required in your calculation script, use the **Function Name** menu to find calculation functions and add them to the script.
   See the **Function description** under the menu to read descriptions of each function.
8. Click **Validate** before saving your script.
   Validating a script verifies the script syntax. For example, incorrectly spelled function names and omitted end-of-line semicolons are identified. Validation also verifies dimension names and member names.
9. Correct any validation errors.
10. Click **Save**.

Classic

1. On the Application page, expand the application.
2. From the Actions menu, to the right of the cube name, launch the inspector.
3. Select the **Scripts** tab, and then select the **Calculation Scripts** tab.
4. Click Add to create a new calculation script.
5. Enter a name in the **Script Name** field.
6. If member names are required in your calculation script, drill into the **Member Tree** to find the members you want to add.
   Right-click dimension or member names to insert them into the script.
7. If function names are required in your calculation script, use the **Function Name** menu to find calculation functions and add them to the script.
   See the **Function description** under the menu to read descriptions of each function.
8. Click **Validate** before saving your script.
   Validating a script verifies the script syntax. For example, incorrectly spelled function names and omitted end-of-line semicolons are identified. Validation also verifies dimension names and member names.
9. Correct any validation errors.
10. Click **Save**.

To learn about calculation script logic, see Developing Calculation Scripts for Block Storage Databases.

Copyright © 2019, 2026, Oracle and/or its affiliates.

To learn about calculation functions and commands, see Calculation Functions and Calculation Commands.

Report on Data

Report Writer is a text-based script language that you can use to report on data in Essbase cubes. You can combine selection, layout, and formatting commands to build a variety of reports.

With Report Writer, you can generate reports whose length or specialized format exceed the capabilities of some grid clients.

Report scripts generate formatted data reports from a cube. Using report script editor in the Essbase web interface, you can create report scripts that specify exactly how you want to report on data. A report script consists of a series of Essbase report commands that define the layout, member selection, and format of a report.

To execute a report script, you must have read or higher access level to all data members specified in the report. Essbase filters any member from the output for which you have insufficient permissions.

- Create Report Scripts
- Execute Report Scripts

Create Report Scripts

Report scripts generate formatted data reports on an Essbase cube. A report script consists of a series of commands. You can use the Essbase web interface to write scripts to generate large-scale reports that consist of many pages of multidimensional data.

Reports of this scale often exceed the capabilities of even the most robust spreadsheet. In report script editor, you use report commands to define formatted reports, export data subsets from a database, and produce free-form reports. You can then execute the script to generate a report.

To learn how to write report scripts, refer to Report Writer for syntax and examples.

- Redwood
- Classic

Redwood

1. On the Home page, open the application and then open the database (cube).
2. Click **Scripts**, and then click **Report Scripts**.
3. Click **Create**.
4. In the **New Report Script** field, enter a name for the report script.
5. In the editing pane, enter the report script contents, or copy and paste it from a text editor.

Classic

1. On the **Applications** page, expand the application.

Copyright © 2019, 2026, Oracle and/or its affiliates.

2. From the **Actions** menu, to the right of the cube name, launch the inspector.
3. Select the **Scripts** tab, and then select the **Report Scripts** tab.
4. Click **Add** to create a new report script.
5. In the **Script Name** field, enter a name for the report script.
6. In the editing pane, enter the report script contents, or copy and paste it from a text editor.

Execute Report Scripts

After creating and saving report scripts, you can execute them in the script editor of the Essbase web interface to report on data loaded in your cube.

- Redwood
- Classic

Redwood

1. Create your report script, upload an existing report script, or select one from the gallery.
2. On the Home page, open the application and then open the database (cube).
3. Click **Scripts**, and click **Report Scripts**.
4. Select the report script you want to run.
5. Click **Execute**.
6. Download or print the report output.

Classic

1. Create your report script, upload an existing report script, or select one from the gallery.
2. On the Applications page, expand an application, and select a cube.
3. From the **Actions** menu, to the right of the cube name, launch the inspector.
4. Select **Scripts**, and select **Report Scripts**.
5. Select the report script you want to run.
6. Click **Execute**.
7. Download or print the report output.

Copyright © 2019, 2026, Oracle and/or its affiliates.

4

Understand Your Access Permissions in Essbase

How you work with Essbase depends on your user role and application-level permissions.

In Essbase, there are three user roles:

- User
- Power User
- Service Administrator

The majority of Essbase users have **User** role. **Power User** and **Service Administrator** roles are reserved for those who require permission to author and maintain applications. Users with **User** role are granted application-level permissions that distinguish their access to data and permissions in each application.

Access to Essbase is restricted by user and group security. User and group accounts are managed in an identity domain when Essbase is deployed on OCI via Marketplace. When Essbase is deployed independently, user and group accounts can be managed either in EPM Shared Services, or WebLogic Embedded LDAP authentication (with or without federation to an external identity provider).

See Manage Essbase User Roles and Application Permissions for independent deployments, or Manage Users and Roles for deployments on OCI via Marketplace.

Security Provider	Add, remove, and manage users and groups	Provision and deprovision roles
EPM Shared Services security mode	In the Shared Services Console	In the Shared Services Console
External security configured in WebLogic	In the external provider	In the Essbase web interface or REST API
WebLogic Embedded LDAP	In the Essbase web interface or REST API	In the Essbase web interface or REST API

> **Note**
>
> WebLogic Embedded LDAP is not recommended for production environments.

EPM Shared Services security mode

The following Essbase web interface items are **disabled** in EPM Shared Services security mode:

- The Security page (there is no Security option in the Essbase web interface)
  Essbase users and groups are stored directly in EPM Shared Services and are not added or managed in the Essbase web interface.
- The Permissions tab

Copyright © 2019, 2026, Oracle and/or its affiliates.

  - In the Redwood Interface, the Permissions tab is in the application, in the left hand column.
  - In the Classic Web Interface, the Permissions tab is in the application inspector.
- The Reset Password option on the Admin menu

External security configured in WebLogic

If you are using an external security provider configured in WebLogic, Essbase users and groups are stored directly in the external provider and are not added or managed in the Essbase web interface. However, you provision and deprovision roles in the Essbase web interface or through the REST API.

The following Essbase web interface items are **enabled** when using external security configured in WebLogic:

- The Security page (there is a Security option in the Essbase web interface)
- The Roles tab (users must have been added in order to be assigned roles)
  - In the Redwood Interface, the Roles tab is in the application, under Permissions.
  - In the Classic Web Interface, Roles are located on the Security page (the Users and Groups tab is disabled).
- The Permissions tab
  - In the Redwood Interface, the Permissions tab is in the application, in the left hand column.
  - In the Classic Web Interface the Permissions tab is in the application inspector.
- The Reset Password option on the Admin menu

> **Note**
>
> If you need to clean up inactive users/groups from Essbase after they have been removed or renamed on the external provider, use the MaxL Drop User and Drop Group statements.

WebLogic Embedded LDAP (an internal LDAP that is part of WebLogic, and is not recommended for production use):

Use the Security page (the Security option on the Applications page) in the Essbase web interface or use the REST API to manage users and groups and to provision and deprovision roles.

User Role

If your user role in Essbase is **User** with no application permissions, you can use the Files catalog (specifically, the `shared`, `users`, and `gallery` folders), download desktop tools from the Console, and explore the Academy to learn more about Essbase.

You must be granted additional access to applications by **Power Users** or **Service Administrators**. Applications are structures that contain one or more cubes, also known as databases. You can see only applications and cubes for which you have been granted application permissions.

You can have a unique application permission for each application on the server. Application permissions, from least privileged to highest, are:

Copyright © 2019, 2026, Oracle and/or its affiliates.

- None (no application permission has been granted)
- Database Access
- Database Update
- Database Manager
- Application Manager

Database Access Permission

If your user role in Essbase is **User** and you have Database Access permission for a particular application, you can view data and metadata in the cubes within the application.

Your ability to view data and metadata may be limited in areas that are restricted by filters. You may be able to update values in some or all areas of the cube, if someone has granted you write access using a filter. You can use drill through reports, if any exist, to access sources of data outside the cube, as long as a filter does not restrict your access to the cells within the drillable region.

With Database Access permission, you can also view the cube outline, and download files and artifacts from the application and cube directories. Job types you can run include building aggregations (if the cube is an aggregate storage cube), and running MDX scripts. Using the Console, you can view database size and monitor your own sessions.

If you are a scenario participant, you can view base data as well as scenario changes, and if you are a scenario approver, you can approve or reject the scenario.

Database Update Permission

If your user role in Essbase is **User** and you have Database Update permission for a particular application, you can make updates to the cubes within the application.

With Database Update permission for a particular application, you can do everything that a user with Database Access permission can do. Jobs you can run include loading, updating, and clearing data in the cube. You can export the cube data to tabular format. You can run any calculation scripts that you have been granted permission to execute. You can create, manage, and delete your own scenarios in block storage cubes that are enabled for scenario management.

Database Manager Permission

If your user role in Essbase is **User** and you have Database Manager permission for a particular application, you can manage the cubes within the application.

With Database Manager permission for an application, you can do everything that a user with Database Update permission can do. Additionally, you can upload files to the cube directory, edit the cube outline, export the cube to an application workbook, and start/stop the cube using the web interface. Job types you can run include building dimensions, exporting data, and exporting the cube to a workbook.

Additionally, as a Database Manager, you can control these operations:

- Enable scenarios or change the number of scenarios allowed
- Manage dimensions, including generation and level names
- Access and manage files related to the database

Copyright © 2019, 2026, Oracle and/or its affiliates.

- Create and edit calculation scripts, drill through reports, MaxL scripts, MDX scripts, report scripts, and rules files for dimension building and data loading
- Assign users permissions to execute calculation scripts
- Create and assign filters to grant or restrict data access for specific users and groups. You can assign filters, for your cube, to any users or groups who are already provisioned to use the application (an Application Manager or higher must provision users).
- Manage cube-level substitution variables
- View locked cube objects and data blocks
- View and change database settings
- View database statistics
- View and export audit records from the web interface

In the Redwood Interface, you can select the database and then manage these tasks from the left hand panel.

In the Classic Web Interface, manage these tasks from the database inspector. To open the database inspector from the web interface, start with the Applications page, and expand the application. From the **Actions** menu to the right of the name of the cube you want to manage, click **Inspect** to launch the inspector.

Application Manager Permission

If your user role in Essbase is **User** and you have Application Manager permission for a particular application, you can manage the application and cubes.

With Application Manager permission for a particular application, you can do everything that a user with Database Manager permission can do, for all cubes in the application. Additionally, you can make copies of any cubes within the application. You can copy or delete the application if you are the owner (the power user who created it), and you can delete any of the cubes in the application, if you are the cube owner (the power user who created it). You can start/stop the application using the Essbase web interface, and you can view and terminate user sessions in the Console. Job types you can run include running MaxL scripts, and using Export LCM to back up cube artifacts to a zip file.

You can manage cubes in your application the same way that a Database Manager can, and additionally, you can purge audit records for cubes.

Additionally, as an Application Manager, you can control these operations:

- Access and manage files related to the application
- Manage application-level connections and Datasources for access to external sources of data
- Change application configuration settings
- Provision and manage user and group permissions for the application and its cubes
- Add and remove application-level substitution variables
- Change general application settings
- View application statistics
- Download application logs

Copyright © 2019, 2026, Oracle and/or its affiliates.

In the Redwood Interface, select an application and then manage tasks from the left hand panel. Some tasks are grouped together. For example, Statistics and Logs are under "General."

In the Classic Web Interface, use the application inspector. To open the application inspector from the Classic Web Interface, start with the Applications page. From the **Actions** menu to the right of the name of the application that you manage, click **Inspect** to launch the inspector.

Power User Role

Power User is a special user role that enables you to create applications on an Essbase service.

If you are a power user, you are automatically granted Application Manager privilege for applications you created. Your options for creating applications and cubes include creating them from scratch in the Applications page of the web interface, importing from an application workbook, building from Cube Designer, and using the **LCM Import** job (or the `lcmimport` CLI command).

You can delete and copy applications that you created.

As a power user, you can be assigned permission to work on applications that you did not create. If your assigned permission is lower than Application Manager, then your actions are restricted to the actions permitted for the application permission you were assigned. For example, if you are assigned Database Manager permission to an application created by another power user, then your access is restricted to what a User with Database Manager permission can do.

Service Administrator Role

A **Service Administrator** has unlimited access to Essbase.

If you are a service administrator, you can do everything that power users and Application Managers can do, for all applications and cubes. Additionally, you can manage users and groups, using the Security page in the web interface. From the **Analyze** view for any cube, you can execute MDX reports impersonating other users (using **Execute As**) to test their access.

From the Console, you can manage connections and Datasources at the server level, configure e-mail settings for scenario management, and manage the antivirus scanner, all user sessions, and system configuration. You can also view statistics for all databases, add and remove global substitution variables, access Performance Analyzer to monitor service usage and performance, and view/change any service-level settings.

Unlike Power User, the Service Administrator role cannot be restricted. Service administrators always have full access to all applications and cubes on the Essbase server.

Filter Access to Data

Filters control security access to data values in the Essbase cube. Filters are the most granular form of security available. Filters designate a set of restrictions on particular cube cells or on a range of cells. Administrators or managers assign the filter to regulate access for users or groups.

Your own security role determines if you can create, assign, edit, copy, rename, or delete filters:

Copyright © 2019, 2026, Oracle and/or its affiliates.

- If you have the Application Manager role, then you can manage any filter for any user or group. Filters do not affect you.
- If you have the Database Update role, then you can manage filters for the applications that you created.
- If you have the Database Manager role, then you can manage filters within your applications or cubes.
- If you have the Database Access role (default), then you have read access to data values in all cells, unless your access is further restricted by filters.

Create a Filter using the Essbase web interface

In the database customization section of the Essbase web interface, you can create a security filter restricting cube access to users or groups.

You can create multiple filters for a cube. If you edit a filter, modifications made to its definition are inherited by all users of that filter.

1. Navigate to the filter editor.

   In the Redwood Interface:

   a. On the Home page, open the application and then open the database (cube).

   b. Click **Filters**.

   In the Classic Web Interface:

   a. On the Applications page, expand the application.

   b. From the Actions menu, to the right of the cube name, launch the inspector.

   c. Select the **Filters** tab.
2. Click **Add**.
3. Enter a filter name in the **Filter Name** text box.
4. In the Filter Editor, click **Add**.
5. Under **Access**, click and use the drop-down menu to select an access level.
   - None: No data can be retrieved or updated
   - Read: Data can be retrieved but not updated
   - Write: Data can be retrieved and updated
   - MetaRead: Metadata (dimension and member names) can be retrieved and updated

     The MetaRead access level overrides all other access levels. Additional data filters are enforced within existing MetaRead filters. Filtering on member combinations (using AND relationships) does not apply to MetaRead. MetaRead filters each member separately (using an OR relationship).
6. Select the row under **Member Specification** and enter member names, then click **Save**.

   You can filter members separately, or you can filter member combinations. Specify dimension or member names, alias names, member combinations, member sets that are defined by functions, or substitution variable names, which are preceded by an ampersand (&). Separate multiple entries with commas.
7. Create additional rows for the filter as needed.

   If filter rows overlap or conflict, more detailed cube area specifications apply over less detailed, and more permissive access rights apply over less permissive. For example, if

Copyright © 2019, 2026, Oracle and/or its affiliates.

you give a user Read access to Actual and Write access to Jan, then the user would have Write access to Jan Actual.

8. Click **Validate** to ensure that the filter is valid.
9. Click **Save**.

To edit a filter in the Redwood Interface, select the cube and click the Filters option in the left hand panel. Then, edit the filter by clicking the filter name and making your changes in the filter editor. To edit an existing row, double click in that row.

To edit a filter in the Classic Web Interface, go to the Filters tab in the inspector, and edit the filter by clicking the filter name and making your changes in the filter editor.

You can copy, rename, or delete a filter by clicking the Actions menu to the right of the filter name and choosing an option.

After creating filters, assign them to users or groups.

See Controlling Access to Database Cells Using Security Filters.

Create Efficient Dynamic Filters

You can create dynamic filters based on external source data to reduce the number of filter definitions needed.

Instead of managing a set of hard-coded data-access filters for many users, you can filter access to cube cells from external source data, based on member and user names.

You do this using dynamic filter definition syntax, including the method `@datasourceLookup` and the variables `$LoginUser` and `$LoginGroup`. Your external source data is a csv file or a relational table. For relational source data, you can load the .csv to a relational table.

- Dynamic Filter Syntax
- Workflow to Create Dynamic Filters
- Example of a Dynamic Filter

Dynamic Filter Syntax

Use dynamic filter syntax to create flexible filters you can assign to multiple users and groups.

Filter rows can contain the following elements as part of their definition, in addition to member expressions.

$loginuser

This variable stores the value of the current logged in user at runtime. It can be used in conjunction with the `@datasourcelookup` method.

$logingroup

This variable stores the value of all the groups that current logged-in user belongs to. It includes both direct and indirect groups. When used in conjunction with the `@datasourcelookup` method, each group is individually looked up against the Datasource.

@datasourcelookup

This method fetches records from a Datasource.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Syntax

```
@datasourcelookup (dataSourceName, columnName, columnValue, returnColumnName)
```

Parameter	Description
dataSourceName	The name of the external Datasource defined in Essbase. For an application-level Datasource, prefix the name with the application name and a period.
columnName	The name of the Datasource column to search for a given *columnValue*.
columnValue	The value to search for in *columnName*.
returnColumnName	The name of the Datasource column from which to return a list of values.

Description

A @datasourcelookup call is equivalent to the following SQL query:

```
select returnColumnName from dataSourceName where columnName=columnValue
```

@datasourcelookup looks up the given Datasource and searches for records where *columnName* contains *columnValue*. If you specify *columnValue* as `$loginuser`, this method will search for records where *columnName* contains the name of the currently logged in user.

Essbase forms the filter definition row by combining the list elements as a comma-separated string. If any record contains special characters, spaces, or only numbers, they are enclosed in quotation marks.

Examples

Enclose the parameters within quotation marks.

The following call looks up a global Datasource, and returns a list of store names where Mary is the store manager.

```
@datasourceLookup("StoreManagersDS","STOREMANAGER","Mary","STORE")
```

The following call looks up an application-level Datasource, and returns a list of store names where the currently logged in user is the store manager.

```
@datasourceLookup("Sample.StoreManagersDS","STOREMANAGER","$loginuser","STORE"
)
```

The following call looks up an application-level Datasource, and returns a list of store names where the store department matches any of the groups to which the logged in user belongs.

```
@datasourceLookup("Sample.StoreManagersDS","STORE_DEPARTMENT","$logingroup","S
TORE")
```

If the logged in user belongs to 3 groups, then the above `@datasourcelookup` method returns all the matching column values for each group.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Workflow to Create Dynamic Filters

Use the following general workflow to create dynamic filters.

This dynamic filters workflow assumes you already have a cube, and have provisioned users and groups.

1. Identify a source of data, whether it is a file or a relational source.
2. Define the connection and the Datasource in Essbase, either globally or at the application level.
3. Create filters at the cube level,
   - In the Redwood Interface, navigate to the cube and choose **Filters**.
   - In the Classic Web Interface, use the **Filters** section of the database inspector.
4. Define filter rows for each filter, using the dynamic filter syntax to employ the `$loginuser` variable, the `$logingroup` variable, and the `@datasourcelookup` method as needed.
5. Assign the filters to users or groups.
6. If you assigned the filter to a group, assign the group to the application to be filtered,
   - In the Redwood Interface, navigate to the application and choose **Permissions**.
   - In the Classic Web Interface, use the **Permissions** section of the application inspector.

Example of a Dynamic Filter

The following dynamic filter works with the cube called Efficient.UserFilters, available in the gallery as a sample template.

DSLookupFilter

Access	Member Specification
MetaRead	@datasourceLookup("EFFICIENT.UserDetails","USERNAME",$loginUser, "COUNTRY")
MetaRead	@datasourceLookup("EFFICIENT.UserDetails","USERNAME",$loginUser, "BUSINESSUNIT")
MetaRead	@datasourceLookup("EFFICIENT.UserDetails","USERNAME",$loginUser, "COSTCENTER")

To learn how to create and apply this dynamic filter, download the workbook template, `Efficient_Filters.xlsx`, from the Technical section of the gallery, and follow the README instructions in the workbook. The gallery is available in the **Files** section of the Essbase web interface.

Copyright © 2019, 2026, Oracle and/or its affiliates.

5
Manage Essbase Files and Artifacts

The Files catalog contains directories and files associated with using Essbase.

Topics:

- Explore the Essbase Files Catalog
- Explore the Gallery Templates
- Access Files and Artifacts
- Explore the Application Directories
- Work with Files and Artifacts
- Specify Files in a Catalog Path

Explore the Essbase Files Catalog

The Files catalog helps you organize the information and artifacts associated with using Essbase applications and cubes. You can access the Files catalog from Cube Designer, the Essbase web interface, the CLI, or MaxL.

The files catalog is grouped into the following folders:

- `applications`
- `gallery`
- `shared`
- `users`

What you can do in each folder depends on your permissions.

The `applications` folder is where Essbase saves applications and cubes.

The `gallery` folder contains application workbooks you can use to build sample cubes. These cubes help you learn about Essbase features, and model a variety of analytical problems across business domains.

The `shared` folder is a good location to store files and artifacts that you can use in more than one cube. Its contents are accessible to all users.

The `users` folder contains individual user directories. You can use your user folder for any files and artifacts that you use while working with Essbase.

In your own user folder, as well as in the shared folder, you can upload files and create subdirectories. No special permissions are required.

Explore the Gallery Templates

Gallery templates are application workbooks that you can use to build fully functional Essbase cubes. Think of these templates as starter kits you can use not only to build cubes, but to learn

Copyright © 2019, 2026, Oracle and/or its affiliates.

about Essbase features, and to model a variety of analytical problems across business domains.

The gallery templates include README worksheets, describing the purpose and usage of the workbook and cube.

Gallery templates are packaged in the form of an application workbook, and may also have additional supporting files. You use an application workbook to create an application and cube using either of these methods: the **Import** button in the Essbase web interface, or the **Build Cube** button on the Cube Designer ribbon in Excel. To access the gallery from the Essbase web interface, click **Files** and navigate to the gallery section. To access the gallery from Cube Designer, use the **Essbase** button on the Cube Designer ribbon.

The gallery templates are grouped into the following categories:

- Applications Templates
- Technical Templates
- System Performance Templates

Applications Templates

Gallery templates in the Applications folder demonstrate various business use cases for Essbase across several organizational domains.

The following cubes, located in the `gallery > Applications > Sales and Operations Planning` folder, connect together to perform their respective aspects of sales and operational planning tasks:

- Forecast Consensus—develop and maintain an agreed-upon forecast shared across departments
- Demand Consolidation—forecast customer demand
- Production Schedule—compute a weekly master production schedule for all products and locations
- Capacity Utilization—ensure that existing plant capacity can handle the production schedule

Compensation Analytics illustrates how Human Resources analysts can perform headcount and compensation analysis, analyze attrition, and allocate compensation increases.

Organization Restatements demonstrates how operational expenses can be restated, after organizational changes, for internal management reporting.

Opportunity Pipe demonstrates how to manage a sales pipeline.

Spend Planning shows how procurement analysts can manage operational spending using top-down and bottom-up forecasting methods.

Project Analytics demonstrates project planning risk analysis, accounting for factors such as workforce skills and costs, revenue, margin, inventory, and schedule.

RFM Analysis demonstrates how to identify the most profitable customers based on metrics.

Consolidation Eliminations is a financial analysis application demonstrating how to identify and eliminate balances between two companies.

Organization Restatements is a financial analysis application demonstrating how to restate expenses after an organizational change.

Copyright © 2019, 2026, Oracle and/or its affiliates.

In addition to these business applications, the Applications grouping of templates also includes:

- Demo Samples— simple examples of block storage and aggregate storage cubes commonly referenced in Essbase documentation.
- Utilities—cubes that may be utilized by other sample cubes. For example, the Currency Rates template takes currency symbols and returns the exchange rate to USD. The Currency Triangulation template uses a calculation script to triangulate currencies.

Technical Templates

The Technical templates demonstrate the use of Essbase features, including allocations, calc script debugging, runtime substitution variables, zig zag and asymmetric calculations, MDX Insert, solve order, real time updates, dynamic filters, sign flipping, and more.

- Calc: Allocation Tracing—perform allocations and debug calculation scripts
- Calc: Sample Basic RTSV—pass member names into a calculation script using runtime substitution variables
- Calc: Zigzag Calculation—learn how Essbase performs complex calculations across a time dimension
- Calc: CalcTuple Tuple—optimize asymmetric grid calculations across dimensions
- Drill Through: Drillthrough Basic—drill through to external sources to analyze data outside of the cube
- Filters: Efficient Filters—design and use variable data-access filters
- MDX: AllocationMDX Insert—allocate and insert missing values
- Partitions: Realtime CSV Updates— access real-time data
- Solve Order: UnitPrice SolveOrder—use and understand solve order in a hybrid mode cube
- Solve Order: Solve Order Performance—compare query performance using dynamic calculations versus using stored members and a calculation script
- Table Format— build Essbase cubes from tabular data
- UDA: Flip Sign— learn how to flip signs of data values during a data load to meet reporting requirements

System Performance Templates

System performance templates monitor system status for optimization purposes.

The Health and Performance Analyzer helps you monitor usage and performance statistics of your Essbase applications.

The Analyzer enables you to scan Essbase logs. After parsing the data, it compiles a csv Excel worksheet, optionally on the time interval that you set in **Settings**. Then, you can use the csv files to build charts and other displays.

Access Files and Artifacts

Your access to the Files catalog in Essbase depends on your user role and application-level permissions.

Copyright © 2019, 2026, Oracle and/or its affiliates.

You can access the Files catalog from Cube Designer or from the Essbase web interface.

If your user role in Essbase is **User** with no application permissions, you can access the `shared`, `users`, and `gallery` folders. The `applications` folder is empty.

The `gallery` folder is read-only for all users.

The `shared` folder is read-write for all users.

Within the `users` folder, users have read-write access to their own folders, and the service administrator has access to all.

If your role is **User** and you have Database Access or Database Update permission for a particular application, you can additionally view (and download from) the appropriate subdirectories beneath the `applications` folder. These subdirectories contain files and artifacts for applications and cubes you can access.

If your role is **User** and you have Database Manager permission for an application, you can additionally upload files and artifacts to the cube directory, as well as delete, copy, and rename them.

If your role is **User** and you have Application Manager permission for an application, you can do everything with files that the Database Manager can do, and your access is expanded to the application directory in addition to the cube directory.

If you are a Power User, you have the same access to files and artifacts that an Application Manager has, for applications you created. Your access to other applications is restricted according to the application permission you have been granted.

A service administrator has full access to all files and directories (except the `gallery` folder, which is read-only).

Explore the Application Directories

The applications directories in the Files catalog contain artifacts associated with using Essbase applications.

For each application that someone creates or imports, Essbase creates a new folder inside the `applications` folder in the Files catalog. The application folder contains the cube folder, and the cube folder contains cube artifacts.

Artifacts are files related to working with Essbase applications and cubes. Artifacts have various purposes, such as defining calculations or reports. Artifacts pertaining to a cube are stored, by default, in a folder associated with the cube -- also known as the database directory.

Common cube artifacts include:

- Text files of data or metadata that can be loaded to the cube (`.txt`, `.csv`)
- Rules files for loading data and building dimensions (`.rul`)
- Calculation scripts that define how to calculate data (`.csc`)
- Application workbooks and other Excel files (`.xlsx`)
- MDX scripts (`.mdx`)
- Stored metadata about the cube (`.xml`)

Copyright © 2019, 2026, Oracle and/or its affiliates.

> **Note**
>
> File extensions must be lower case. For example, *filename*`.txt`

Work with Files and Artifacts

Depending on your level of access defined in Essbase, you can perform file operations on folders and artifacts in the Files catalog. You can upload, download, copy, rename, move, open, and delete files using the Files section of the Essbase web interface.

This topic describes working in the Essbase web interface, but you can also work with files from Cube Designer or the Command Line Interface (CLI).

To upload an artifact,

1. On the Home page, click **Files**.
2. Navigate to a directory for which you have write access.
3. Optionally, click **Create Folder** to add a subdirectory (available for `shared` and user directories only).
4. Click **Upload**.
5. Drag and drop, or select a file from the file system.
6. Click **Close**.

> **Note**
>
> You can enable antivirus scanning in the Essbase web interface so that files are scanned for viruses before they are uploaded to the server.

To download an artifact,

1. Navigate to a directory for which you have read access.
2. From the **Actions** menu to the right of the file, select **Download**.

To copy an artifact,

1. Navigate to a directory for which you have read access.
2. From the **Actions** menu to the right of the file, select **Copy**.
3. Navigate to another folder for which you have write access.
4. Click **Paste**.

To rename an artifact,

1. Navigate to a directory for which you have write access.
2. From the **Actions** menu to the right of the file, select **Rename**.
3. Enter a new file name, omitting the extension.

To move an artifact,

1. Navigate to a directory for which you have write access.
2. From the **Actions** menu to the right of the file, select **Cut**.

Copyright © 2019, 2026, Oracle and/or its affiliates.

3. Navigate to a new directory for which you have write access.
4. Click **Paste**.

To open an artifact in a new browser tab or in an editor,

1. On the Home page, select the application.
2. Select the cube.
3. Click **Files**.
4. Go to the Actions menu for a file ⋯ .
5. Select **Open in Browser** or **Open in Editor**.

> **Note**
> Some file types, such as .txt and .xml cannot be opened in this way.

To open an artifact in an editor, from within the application,

1. From the Home page, click **Files**.
2. Navigate to the cube directory, **Applications** > *application name* > *cube name*.
3. Click the name of the file you want to open in an editor.

> **Note**
> Some file types, such as .txt and .xml cannot be opened in this way.

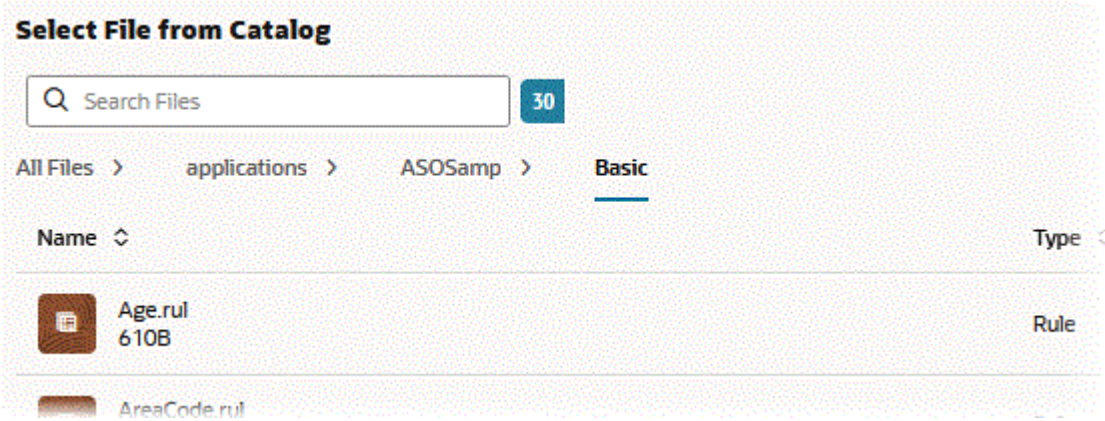
To delete an artifact,

1. Navigate to a directory for which you have write access.
2. From the **Actions** menu to the right of the file, select **Delete**.
3. Click OK to confirm that you want to delete.

Specify Files in a Catalog Path

If a data load or dimension build you are initiating for a cube requires a file or artifact that is located somewhere in Essbase other than the directory for the current cube, you can specify its catalog path.

Copyright © 2019, 2026, Oracle and/or its affiliates.



Other operations involving files or artifacts require them to be located either in the *cube directory*, or a directory specified by an administrator.

When you leave the catalog path unspecified, the cube directory is the assumed location, unless an administrator has specified an alternate path (by using FILEGOVPATH configuration).

The *cube directory* means the `<Application Directory>/app/appname/dbname` folder.

If you do not know where *<Application Directory>* is in your environment,

- Refer to Environment Locations in the Essbase Platform if you use an independent Essbase deployment.
- If you use an Essbase deployment on Oracle Cloud Infrastructure Marketplace, then *<Application Directory>* is `/u01/data/essbase/app`.

Whether you are using Jobs, MaxL, or Command Line Interface (CLI) for data loads or dimension builds, you can specify the catalog path to the needed files.

For example, the following MaxL import data statement performs a data load using a data file stored in the shared folder of the Essbase file catalog. The rule file is in the cube directory for Sample Basic.

```
import database 'Sample'.'Basic' data from server data_file 'catalog/shared/
Data_Basic' using server rules_file 'Data' on error write to "dataload.err";
```

In the following CLI dimbuild example, the rule file is specified in a user directory, and the data file in a shared directory.

```
esscs dimbuild -a Sample -db Basic -CRF /users/admin/Dim_Market.rul -CF /
shared/Market.txt -R ALL_DATA -F
```

Note

If your Essbase deployment is on Oracle Cloud Infrastructure Marketplace and uses object storage integration, then any jobs requiring access to files in `shared` or `user` directories in the Essbase catalog will search for them in (or export them to) the OCI object storage bucket associated with the Essbase stack on OCI. For more information, see Create Stack.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Copyright © 2019, 2026, Oracle and/or its affiliates.

6

Create and Manage Cube Outlines Using the Web Interface

An Essbase outline defines the structure of the cube through dimensions, members, attributes, and their properties. The outline structure, along with consolidation operators and formulas, determines how data is stored and calculated.

Dimensions and members represent data hierarchies. In an outline, each dimension consists of one or more members. The members, in turn, may have child members. This ancestral rollup is called a hierarchy. Unary operators (such as +, -, *, /), assigned to each member in a hierarchy define how a child member consolidates to its parent.

> **Note**
>
> To migrate an existing cube outline to Essbase 21c, use a migration tool. Do not copy the .otl file to migrate an outline.

Prerequisite: Import a cube from the gallery. The topics in this chapter assume you have built or imported Sample Basic. If you aren't a power user, ask one to create an application for you and provision you as Database Manager for the application.

- View and Edit Outline Properties for a Newly Created Cube
- Add Dimensions and Members to Outlines
- Name Generations and Levels
- Restructure Cubes
- Create Attribute Dimensions and Members
- About Duplicate Member Names
- Set Dimension and Member Properties
- Select the Member Properties to Display in the Outline
- Compare Outlines
- Copy and Paste Members Within and Between Outlines

View and Edit Outline Properties for a Newly Created Cube

Outline properties, in part, control the functionality available in an Essbase cube, but they also control member naming and member formatting for attribute dimensions, alias tables and text measures.

To view and edit an outline:

- Redwood

Copyright © 2019, 2026, Oracle and/or its affiliates.

- Classic

Redwood

1. Log into the Essbase web interface as a power user.
2. On the Home page, click **Create** to create a new application.
3. Give the application a unique name.
4. Give the database (cube) any name.
5. (Optional) Choose a database type, and select to allow duplicate member names or enable scenarios.
6. Click **Save**.
7. On the Home page, navigate to the new application, and select the Actions menu ⋯.
8. Choose **Databases** > *Database Name* > **Launch Outline**.
9. Click **Edit** .
10. Click **Outline Properties** .

Classic

1. Log into the Essbase web interface as a power user.
2. On the Applications page, click **Create** to create a new application.
3. Give the application a unique name.
4. Name the cube.
5. (Optional) Click **Advanced Options** to select a database type, allow duplicate member names, or enable scenarios.
6. Click **OK**.
7. On the Applications page, expand the new application.
8. From the Actions menu, to the right of the cube name, select **Outline**.
9. Click **Edit outline**.
10. Click **Outline Properties**.

Work with General and Attribute-related Outline Properties

Outline properties-General tab shows what outline features are enabled for your cube and how they are formatted. Some fields on this tab can be changed and others cannot be changed and are for your information.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Table 6-1 General Outline Properties

Field	Description	View or Edit
Allow Duplicate Member Names	Enabling a cube for duplicate member names is an option when a new application is created. If you migrate an on-premises Essbase application with a unique member outline to an Essbase instance, you cannot change the outline to allow duplicate members. To allow duplicate member names in your Essbase instance, convert the on-premises unique member outline to a duplicate member outline before migrating the application.	This field cannot be changed and is for your information.
Typed Measures Enabled	All Essbase applications are enabled for typed measures by default.	If typed measures is disabled and you want to enable it, select True. If typed measures is enabled, you cannot change the setting and this field is for your information.
Date Format	You can change the date format if you plan to use typed measures that are dates.	Use the dropdown list to select the date format that will be displayed when you query typed measures that are dates.
Auto configure dimension storage type	When "Auto configure dimension storage type" is enabled, the dimensions are set to dense or sparse automatically. When using this option, there is a limit of twenty four dimensions. This setting applies to block storage cubes only.	If auto configure is disabled and you want to enable it, select True. If auto configure is enabled and you want to disable it, select False.

Table 6-2 Boolean, Date and Numeric

Field	Description	View or Edit
True Member Name	Although your cube can contain more than one Boolean attribute dimension, all Boolean attribute dimensions will share the same value for True Member Name and False Member Name. By default, Essbase assigns member names of True and False. If you want to change these names, you must change them before you add the first Boolean attribute to your cube. Once the first Boolean attribute dimension is created, you cannot change these names.	This field can only be changed before you add the first Boolean attribute dimension to your cube.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Table 6-2 (Cont.) Boolean, Date and Numeric

Field	Description	View or Edit
False Member Name	Although your cube can contain more than one Boolean attribute dimension, all Boolean attribute dimensions will share the same value for True Member Name and False Member Name. By default, Essbase assigns member names of True and False. If you want to change these names, you must change them before you add the first Boolean attribute to your cube. Once the first Boolean attribute dimension is created, you cannot change these names.	This field can only be changed before you add the first Boolean attribute dimension to your cube.
Date Member Names	You can change the format of members of date attribute dimensions.	Select Month First or Day First formatting convention for Date Member Names.
Numeric Range	Members of numeric attribute dimensions can be defined in dimension build rules to represent date ranges. Here, you can define these ranges to be Top or Bottom of Ranges. All numeric attribute dimensions built using ranges will have the same numeric range setting.	Options are Tops of Ranges and Bottoms of Ranges.

Table 6-3 Attribute Settings – Prefix and Suffix Format

Field	Description	View or Edit
Value	A prefix or suffix may be required for your attribute member names to support member name uniqueness. Prefix or suffix values display when attribute dimension members are included in a query.	To enable prefix or suffix values for your cube, make a selection in the Value drop-down menu. The default value of None disables all prefix or suffix options.
Format	You can define unique names by attaching a prefix or suffix to member names in Boolean, date, and numeric attribute dimensions in the outline.	After selecting a prefix or suffix Value, such as Parent, you can select the format.
Separator	Select a separator (to place between the prefix or suffix and the original name).	Options are underscore (_), pipe (|), or caret (^).

Copyright © 2019, 2026, Oracle and/or its affiliates.

Table 6-4 Calculation Dimension Names

Field	Description	View or Edit
Name	Every Essbase cube containing attribute dimensions contains a dimension containing standard math functions that can be applied to attribute queries. You can edit the name of this dimension, and the name of each standard math function. You cannot change which math functions are automatically calculated.	Type a name for the attribute calculations dimension, if you want to change it.
Sum Member	This is a member of the attribute calculations dimension. The name to use when requesting sum data.	Type a name for the Sum member in the attribute calculations dimension, if you want to change it.
Count Member	This is a member of the attribute calculations dimension. The name to use when requesting count data.	Type a name for the Count member in the attribute calculations dimension, if you want to change it.
Minimum Member	This is a member of the attribute calculations dimension. The name to use when requesting minimum data.	Type a name for the Minimum member in the attribute calculations dimension, if you want to change it.
Maximum Member	This is a member of the attribute calculations dimension. The name to use when requesting maximum data.	Type a name for the Maximum member in the attribute calculations dimension, if you want to change it.
Average Member	This is a member of the attribute calculations dimension. The name to use when requesting average data.	Type a name for the Average member in the attribute calculations dimension, if you want to change it.

Understand and Create Alias Tables

Aliases are stored in one or more tables as part of an Essbase database outline. An alias table maps a specific, named set of alias names to member names.

To create an alias table using the Essbase web interface,

- Redwood
- Classic

Redwood

1. On the Home page, navigate to the application, and select the Actions menu ⋯.
2. Choose **Databases** > *Database Name* > **Launch Outline**.
3. If the outline is locked, and you are an administrator, click **Unlock**.
   Before you forcefully unlock a locked outline, make sure that no one else is working with it.

Copyright © 2019, 2026, Oracle and/or its affiliates.

4. Click **Edit** .
5. Click **Outline properties** .
6. Click the **Aliases** tab.
7. Enter the name of the alias table you want to create and click **Add**.
   You can have a maximum of 56 alias tables.
8. Click **Apply and Close**.

Classic

1. On the Applications page, expand the application.
2. Click the **Actions** menu to the right of the cube name and click **Outline**.
3. Click **Edit**.
4. Click **Outline Properties**.
5. Select the **Aliases** tab.
6. Enter the name of the alias table you want to create and click **Add**.
   You can have a maximum of 56 alias tables.
7. Click **Apply and Close**.

See Create Aliases and Setting Aliases.

You cannot delete or rename the default alias table.

Understand and Work with Dynamic Time Series Outline Properties

To dynamically calculate period-to-date values, you can enable dynamic-time-series members for an outline. You must also associate the dynamic time series member with a generation member.

You use the Dynamic Time Series tab in the Outline Properties dialog box to enable and disable dynamic time series members, to associate dynamic time series members with generations, and to specify aliases for dynamic time series members. Your outline must contain a time dimension in order for you to work with dynamic time series members.

The **Series** column lists the eight system-defined dynamic time series members. See Using Dynamic Time Series Members:

- H-T-D (history-to-date)
- Y-T-D (year-to-date)
- S-T-D (season-to-date)
- P-T-D (period-to-date)
- Q-T-D (quarter-to-date)
- M-T-D (month-to-date)
- W-T-D (week-to-date)
- D-T-D (day-to-date)

To enable dynamic time series members:

Copyright © 2019, 2026, Oracle and/or its affiliates.

1. Go to **Outline Properties**.
   In the Redwood Interface:
   a. On the Home page, navigate to the application, and select the Actions menu ⋯.
   b. Choose **Databases** > *Database Name* > **Launch Outline**.
   c. If the outline is locked, and you are an administrator, click **Unlock**.
      Before you forcefully unlock a locked outline, make sure that no one else is working with it.
   d. Click **Edit**.
   e. Click **Outline properties**.

   In the Classic Web Interface:
   a. On the Applications page, expand the application.
   b. Click the **Actions** menu to the right of the cube name and click **Outline**.
   c. Click **Edit**.
      To view outline properties, simply click **Outline Properties**. You don't need to click **Edit** first.
   d. Click **Outline Properties**.
2. Click **Dynamic Time Series**.
3. Select or clear items in the **Enabled** column to enable or disable the member associated with the option.
4. In the **Generation** column, select a generation number.
   You cannot associate dynamic time series members with level 0 members of the time dimension, and you should not assign a generation number to multiple members.
5. (Optional) In the **Default** column, in the member row, enter one or more aliases (one each from one or more alias tables).

Understand and Create Textual Measures

Text measures extend the analytical capabilities of Essbase beyond numerical data to text-based content.

For example, assume that a user is to provide an input indicating risk assessment. It might be preferable to select from a list of strings: low, medium, high. To accomplish this in Essbase, you would create a text list object in the outline properties, and use it to assign the appropriate strings to numeric values stored in the database.

For information about creating text measures in Essbase, see Working with Text Measures.

To experiment with implementing text measures from an application workbook, follow the instructions in Text Measures Workflow, under Text Measures Workflow using Application Workbooks.

See also: Performing Database Operations on Text and Date Measures.

Add Dimensions and Members to Outlines

The top level members of any hierarchy in an Essbase outline are called dimension names or dimensions. There are two types of dimensions: standard dimensions and attribute dimensions.

Copyright © 2019, 2026, Oracle and/or its affiliates.

You can add dimensions and members to a cube using any of the following methods:

- Add dimensions and members manually with the outline in edit mode.
- Import an Excel file containing dimension definitions (either tabular data or an application workbook).
- Build dimensions using a datasource and rule file.

In this chapter, we focus on manual outline updates.

Add Dimensions to Outlines Manually

In block storage or partial hybrid mode cubes (which have one or more stored dimensions), if you add, delete, or move members in dimensions and then save the outline, then the cube is restructured.

After restructuring is complete, recalculate the data. Aggregate storage and fully hybrid mode cubes do not need to be recalculated because they are dynamic (upper level data is not stored).

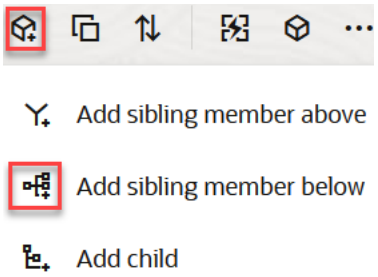
If you add a dimension that is virtual (dynamic calc or label only), then any data existing in the cube is stored with the first level-0 stored member in the new dimension. There must be at least one stored member in the hierarchy.

Dimension names must always be unique in the outline, even if the outline allows duplicate member names. To add a dimension to an outline:

- Redwood
- Classic

Redwood

1. On the Home page, navigate to the application, and select the Actions menu ⋯.
2. Choose **Databases** > *Database Name* > **Launch Outline**.
3. If the outline is locked, and you are an administrator, click **Unlock**.
   Before you forcefully unlock a locked outline, make sure that no one else is working with it.
4. Click **Edit**, and then select a dimension.
   If the outline doesn't contain any dimensions, the **Create your first dimension** text box is displayed. Type a dimension name, click **Add dimension**, and select the new dimension.
5. On the outline toolbar, from the Add member menu, select **Add sibling member below**.



Copyright © 2019, 2026, Oracle and/or its affiliates.

6. In the **Add Member(s)** dialog box, under **Member name**, enter a name.
   Use no more than 1024 characters when naming dimensions, members, or aliases.
7. Still in the **Add Member(s)** dialog box, select the member properties that you want for the new dimension.
8. Press **Add**.
9. Press **Close**.
10. Press **Validate** .
11. Press **Save** .
12. Select from restructure database options and save your changes. See Restructure Cubes.

Classic

1. On the Applications page, expand the application.
2. Click the **Actions** to the right of the cube name and then choose **Outline**.
3. Click **Unlock**. This is only needed if the outline is locked. Otherwise, proceed to step 4.
4. Click **Edit** and then select a dimension.
5. On the outline toolbar, under **Actions**, select **Add a sibling below the selected member**.
6. Enter a name for the new dimension and press Tab.
   Use no more than 1024 characters when naming dimensions, members, or aliases.
7. On the outline toolbar, under **Actions**, select **Display member properties panel on the right side** to open the properties pane, and select the properties that you want for the new dimension.
8. Click **Save**.

Add Members to Outlines Manually

You can manually add members to an outline using the outline editor in the Essbase web interface. Unless the cube is enabled for duplicate member names, each member has a unique name.

- Redwood
- Classic

Redwood

1. On the Home page, navigate to the application, and select the Actions menu ••• .
2. Choose **Databases** > *Database Name* > **Launch Outline**.
3. If the outline is locked, and you are an administrator, click **Unlock** .
   Before you forcefully unlock a locked outline, make sure that no one else is working with it.
4. Click **Edit** .

Copyright © 2019, 2026, Oracle and/or its affiliates.

5. To view and select lower-level members in a dimension, drill down in the dimension by expanding the dimension name and subsequent member names.
6. When you reach the member to which you want to add a child or sibling member, select it.
7. On the toolbar, from the Add member menu, select **Add sibling member above**, **Add sibling member below**, or **Add child**.
8. In the **Add Member(s)** dialog box, under **Member name**, enter a name for the new member.
   Use no more than 1024 characters when naming dimensions, members, or aliases.
9. Still in the **Add Member(s)** dialog box, select the properties that you want for the new member.
10. Press **Add** and then **Close**.
11. Press **Validate** .
12. Press **Save** .

Classic

1. On the Applications page, expand the application.
2. From the **Actions** menu, to the right of the cube name, select **Outline**.
3. Click **Edit**.
4. To view and select lower-level members in a dimension, drill down in the dimension by expanding the dimension name and subsequent member names.
5. When you reach the member to which you want to add a child or sibling member, select it.
6. From the outline toolbar, under **Actions**, select **Add a sibling above the selected member**, **Add a sibling below the selected member** or **Add a child to the selected member**.
7. Enter the name for the new member and press Tab.
   Use no more than 1024 characters when naming dimensions, members, or aliases.
8. On the outline toolbar, under **Actions**, select **Display member properties panel on the right side** to open the properties pane, and select the properties that you want for the new member.
9. Click **Save**.

Name Generations and Levels

You can create names for generations and levels in an Essbase outline, using a word or phrase that describes the generation or level. For example, you might create a generation name called Cities for all cities in the outline. You can define only one name for each generation or level.

Use generation and level names in calculation scripts wherever you need to specify either a list of member names or a list of generation or level numbers. For example, you can limit a calculation in a calculation script to the members of a specific generation.

1. Open **Dimensions**.

Copyright © 2019, 2026, Oracle and/or its affiliates.

In the Redwood Interface:

a. On the Home page, open an application and then open the database (cube).

b. Click **Dimensions**.

In the Classic Web Interface:

a. On the Applications page, expand an application.

b. From the **Actions** menu, to the right of the cube name, click **Inspect**.

c. In the inspector, select **Dimensions**.

2. On the **Dimensions** page, select the dimension in which you want to name generations or levels.
3. Double-click a generation or level name placeholder (for example, Gen1 or Lev1) to enable editing of that field.
4. Enter a generation or level name.
5. Click **Save.**

   For example, you can replace the placeholder texts, Gen1, Gen2, and Gen3, with descriptive generation names.

If you add generation names, then they will be included on a Cube.Generations worksheet when you export your cube to an application workbook.

Restructure Cubes

When you add a dimensions and members to an Essbase outline, and save the outline, it triggers a cube restructure. You specify how data values will be handled during the restructure. If you added or deleted a dimension, you are prompted to indicate data association changes.

1. In the outline editor, add a dimension to the outline. See Add Dimensions to Outlines Manually.
2. Add members as children of the new dimension. See Add Members to Outlines Manually.
3. Press **Validate** ✓.
4. Press **Save** .
5. In the **Restructure Database Options** dialog box, specify how data values will be handled during the restructure by selecting one of the following options:
   - **All Data** - All data values are preserved.
   - **Discard all Data** - All data values are cleared.

Copyright © 2019, 2026, Oracle and/or its affiliates.

- **Level 0 Data** - Only level 0 values are preserved. If all data required for calculation resides in level 0 members, you should select this option. If the option is selected, all upper level blocks are deleted before the cube is restructured. Therefore, the disk space required for restructuring is reduced, and calculation time is improved. When the cube is recalculated, the upper-level blocks are re-created.
- **Input Data** - Only the blocks that contain the data that is being loaded are preserved. However, all blocks (both upper-and lower-level) that contain loaded data are preserved.

6. Still in the **Restructure Database Options** dialog box, if prompted, select the member from the added dimension with which you want the existing data to be associated, or (if you have deleted a dimension) select the member from the deleted dimension for which you want to keep the data.
7. Click **Save**.

Create Attribute Dimensions and Members

Attributes describe characteristics of Essbase data, such as the size and color of products. You can use attributes to group and analyze members of dimensions based on their characteristics.

For example, you can analyze product profitability based on size or packaging, and you can make more effective conclusions by incorporating market attributes, such as the population size of each market region, into your analysis.

Workflow for manually building attribute dimensions:

- Redwood
- Classic

Redwood

When manually working with attributes in the Redwood Interface, use the outline editor, the Add member dialog box, and the member inspector in the outline editor.

1. Create dimensions with the dimension type of attribute. While in the **Add member** dialog box,
   a. Set the attribute dimension type (text, numeric, Boolean, or date).
   b. Associate a sparse standard dimension with each attribute dimension. Select the standard dimension you want to associate from the **Associated dimension** menu. This defines the base dimensions.
2. Add members to the attribute dimensions.
3. Save the outline.
4. Associate members of the base dimension with members of the attribute dimension; the base members must all be from the same level in the base dimension:
   a. Click **Edit** , select a base member to associate with the attribute dimension, and select **Inspect** .
   b. Click **Attributes**.

Copyright © 2019, 2026, Oracle and/or its affiliates.

   c. For each attribute dimension listed, under **Attribute value** begin typing and then select a specific attribute member.
   d. Repeat for each base member you want to associate to an attribute member.
5. Save the outline.

Classic

When manually working with attributes in the Classic Web Interface, use the outline editor, and the member inspector in the outline editor.

1. Create attribute dimensions.
2. Tag the dimensions as attribute dimensions and set the attribute dimension type (text, numeric, Boolean, or date).
   Use the member inspector, **General** tab to set the dimension as an attribute dimension, and to set the attribute dimension type.
3. Add members to attribute dimensions.
4. Associate a standard dimension with an attribute dimension, thereby defining the base dimension of the attribute dimension. Use the **General** tab in the member inspector to associate an attribute dimension to a base dimension.
5. Select members of the base dimension and use the member inspector, **Attributes** tab to associate them with members of the attribute dimension

---

When creating an attribute dimension, if you do not select a base dimension, a base dimension is associated with the newly created attribute dimension by default. The associated base dimension is either a newly created last sparse dimension or the last existing sparse dimension.

For example, if you create two sparse dimensions, dim1 and dim2, and then create an attribute dimension attr1, attr1 is associated with dim2 (the last sparse dimension that was created). If no sparse dimension was created recently, attr1 is associated with the last sparse dimension.

See Work with Essbase Attributes.

About Duplicate Member Names

When you create an Essbase cube, you can specify that duplicate (non-unique) member names and aliases are allowed in a cube outline, with some restrictions.

1. From the Essbase web interface, log in as a power user, and click **Create**.
2. Enter a unique application name and any cube name.
3. Select **Allow Duplicate Member Names**.
4. Click **OK**.

A duplicate member outline might, for example have a Market dimension and require two members named New York: one as a child member of the dimension parent member, Market, and one as a child of the member, New York. The member names are displayed as New York. The qualified member names are:

- [Market].[New York]
- [Market].[New York].[New York]

Copyright © 2019, 2026, Oracle and/or its affiliates.

To add a duplicate member name, enter the duplicate member in the outline. There are no additional requirements for adding a duplicate member. See Add Members to Outlines Manually.

Duplicate Member Outline Restrictions:

- If the outline is not enabled for duplicate members, then an error is returned when a duplicate member name is entered.
- Dimension names, generation names, and level names must always be unique, and sibling members under a parent member must always be unique.
- You must enable duplicate member names at the time you create the application. You cannot convert a unique member outline to a duplicate member outline.
- Duplicate member names applies to the entire outline and cannot be assigned only to a single dimension.
- After you migrate a cube with a unique member outline to Essbase 21c, you cannot change the outline to allow duplicate members. If you want the cube to allow duplicate members, you must convert the unique member outline to a duplicate member outline before migrating.

Set Dimension and Member Properties

To set dimension and member properties, open the outline in Edit mode.

Once in edit mode, choose a method to set dimension and member properties:

- Redwood
- Classic

Redwood

- In the Member inspector, by right clicking the member name and selecting **Inspect**.
- On the outline toolbar, by highlighting a member and selecting the options that you want on the toolbar.

Classic

- In the properties panel, by highlighting a member and, on the outline toolbar, under **Actions**, selecting **Display member properties panel on the right side**.
- On the outline toolbar, by highlighting a member and selecting the options that you want on the toolbar.

Open the Outline in Edit Mode

Before you can change or set member properties, you need to open the outline in Edit mode.

- Redwood

Copyright © 2019, 2026, Oracle and/or its affiliates.

- Classic

Redwood

1. On the Home page, navigate to the application, and select the Actions menu ⋯.
2. Choose **Databases** > *Database Name* > **Launch Outline**.
3. If the outline is locked, and you are an administrator, click **Unlock**.
   Before you forcefully unlock a locked outline, make sure that no one else is working with it.
4. Click **Edit**.

Classic

1. From the Applications page, expand the application.
2. Click the **Actions** menu to the right of the cube name and select **Outline.**
3. If the outline is locked, click **Unlock outline**.
4. Click **Edit outline**.

Set Member Properties while in Edit Mode

With the Essbase outline in Edit mode, you can set properties for individual members. You can make these changes using the keyboard or the member inspector.

To enable inline editing, put the outline in Edit mode, then double click on a member or in one of the columns to the right of the member name in the outline. For example, if you click along a row for a member you want to edit in the Data Storage Type column, you can use a menu to select a storage type for the highlighted member. If you double-click in the formula column, you can type a member formula.

With inline editing enabled you can:

- Type member names, or rename existing members.
- Use the Tab key to move from left to right between columns.
- Use the Enter key to move down in the outline tree.
- Use the space bar to expand menus, and use the up and down arrows to navigate the menu items.

You can also select multiple rows and change member properties in all selected rows at one time. For example, you can select several rows and change the member consolidation to + by clicking the + sign on the toolbar.

Set Properties in the Member Inspector

You can view and set Essbase outline member properties in the member inspector.

To open the Member Inspector:

1. Open the outline

   In the Redwood Interface:

   a. On the Home page, navigate to the application, and select the Actions menu ⋯.

Copyright © 2019, 2026, Oracle and/or its affiliates.

b. Choose **Databases** > *Database Name* > **Launch Outline**.

c. If the outline is locked, and you are an administrator, click **Unlock** .
Before you forcefully unlock a locked outline, make sure that no one else is working with it.

In the Classic Web Interface:

a. On the Applications page, expand the application.

b. Click the **Actions** menu to the right of the cube name and select **Outline**.

2. Click **Edit** .
3. Drill into the outline to find the member you want to update and select it.
4. Right click and select **Inspect**.
5. In the Member Inspector, choose where to make your modifications:
   - **General**
   - **Aliases**
   - **Formula**
   - **Attributes**
   - **User-defined Attributes**

See Setting Dimension and Member Properties.

Set General Properties

In the outline editor in the Essbase web interface, when you inspect a dimension or member's properties, the General tab is where you can view or modify basic Essbase dimension or member information (such as consolidation properties, storage properties, and comments).

The options available on the tab vary, depending on the outline type, and the dimension and member type. For example, the items available vary depending on whether the cube is block storage or aggregate storage, or whether you selected a dimension name or a member within a dimension.

The following is a partial list of general properties for dimensions and members.

Table 6-5 Dimension and Member General Properties

Field Name	Description	Applies to...
Name	Enter a dimension or member name. Use no more than 1024 bytes when naming dimensions, members, or aliases.	• Aggregate storage dimensions and members • Block storage dimensions and members
Comment	Enter a comment. Comments can contain up to 255 characters.	• Aggregate storage dimensions and members • Block storage dimensions and members

Copyright © 2019, 2026, Oracle and/or its affiliates.

Table 6-5 (Cont.) Dimension and Member General Properties

Field Name	Description	Applies to...
Dimension type	For a dimension, select: • None • Accounts • Time • Attribute	• Aggregate storage dimensions • Block storage dimensions
Dimension storage type	For a dimension, select: • Dense • Sparse There are two types of dimension storage available for block storage cubes: dense, and sparse. Dense dimensions have most of the data points filled, while sparse dimensions have most of the data points empty. The default storage type is sparse, but at least one dense dimension is required.	Block storage dimensions
Consolidation	For a member that is not a dimension or an attribute, select a consolidation operator: • + (addition) • – (subtraction) • * (multiplication) • / (division) • % (percentage) • ~ (ignore) • ^ (nonconsolidating) Addition (+) is the default. The ^ (nonconsolidating) operator applies only to block storage cubes.	• Aggregate storage members • Block storage members
Two-Pass	Select the **Two-Pass calc** check box to calculate the member during a second pass through the outline.	• Block storage stored members • For dynamic members, set solve order instead
Data Storage	Select an option to determine how data values for the current dimension or member are stored: • Store data • Dynamic calc (This option does not apply to aggregate storage cubes.) • Never share • Label only • Shared member	• Aggregate storage dimensions and members • Block storage dimensions and members
Member solve order	Specify a solve order between 0 and 127 to indicate the priority in which the member is calculated.	• Aggregate storage members • Dynamic block storage members

Copyright © 2019, 2026, Oracle and/or its affiliates.

Table 6-5 (Cont.) Dimension and Member General Properties

Field Name	Description	Applies to...
Hierarchy	Specify **Stored** (the default) or **Dynamic** or, for a dimension within an aggregate storage outline, select the **Multiple hierarchy enabled** option (which equates to selecting both **Stored** and **Dynamic**). The storage option that you select is applied to the hierarchy headed by the dimension or generation 2 member.	• Aggregate storage dimensions • Generation 2 aggregate storage members
Aggregate level usage	Select one of these options to provide a way for an administrator to influence both default and query-based view selection: • Default: Internal mechanisms decide how to create aggregations. • No aggregation: Aggregation is not performed along this hierarchy. All views selected are at the input level. • Top level only: (Applies to primary hierarchies.) Queries are answered directly from input data. • No intermediate levels: (Applies to primary hierarchies.) This selects top and bottom levels only.	Aggregate storage dimensions
Variance reporting expense	Members from the dimension tagged as type Accounts can have an Expense property value of True or False. When @VAR or @VARPER formulas are evaluated, Account members whose expense property is False will have opposite sign to those whose expense property is True. Example: Scenario dimension member Variance with formula @VAR(Actual, Budget). For Account dimension member Sales [with Expense property False], Variance member will be calculated as Actual-Budget. For Account dimension member COGS [with Expense property True], Variance member will be calculated as Budget-Actual.	Block storage accounts dimension and members

Copyright © 2019, 2026, Oracle and/or its affiliates.

Table 6-5 (Cont.) Dimension and Member General Properties

Field Name	Description	Applies to...
Account information	Time Balance: To use time balance properties, you must have a dimension tagged as Accounts and a dimension tagged as Time. • None: Apply no time balance property. Member values are calculated in the default manner. • Average: A parent value represents the average value of a time period. • First: A parent value represents the value at the beginning of a time period. • Last: A parent value represents the value at the end of a time period. Skip option: Select an option (None or Missing) to determine what values are ignored during time balance calculations. If you select None, then no values are ignored, and, if you select Missing, then #MISSING values are ignored. You can specify skip settings only if the time balance property is set as first, last, or average. • None • Missing You can set these properties for any members except Label Only members.	Block storage Accounts dimension only

Create Aliases

On the Aliases tab, you can assign alternate names, or aliases, to a dimension, member, or shared member. For example, in the Sample.Basic cube outline, members of the Product dimension are identified by product codes, such as 100, and by descriptive aliases, such as Cola.

1. Open the outline

   In the Redwood Interface:

   a. On the Home page, navigate to the application, and select the Actions menu ⋯ .

   b. Choose **Databases** > *Database Name* > **Launch Outline**.

   c. If the outline is locked, and you are an administrator, click **Unlock** 🔓.
   Before you forcefully unlock a locked outline, make sure that no one else is working with it.

   In the Classic Web Interface:

Copyright © 2019, 2026, Oracle and/or its affiliates.

   a. On the Applications page, expand the application.
   b. Click the **Actions** menu to the right of the cube name and select **Outline**.

2. Click **Edit** .
3. Drill into the outline to find the member you want to update and select it.
4. Right click and select **Inspect**.
5. Click **Aliases**.
6. In the field for the alias table you want to use, enter the value of the alias.
7. Click **Apply and Close**.
8. Click **Save** .

See Understand and Create Alias Tables and Setting Aliases.

Create Member Formulas

You can create and edit member formulas for both block storage and aggregate storage cubes. Formulas are calculated through default calculations and calculation scripts. Add member formulas using the outline editor in the Essbase web interface.

You can construct block storage member formulas from operators, functions, dimension names, member names, substitution variables, and numeric constants. To write formulas for block storage outlines, a set of calculation functions and operators is provided. For syntax and examples, see Calculation Functions.

Aggregate storage member formulas cannot be created using Calculator language. Instead, create them using Multidimensional Expression Language (MDX).

To work with member formulas, we'll import the Sample_Basic application.

1. In the Essbase web interface, click **Import**.
2. In the **Import** dialog box, click **Catalog**.
3. Go to gallery > Applications > Demo Samples > Block Storage, select **Sample_Basic.xlsx** and click **Select**.
4. Rename the application if needed and click **OK** to build the cube.

Let’s create an example member formula. Suppose you have a dynamic calc member called “Watchlist Products” and you want it to be the sum of products "100-10", "200-10" and "300-10."

- Redwood
- Classic

Redwood

1. On the Home page, navigate to the application, and select the Actions menu ••• .
2. Choose **Databases** > *Database Name* > **Launch Outline**.
3. If the outline is locked, and you are an administrator, click **Unlock** .

Copyright © 2019, 2026, Oracle and/or its affiliates.

Before you forcefully unlock a locked outline, make sure that no one else is working with it.

4. Click **Edit** .
5. Select the Product dimension, add a child called Watchlist_Products, and click **Add**.
6. Click **Close** to close the Add Members dialog box.
7. Right-click Watchlist_Products, and select **Inspect**, and then click **Formula**.
8. To show the member tree, click the **Show Member Tree** arrow on the left hand side of the Formula Editor.
9. To show the functions list, click the **Show Functions List** arrow on the right hand side of the Formula Editor.
10. In the member tree, in the left panel of the Formula Editor, drill into Product to find the first product member to add to your formula, "100-10". Right click the member name and click **Insert Name** to insert it into your formula.
11. Place the cursor after "100-10" in the formula being created and press the + key.
12. Use the member tree to pick the next product member to insert, 200-10. Right click the member name and click **Insert Name** to insert it into your formula.
13. Place the cursor after "200-10" and press the + key.
14. Repeat for the last product member, 300-10 and put a semi-colon (;) at the end of the formula.

    The formula should look like this: "100-10"+"200-10"+"300-10";
15. Click **Verify** and fix any errors.
16. Click **Apply and Close**.
17. For the Watchlist_Products member, double-click **Store data** in the **Data storage type** column, and select **Dynamic calculation**.
18. Click **Save** .

Classic

1. On the Applications page, expand the Sample application and select the Basic cube.
2. Click the **Actions** menu and select **Outline**.
3. Click **Edit**.
4. Select the Product dimension, add a child called Watchlist_Products, and press the Tab key.
5. Right click on Watchlist_Products and select **Inspect**.
6. Select the **Formula** tab.
7. In the member tree, in the left panel of the Formula Editor, drill into Product to find the first product member to add to your formula, “100-10.” Right click the member name and click **Insert Name** to insert it into your formula.
8. Place the cursor after "100-10" and press the + key.
9. Use the member tree to pick the next product member to insert, 200-10. Right click the member name and click **Insert Name** to insert it into your formula.
10. Place the cursor after "200-10" and press the + key.

Copyright © 2019, 2026, Oracle and/or its affiliates.

11. Repeat for the last product member, 300-10 and put a semi-colon (;) at the end of the formula.
    The formula should look like this: "100-10"+"200-10"+"300-10";
12. Click **Verify** and fix any errors.
13. Click **Apply and Close**.
14. In the Data Storage Type column for Watchlist_Products, select **Dynamic Calculation**.
15. Click **Save** to save the outline.
16. Choose from **Restructure Database Options**. See Restructure Cubes.

Member formulas like the one you just created can also include Essbase functions. When using Essbase functions in member formulas, use the **Function Name** menu on the right side of the formula editor to find and add calculation functions to the formula. See the Function description under the menu to read descriptions of each function.

See Developing Formulas for Block Storage Databases.

To write formulas for block storage outlines, a set of calculation functions and operators, known as the Calculator, or Calc, language, is provided. For descriptions of calculation commands and functions, see Calculation Commands and Calculation Functions.

Aggregate storage member formulas cannot be created using Calculator language. Instead, create them using Multidimensional Expression Language (MDX). See Aggregate Storage and MDX Outline Formulas and Developing Formulas on Aggregate Storage Outlines.

Set Attribute Associations

When manually working with attributes, use the outline editor and the Attributes tab in the member inspector, in the Essbase web interface. First associate attribute dimensions with base dimensions, and then associate attribute members with members of the base dimension.

Attributes are associated with Base dimensions; base dimensions are sparse standard dimensions containing members with which you would like to associate attributes.

Associate an Attribute Dimension with a Base Dimension

To associate an attribute dimension in Sample.Basic with a base dimension:

1. Open the outline:
   In the Redwood Interface,
   a. On the Home page, navigate to the application, and select the Actions menu ⋯ .
   b. Choose **Databases** > *Database Name* > **Launch Outline**.
   c. If the outline is locked, and you are an administrator, click **Unlock** 🔒.
      Before you forcefully unlock a locked outline, make sure that no one else is working with it.

   In the Classic Web Interface,
   a. On the Applications page, expand the application.
   b. From the **Actions** menu to the right of the cube name, select **Outline**.
2. Select a base dimension to which you want to associate an attribute dimension. For this exercise, choose Market.

Copyright © 2019, 2026, Oracle and/or its affiliates.

3. If you are not already in Edit mode, click **Edit** .
4. Right click on Market and select **Inspect**.
5. Click **Attributes**.
6. Select an attribute dimension. For this exercise, select Intro Date from the **Attribute Name** column.
7. Click the right arrow next to **Associated Attributes** to associate the selected attribute to the regular dimension you selected in step 4.
8. Click **Apply and close**.
9. Click **Save** to save the outline.
10. In **Restructure Database Options**, click **All Data** and click **Save**.

After you associate an attribute dimension with a base dimension, you must associate members of the attribute dimension with members of the base dimension; these members must all be from the same level in the base dimension.

Associate an Attribute Member with a Member of the Base Dimension

To associate an attribute member in Sample.Basic with a member of a base dimension:

1. With the Sample.Basic outline still open, click **Edit**.
2. Expand Market, then East and select New York.
   New York is the base member to which we'll associate an attribute.
3. Right click on New York and select **Inspect**.
4. Select **Attributes**.
5. Select the attribute member you want to associate with New York.
   - In the Redwood Interface, select the down arrow in the **Population** row and select the member.
   - In the Classic Web Interface, from the member tree, expand **Population** and select the member.
6. Click **Apply and Close**.
7. Click **Save** to save the outline.

See Working with Attributes.

Create User-Defined Attributes

You can create, assign, and unassign user-defined attributes (UDAs) in Essbase block storage (BSO) databases. A UDA is a descriptor you can use to filter data retrievals. For example, you might create a UDA called Major Market and assign it to all members that are part of a major market.

Like attributes, UDAs help you filter data retrievals based on characteristics. Unlike attributes, UDAs have no built-in calculation functionality. However, UDAs can be assigned to dense and sparse dimensions, whereas attributes can be assigned to only sparse dimensions. Also, a UDA can be assigned to any level or generation in a BSO dimension.

- Redwood

Copyright © 2019, 2026, Oracle and/or its affiliates.

- Classic

Redwood

1. On the Home page, navigate to the application, and select the Actions menu ⋯.
2. Choose **Databases** > *Database Name* > **Launch Outline**.
3. If the outline is locked, and you are an administrator, click **Unlock**.

   Before you forcefully unlock a locked outline, make sure that no one else is working with it.
4. If the outline is not already in edit mode, click **Edit**.
5. Highlight a member to which you would like to assign a UDA.
6. Right click the member and select **Inspect**.
7. Click **User-defined Attributes**.
8. In the **User-defined Attributes** field, enter a UDA name and press the Enter key.
9. Click **Apply and Close** to create the UDA for the dimension and assign the new UDA to the member.
10. Click **Save** to save the outline.

Classic

1. On the Applications page, expand the application.
2. Click the **Actions** menu to the right of the cube name and select **Outline**.
3. If the outline is not already in edit mode, click **Edit**.
4. Highlight a member to which you would like to assign a UDA.
5. Right click the member and select **Inspect**.
6. Click the User-defined Attributes tab.
7. In the **User-defined Attributes** field, enter a UDA name and press the Enter key.
8. Click **Apply and Close** to create the UDA for the dimension and assign the new UDA to the member.
9. Click **Save** to save the outline.

Select the Member Properties to Display in the Outline

You can customize which member properties to display in the outline editor in the Essbase web interface.

- Redwood
- Classic

Copyright © 2019, 2026, Oracle and/or its affiliates.

Redwood

1. On the Home page, navigate to the application, and select the Actions menu ⋯.
2. Choose **Databases** > *Database Name* > **Launch Outline**.
3. If the outline is locked, and you are an administrator, click **Unlock**.
   Before you forcefully unlock a locked outline, make sure that no one else is working with it.
4. Click **Edit**.
5. On the outline toolbar, click **Select Columns to Display**.
6. Under **Available Columns**, select the items you want to display in the outline, then click the right arrow to add them to the **Selected Columns** list.
7. Under **Selected Columns**, select the items you do not want displayed in the outline and use the left arrow to add them to the **Available Columns** list.
8. Optional: select the **Show in Name** check boxes (available only for some properties) to display those properties next to the dimension or member names instead of in the subsequent columns.
9. Click **Apply and Close**.

Only the selected properties are displayed in the outline.

Classic

1. On the Applications page, expand the application.
2. From the **Actions** menu, to the right of the cube name, select **Outline**.
3. Select **Edit outline**.
4. On the outline toolbar, under **Inspect**, select **Display selected columns in the table**.
5. In the **Select the member properties to display** dialog box, clear the check box next to **Property name** to deselect all the properties.
6. Select the properties that you want to display in the outline.
7. Optional: select the **Show in Name** check boxes (available only for some properties) to display those properties next to the dimension or member names instead of in the subsequent columns.
8. Press **Apply and Close**.
9. Click **Save** to save the outline.

Only the selected properties are displayed in the outline.

Compare Outlines

You can compare two outlines in the Essbase web interface. The outlines must be of the same type, both aggregate storage, or both block storage. They can be on the same Essbase server, or on different Essbase servers.

To demonstrate outline comparison, we'll import two sample applications.

1. In the Essbase web interface, click **Import**.

Copyright © 2019, 2026, Oracle and/or its affiliates.

2. In the **Import** dialog box, click **Catalog**.
3. Go to gallery > Applications > Demo Samples > Block Storage, select **Sample_Basic.xlsx** and click **Select**.
4. Click **OK** to build the cube.
5. Repeat steps 1 through 4, this time building **Demo_Basic.xlsx**.

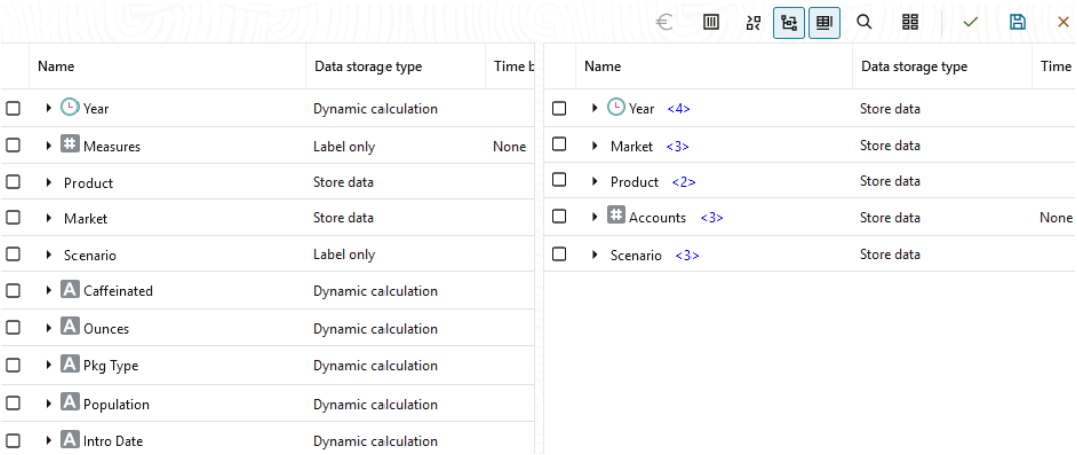
To open two outlines side-by-side,

- Redwood
- Classic

Redwood

1. On the Home page, navigate to the **Sample** application, and select the Actions menu ⋯ .
2. Choose **Databases** > **Basic** > **Launch Outline**.
3. On the outline toolbar, click the **Compare Outlines** icon.

4. In the **Compare** dialog box, keep the **(Current)** connection and select the **Demo** application and the **Basic** database (cube).
   Alternatively, you can select a connection to another server in the **Connection** drop down menu, and compare two outlines on different servers.
5. Click **Open**, and **Demo.Basic** is opened to the right of **Sample.Basic**. **Demo.Basic** is read only. The read only outline is the source outline. The writeable outline is the target outline.



Classic

1. On the **Applications** page, expand the **Sample** application, and from the **Actions** menu to the right of the cube name, choose **Outline**.
2. On the outline toolbar, click the **To compare outlines** icon.

Copyright © 2019, 2026, Oracle and/or its affiliates.

3. In the **Compare outline** dialog box, keep the **(Current)** connection and select the **Demo** application and the **Basic** database.
   Alternatively, you can select a connection to another server in the **Connection** drop down menu, and compare two outlines on different servers.
4. Click **Open**, and **Demo.Basic** is opened to the right of **Sample.Basic**. Notice that **Demo.Basic** is read only. The read only outline is the source outline. The writeable outline is the target outline.

To synchronize expanding and collapsing hierarchies,

- Redwood
- Classic

Redwood

1. Expand the **Year** dimension in **Sample.Basic** and notice that the **Year** dimension in **Demo.Basic** is expanded as well.
2. In the outline toolbar, next to **Compare**, click **Auto expand/collapse target members** to deselect it (it is selected by default) .
3. Collapse the **Year** dimension in **Demo.Basic** and notice that the **Year** dimension in **Sample.Basic** is still expanded. Collapse the **Year** dimension in **Sample.Basic**.

Classic

1. Expand the **Year** dimension in **Sample.Basic** and notice that the **Year** dimension in **Demo.Basic** is expanded as well.
2. In the outline toolbar, under **Compare**, click the **Auto expand/collapse target members** icon to deselect it (it is selected by default) .
3. Collapse the **Year** dimension in **Demo.Basic** and notice that the **Year** dimension in **Sample.Basic** is still expanded. Collapse the **Year** dimension in **Sample.Basic**.

To synchronize scrolling,

Copyright © 2019, 2026, Oracle and/or its affiliates.

- Redwood
- Classic

Redwood

1. Click **Auto expand/collapse target members** and expand several dimensions so that you need to scroll in order to see the entire outline.
2. Confirm that in the outline toolbar **Enable/disable synchronize scroll** is selected.
3. Scroll the **Sample.Basic** outline and observe that the **Demo.Basic** outline scrolls in sync with it.

Classic

1. Click the **Auto expand/collapse target members** icon and expand several dimensions so that you need to scroll in order to see the entire outline.
2. Confirm that in the outline toolbar, under **Compare**, the **Enable/disable synchronize scroll** icon is selected .
3. Scroll the **Sample.Basic** outline and observe that the **Demo.Basic** outline scrolls in sync with it.

To synchronize showing and hiding columns,

- Redwood
- Classic

Redwood

1. In the outline toolbar, click **Select Columns to Display**.

2. In the **Selected Columns** column, select **operator** and **data storage type** and click the left arrow. Notice that the displayed columns in both **Sample.Basic** and **Demo.Basic** change accordingly.

Classic

1. In the outline toolbar, under **Inspect**, click **Display selected columns in the table**.

Copyright © 2019, 2026, Oracle and/or its affiliates.

2. In the **Select member properties to display** dialog box, clear the selections for **operator** and **data storage type** and click **Apply and Close**. Notice that the displayed columns in both **Sample.Basic** and **Demo.Basic** change accordingly.

To copy members from the source (compared) outline to the target (writeable) outline,

1. In the upper right-hand corner of the outline editor, click **Edit outline** .
2. In the **Demo.Basic** outline, expand the **Product** dimension.
3. Right click **Audio** and select **Copy**.
4. In the **Sample.Basic** outline, expand the **Product** dimension.
5. Right click **Product**, and select **Paste**, and then **As child**.
   Audio and its children are added as children of **Product**.

To search either the source (compared) or the target (writeable) outline,

1. In the outline toolbar, click **Search Members** .
2. Notice that search bars open for both **Sample.Basic** and **Demo.Basic**, allowing you to search either one.

> **Note**
>
> The search bar does not open for the target outline when it is from an Essbase version prior to 21c.

Copy and Paste Members Within and Between Outlines

In the outline editor, you can copy and paste members within a non-unique outline, between two different outlines, shown on different browser tabs, or from a source to a target outline when using outline compare.

Copy and paste members within a non-unique outline:

To successfully copy and paste members within a non-unique outline, you must follow the rules for non-unique outlines. See About Duplicate Member Names.

1. In the outline editor, open a non-unique outline.
2. If the outline is locked, and you are an administrator, click **Unlock** .
   Before you forcefully unlock a locked outline, make sure that no one else is working with it.
3. Click **Edit** .
4. Right-click a member and select **Copy**.
5. In another location, where allowed according to the rules for non-unique outlines, right click a member and select **Paste** and then **As child** or **As sibling**.
6. The member is added as "copy of *member*." Rename the member to the name that you want.
7. Validate and save the outline.

Copy and paste a member from another outline shown on a different browser tab:

Copyright © 2019, 2026, Oracle and/or its affiliates.

1. Open an outline.
2. If the outline is locked, and you are an administrator, click **Unlock**.
   Before you forcefully unlock a locked outline, make sure that no one else is working with it.
3. Click **Edit**.
4. Open a second outline on another browser tab or browser window. This outline can be from the same Essbase server or a different Essbase server.
5. If the outline is locked, and you are an administrator, click **Unlock**.
   Before you forcefully unlock a locked outline, make sure that no one else is working with it.
6. Click **Edit**.
7. Right-click a member from one of the outlines and select **Copy**.
8. Right-click a member from the other outline in a valid location and select **Paste** and then **As child** or **As sibling**.
9. If the member name already exists in the outline, the member is added as "copy of *member*." Rename the member to the name that you want.
10. Validate and save the outline.

Copy and paste a member from a compared outline:

1. Open an outline.
2. If the outline is locked, and you are an administrator, click **Unlock**.
   Before you forcefully unlock a locked outline, make sure that no one else is working with it.
3. Click **Edit**.
4. In the outline toolbar, click the **Compare Outlines** icon.

5. In the Compare Outlines dialog box, open another outline, either on the same Essbase server or a different Essbase server. See Compare Outlines.
6. In the source outline (the second outline you opened, the non-writeable one), expand a dimension and select a member.
7. Right-click and select **Copy**.
8. In the target outline, select a member in a valid location to add the member and right-click.
9. Select **Paste** and then **As child** or **As sibling**.
10. If the member name already exists in the outline, the member is added as "copy of *member*." Rename the member to the name that you want.
11. Validate and save the outline.

Copyright © 2019, 2026, Oracle and/or its affiliates.

7

Manage Application and Cube Artifacts and Settings

You can manage many application and cube artifacts and settings in the Essbase web interface.

Topics:

- Set Calculation-Related Cube Properties
- Unlock Objects
- Remove Data Locks
- Set Buffer Sizes to Optimize Reports
- Understand Transaction Semantics in Essbase
-

Set Calculation-Related Cube Properties

Using the Essbase web interface, you can set calculation properties for the current Essbase block storage (BSO) cube, by selecting or deselecting options in the database settings. You can select whether to aggregate missing values, create blocks on equations, or enable two-pass calculation.

- **Aggregate missing values**: If you never load data at parent levels, selecting this option may improve calculation performance. If this option is selected and you load data at the parent level, then the parent-level values are replaced by the results of the cube consolidation, even if the results are #MISSING values.
- **Create blocks on equations**: If this option is selected, then when you assign a non-constant value to a member combination for which no data block exists, a data block is created.
  Selecting this option can produce a very large cube.
- **Two-Pass calculation**: If this option is selected, then after a default calculation, members that are tagged as two-pass are recalculated. Do not use two-pass calculation for hybrid mode cubes. Use solve order instead.

1. Navigate to the Settings tab.
   In the Redwood Interface, on the Home page, open the application and then open the database (cube).
2. Click the Settings tab.
3. Click **Calculation**.
4. Select the options that you want.
5. Click **Save**.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Unlock Objects

Essbase uses a checkout facility for cube objects (such as calculation scripts, rules files, and outlines). Objects are locked automatically when they are in use, and the locks are deleted when they are no longer in use.

You can view and unlock Essbase objects, according to your security role. Service administrators can unlock any object. Other users can unlock only those objects that they locked.

1. Navigate to **Locks**.

   In the Redwood Interface:

   a. On the Home page, open the application and then open the database (cube).

   b. Click **General** in the left hand panel.

   In the Classic Web Interface:

   a. On the Applications page, expand the application.

   b. From the **Actions** menu, to the right of the cube name, click **Inspect**.

2. Click **Locks**.

3. From the Display menu, select **Objects**.

4. Select the object you want to unlock and click **Unlock**.

In the Classic Web Interface, you can also unlock outlines directly from the **Actions** menu, to the right of the cube name. Click the **Actions** menu icon and select **Unlock outline**.

Remove Data Locks

Data locks apply to block storage cubes only.

Occasionally, you may need to release a lock that you created in the cube, generally from a Smart View Submit Data action. For example, if you're calculating a cube that has active locks on data, and the calculation encounters a lock, then the calculation must wait. If you release the lock, the calculation can resume.

You can always unlock data that you locked. To remove another user's data locks, you must have the Application Manager or Database Manager role.

Use these steps to remove data locks from Essbase cubes in the Essbase web interface.

1. Navigate to **Locks**.

   In the Redwood Interface:

   a. On the Home page, open the application and then open the database (cube).

   b. Click **General** in the left hand panel.

   In the Classic Web Interface:

   a. On the Applications page, expand the application.

   b. From the **Actions** menu, to the right of the cube name, click **Inspect**.

2. Click the **Locks** tab.

3. From the Display menu, select **Blocks**.

Copyright © 2019, 2026, Oracle and/or its affiliates.

4. Select the lock and click **Unlock**.

Set Buffer Sizes to Optimize Reports

You can change the retrieval buffer and retrieval sort buffer sizes in Essbase to optimize Report Writer reports and Smart View Query Designer queries.

The time required to generate a report varies depending upon factors such as the size of the cube you are reporting from, the number of queries included in the script, and the size of the retrieval buffer and retrieval sort buffer.

Configurable variables specify the size of the buffers used for storing and sorting data extracted by retrievals. The retrieval buffer and retrieval sort buffer should be large enough to prevent unnecessary read and write activities. You can set them in the Essbase web interface or in MaxL.

The **retrieval buffer** holds extracted row data cells before they are evaluated by the RESTRICT or TOP/BOTTOM Report Writer commands. The default size is 20 KB. The minimum size is 2 KB. Increasing the size may improve retrieval performance.

When the retrieval buffer is full, the rows are processed and the retrieval buffer is reused. If this buffer is too small, frequent reuse of the area can increase retrieval times. If this buffer is too large, too much memory may be used when concurrent users perform queries, also increasing retrieval times.

The **retrieval sort buffer** holds data until it is sorted. Report Writer and Query Designer (in Smart View) use the retrieval sort buffer. The default size is 20 KB. The minimum size is 2 KB. Increasing the size may improve retrieval performance.

To set the retrieval buffer size and retrieval sort buffer size:

1. Navigate to **Buffers**.
   In the Redwood Interface:
   a. On the Home page, open the application and then open the database (cube).
   b. Click **Settings**, and click **Buffers**.

   In the Classic Web Interface:

   On the Applications page, go to the cube inspector, then go to the **Settings** tab and select **Buffers**.
2. Add the values you want to use, and click **Save**.

Understand Transaction Semantics in Essbase

When a cube is in read/write mode, Essbase considers every update request to the server (such as a data load, a calculation, or a statement in a calculation script) as a transaction.

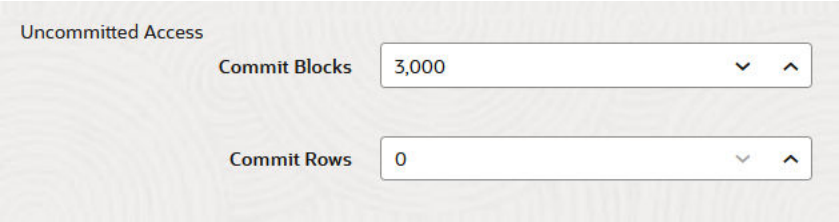
Each transaction has a current state: active, committed, or aborted. When data is committed, it is taken from server memory and written to the cube on disk.

The Commit Blocks/Commit Rows options in the Transactions tab of the cube inspector settings indicate the frequency at which Essbase commits data blocks or rows.

Essbase allows transactions to hold read/write locks on a block-by-block basis; Essbase releases a block after it is updated but does not commit blocks until the transaction completes or until a set limit (a “synchronization point”) has been reached.

Copyright © 2019, 2026, Oracle and/or its affiliates.

You control when Essbase performs an explicit commit operation by specifying the following synchronization point parameters:



- **Commit Blocks** (number of blocks modified before a synchronization point occurs). Essbase commits after the specified number of blocks has been reached. This frequency may adjust dynamically during a calculation.
  If you set Commit Blocks to 0, the synchronization point occurs at the end of the transaction.
- **Commit Rows** (number of rows to load before a synchronization point occurs). The default is 0, which means that the synchronization point occurs at the end of the data load.

If either Commit Blocks or Commit Rows has a nonzero value, a synchronization point occurs when the first threshold is reached. For example, if Commit Blocks is 10 but Commit Rows is 0 and you load data, a synchronization point occurs after 10 blocks are updated. If Commit Blocks is 5 and Commit Rows is 5 and you load data, a synchronization point occurs after 5 rows are loaded or 5 blocks are updated, whichever happens first.

If Essbase Server runs on a Windows independent deployment, Commit Blocks is set to 3000 by default.

If a user-defined threshold is exceeded during an operation, Essbase creates a synchronization point to commit the data processed to that point. Essbase creates as many synchronization points as are necessary to complete the operation.

Essbase analyzes the value of Commit Blocks and Commit Rows during its analysis of feasibility for parallel calculation use. If Essbase finds the values set too low, it automatically increases them.

Essbase retains redundant data to enforce transactional semantics. Allow disk space for double the size of the database to accommodate redundant data, particularly if both Commit Blocks and Commit Rows are set to 0.

To set commit blocks and commit rows

1. Navigate to **Transactions** in the Essbase web interface.
   In the Redwood Interface:
   a. On the Home page, open the application and then open the database (cube).
   b. Click **Settings** and click **Transactions**.

   In the Classic Web Interface:
   a. On the Applications page, expand the application.
   b. From the **Actions** menu, to the right of the cube name, click **Settings**, then click **Transactions**.
2. Make your selections and click **Save**.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Ask Essbase

Copyright © 2019, 2026, Oracle and/or its affiliates.

8

Run and Manage Jobs Using the Web Interface

The Jobs page in the Essbase web interface is a centralized interface from which to run routine operations and processes in the Essbase platform.

Essbase administrators or users with execute permissions on certain applications can use the Jobs page to quickly execute jobs such as clearing and loading data, importing and exporting applications, running calculations and much more.

The Jobs page is convenient for one-time execution of administrative tasks, but it is not a replacement for scripted administration of Essbase platform jobs. MaxL, CLI, REST, and API programs are the most efficient way to schedule jobs for production activities and life cycle maintenance.

View Job Status and Details

Essbase users have access to job status based on their assigned roles. For example, a Service Administrator can see all jobs; if you have the User role, you can see only the jobs you ran.

Because Essbase jobs run in the background, you must refresh the Jobs page to view their status.

The job listing shows all the jobs for all the applications provisioned to the logged in user. You can scroll down to see the history of all the jobs that you ran.

1. On the Home page, click **Jobs**.
2. Click **Refresh** to refresh once, or toggle **Auto Refresh** to refresh the jobs every few seconds. In Cube Designer, job status refreshes automatically.

You can also view details for an individual job. To view job details, click the **Actions** menu to the right of the job listing, and select **Job Details** to see input and output details for a job.

You can terminate jobs on the Console page, on the **Sessions** tab:

1. On the Applications page, click **Console**, and then click **Sessions**.
2. Select the user, and the applications and cube in which the job is running.
3. Select **Terminate All**.
   This terminates all jobs in the application and cube, started by the selected user.

Execute Jobs

From the Jobs page in the Essbase web interface, you can build dimensions, build aggregations, clear data and aggregations, execute report scripts, export data and Excel workbooks, export and import LCM, export to table format, load data, run calculations, run MDX scripts, and vectorize the outline.

You can execute numerous types of jobs. For each, you choose an option from the **New Job** drop-down list, and then provide the necessary information.

Copyright © 2019, 2026, Oracle and/or its affiliates.

You can execute up to 10 jobs concurrently, or change the default setting.

Aggregate storage:

- Build Aggregations
- Clear Aggregations

Block storage:

- Export to Table Format
- Run Calculation

Aggregate storage and block storage:

- Build Dimension
- Clear Data
- Export Data
- Export Excel
- Export LCM
- Import LCM
- Load Data
- Run MDX
- Vectorize Outline

Build Aggregations

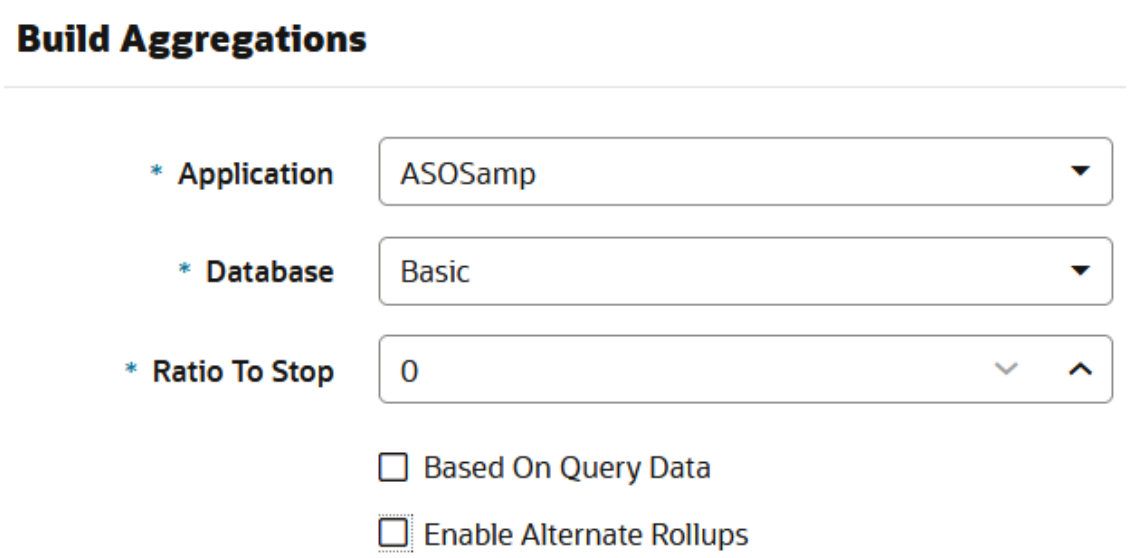
Build an aggregation. Essbase selects aggregate views to be rolled up, aggregates them based on the outline hierarchy, and stores the cell values in the selected views.

Building aggregations requires Database Access permission.

Aggregations are intermediate stored consolidations of aggregate storage cubes, consisting of one or more aggregate views. Aggregate views store upper-level intersections. This supports query performance by avoiding dynamic aggregations on the most commonly queried intersections.

If an aggregation includes aggregate cells dependent on level 0 values that are changed through a data load, the higher-level values are automatically updated at the end of the data load process.

Copyright © 2019, 2026, Oracle and/or its affiliates.



To build aggregations:

1. On the Home page, click **Jobs**.
2. From the **New Job** menu, select **Build Aggregations**.
3. For **Application**, choose an application.
4. For **Database**, choose a cube.
5. Optionally, enter a non-zero value for **Ratio To Stop**.
   Leaving **Ratio To Stop** at zero (the default) means there is no stopping ratio set.

   Consider this option if there is no known common type of query executed by your cube's users, and you want to improve performance by limiting the cube's growth. Essbase aggregates the selected views, with the exception that the maximum growth of the aggregated cube must not exceed the given ratio. For example, if the size of a cube is 1 GB, specifying the total size as 1.2 means that the size of the resulting data cannot exceed 20% of 1 GB, for a total size of 1.2 GB.
6. Check or clear the box for **Based on Query Data**.
   If you check the box for **Based on Query Data**, Essbase aggregates a selection of views that is defined based on analysis of user querying patterns. This is a good approach if similar types of queries are typically executed by the users of your cube.

   This check box has no effect unless you have first enabled query tracking. For general information about query tracking, see Selecting Views Based on Usage.

   After you've enabled query tracking, allow sufficient time to collect user data-retrieval patterns before running this job. A good approach is to prepare a set of your most important and long running queries, enable query tracking, run the prepared set of queries, and then run this job to create an aggregate view based on the query tracking.

   While query tracking is enabled, the cost of retrieving cells is recorded for every level combination. This recording continues until the application is shut down or until you turn off query tracking (using the MaxL statement alter database <dbs-name> disable query_tracking).
7. Select whether to enable alternate rollups.
   Consider checking this box if your cube implements alternate hierarchies for shared members or attributes, and you want to include them in the aggregation.
8. Click **Submit**.

Copyright © 2019, 2026, Oracle and/or its affiliates.

See Also

Aggregation of Data in an ASO Cube

Hierarchies in ASO Cubes

Clear Aggregations

You can clear aggregations using a Job in the Essbase web interface. Essbase clears aggregations from your aggregate storage (ASO) cube, removing non level-0 data. User queries then calculate retrieved values dynamically from the level-0 values.

Clearing aggregations requires Database Update permission.

1. On the Home page, click **Jobs**.
2. From the **New Job** menu, select **Clear Aggregations**.
3. For **Application**, choose an application.
4. For **Database**, choose a cube.
5. Click **Submit**.

See Build Aggregations and Clear Aggregated Data from the Cube.

Export to Table Format

The Export to Table Format job in the Essbase web interface enables you to export a cube into Excel, in tabular format. Essbase generates flattened output from the cube into Excel. Exporting a cube in tabular format can facilitate moving and sharing data between Essbase and a relational source.

Exporting to table format requires at least Database Update application permission.

This exported tabular data is organized into columns with headers that Essbase can use to deploy a new multidimensional cube. See Export a Cube to Tabular Data.

To export a cube in tabular format,

1. On the Home page, click **Jobs**.
2. From the **New Job** menu, select **Export to Table Format**.
3. For **Application**, select an application.
4. Choose whether to export dynamic blocks.

   If you choose **Export Dynamic Blocks**, cells for dynamic members in the dense dimensions are exported.
5. Click **Submit**.

Run Calculation

The Run Calculation job in the Essbase web interface enables you to execute a stored block storage (BSO) calculation script. Calculation scripts enable you to procedurally calculate a BSO cube; for example, you can calculate one part of a cube before another, or copy data values between members.

Running calculation scripts requires at least Database Update permission, as well as provisioned access to the calculation script.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Before you can run a calculation script, you must write and save the calculation script to the cube, or upload the script, as a `.csc` file, to the cube directory. See Work with Files and Artifacts.

To run a calculation from the Essbase web interface:

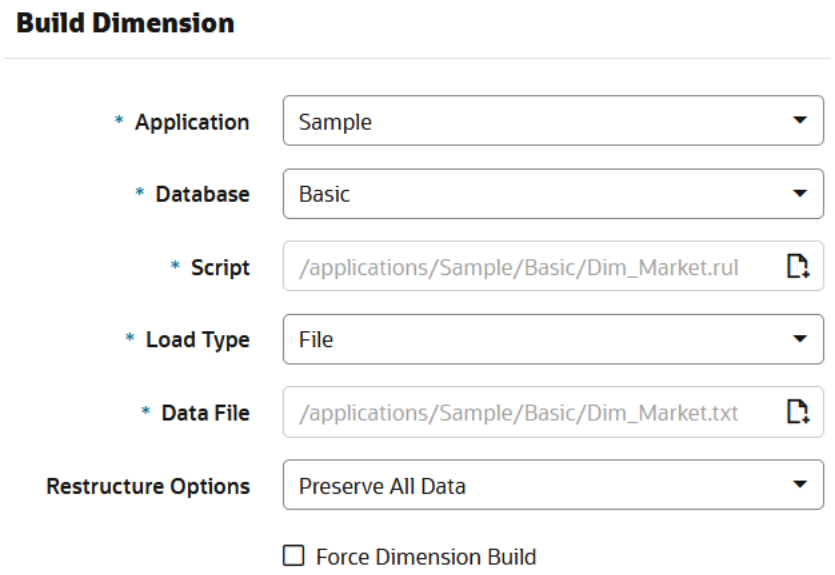
1. On the Home page, click **Jobs**.
2. From the **New Job** menu, select **Run Calculation**.
3. For **Application**, choose an application.
4. For **Database**, choose a cube.
5. Select a calculation script.
6. Click **Submit**.

See Calculate Cubes.

Build Dimension

Run a dimension build. Building dimensions in Essbase is the process of loading dimensions and members to a cube outline using a data source and a rule file.

Building dimensions requires at least Database Manager permission.



This procedure covers how to build dimensions using the **File** load type. **SQL** and **Datasource** types are also available. For information on loading different sources of data, see Define Rules that Query External Sources.

To build a dimension:

1. On the Home page, click **Jobs**.
2. From the **New Job** menu, select **Build Dimension**.
3. For **Application**, choose an application.
4. For **Database**, choose a cube.
5. Click the Actions menu to the right of the **Script** field and select a rule file.

Copyright © 2019, 2026, Oracle and/or its affiliates.

6. Select the **File** load type.
7. Click the actions menu to the right of the **Data File** field to select a data file.
8. Choose a restructure option.
   - **Preserve All Data**: Preserves all existing data.
   - **Preserve No Data**: Discards existing data (valid for block storage and aggregate storage cubes).
   - **Preserve Leaf Level Data**: Preserves data in existing level 0 blocks (block storage only). If you select this option, all upper-level blocks are deleted before the cube is restructured. After restructure, only data in level 0 blocks remains.
   - **Preserve Input Data**: Preserves existing input-level blocks (block storage only).
9. Select **Force Dimension Build** if you want to forcibly quit all ongoing jobs on that database and run the dimension build job. If you don't select this option, dimension build jobs fail if there are other active jobs on the database.
10. Click **Submit**.

Clear Data

When you run the Clear Data job in the Essbase web interface, Essbase changes the values of cells containing data to #Missing. You can perform a full clear or a partial clear. Clearing data requires at least Database Update permission.

1. On the Home page, click **Jobs**.
2. From the **New Job** menu, select **Clear Data**.
3. For **Application**, choose an application.
4. For **Database**, choose a cube.
5. Choose a **Clear Data option**.
   - For block storage cubes, choose:
     - **All Data**—All data, linked objects, and the outline are cleared
     - **Upper-Level Blocks**—Upper-level blocks are cleared
     - **Non-Input Blocks**—Non-input blocks are cleared
   - For aggregate storage cubes, choose:
     - **All Data**—All data, linked objects, and the outline are cleared
     - **All Aggregations**—All aggregated data is cleared
     - **Partial Data**—Only specified data region is cleared.
       Specify the data regions to clear in the **MDX Expression** text box.

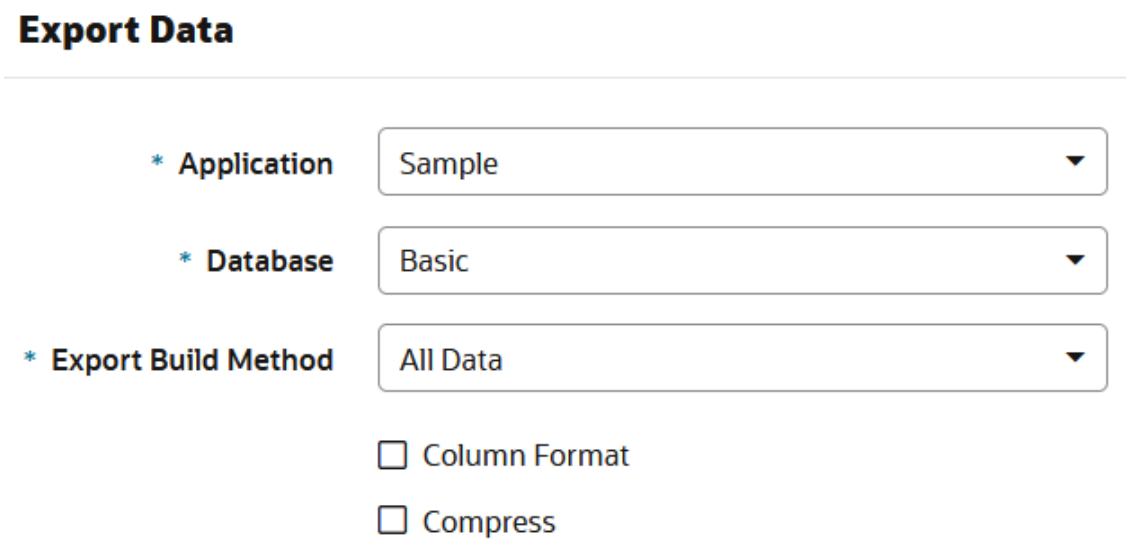
       Select the **Physical** check box to physically remove the cells specified in the **MDX Expression** text box from the cube. See Clear Data from Aggregate Storage Cubes.
6. Click **Submit**.

Export Data

The Export Data job in the Essbase web interface enables you to export data to a text file. You can choose which data level to export, whether to export in column format, and whether to

Copyright © 2019, 2026, Oracle and/or its affiliates.

compress the data to a ZIP file. Exporting data requires at least Database Manager permission.



To export data,

1. On the Home page, click **Jobs**.
2. From the **New Job** menu, select **Export Data**.
3. For **Application**, choose an application.
4. For **Database**, choose a cube.
5. For **Export Build Method**, select a data level.

   You can choose from **All Data**, **Level 0 Data**, or **Input Data**.
6. Select **Column Format** to export the data in column format.
7. Select **Compress** to export the data to a ZIP file.
8. Click **Submit**.

To download the exported data file,

1. On the applications page, click **Jobs**.
2. Select the Actions menu to the right of the export job.
3. Select **Job Details**.
4. To view the data file, you can click the **Output Path** link, or to download the file, select download .

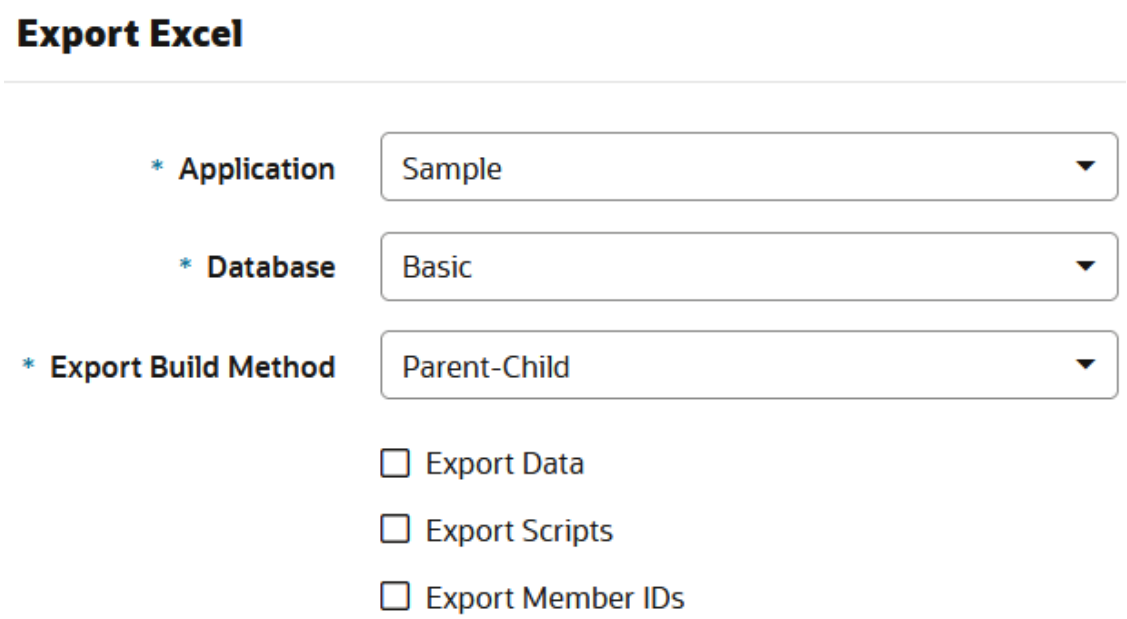
   The exported data file is stored in the database folder in the catalog.

Export Excel

The Export Excel job in the Essbase web interface enables you to export an Essbase cube to an Excel application workbook. Application workbooks comprise a series of worksheets, which can appear in any order, and define a cube. The application workbook can later be imported to create a new cube.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Exporting a cube to Excel requires at least Database Manager permission.



To export to Excel,

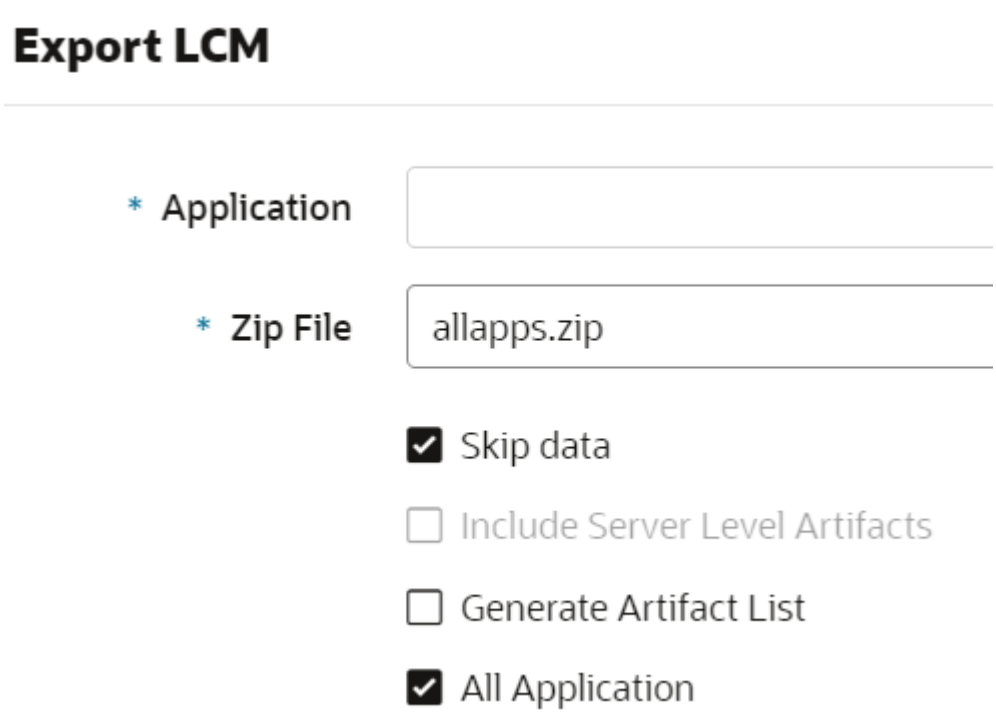
1. On the Home page, click **Jobs**.
2. From the **New Job** menu, select **Export Excel**.
3. For **Application**, choose an application.
4. For **Database**, choose a cube.
5. Select a build method.

   See Understanding Build Methods.
6. Choose whether to export data. If you export data, a data worksheet is added to the application workbook.
7. Choose whether to export scripts. If you export scripts, Calc / MDX sheets are added to the application workbook, if calculation scripts and MDX scripts exist in the cube.
8. Choose whether to export Member IDs. If you export member IDs, they are added to the application workbook.
9. Click **Submit**.

Export LCM

The Export LCM job in the Essbase web interface enables you to back up Essbase cube artifacts to a Lifecycle Management (LCM) .zip file. To run this job, you must have at least Application Manager permission, or you must be the power user who created the application.

Copyright © 2019, 2026, Oracle and/or its affiliates.



To back up cube artifacts to a .zip file,

1. On the Home page, click **Jobs**.
2. From the **New Job** menu, select **Export LCM**.
3. Select an application in the **Application** selector, OR, click All Application to export all applications to the zip.
4. Enter a name for the .zip file. If no location is specified, it is saved in `<Application Directory>/catalog/users/<user_name>`.
5. Optionally, select any of the following backup-related actions:
   - **Skip Data** - Excludes data from the backup.
   - **Include Server Level Artifacts** - Include globally defined connections and Datasources as part of the export.
   - **Generate Artifact List** - Generate a text file containing a complete list of the exported artifacts. You can use this text file to manage the import of artifacts. For example, you can rearrange the order of artifacts in the list to control the order in which they are imported. You can skip importing some artifacts by removing or commenting out items in the list.
6. Click **Submit**.

Notes

By default, the ZIP file is stored on the Essbase Server file catalog, in the user directory of the user who exported it.

Lifecycle Management (LCM) import operations (and Migration Utility import) are not supported for migration of federated partitions. Federated partitions (applicable only for deployments on OCI) must be recreated manually on the target.

See also: LcmExport: Back Up Cube Files.

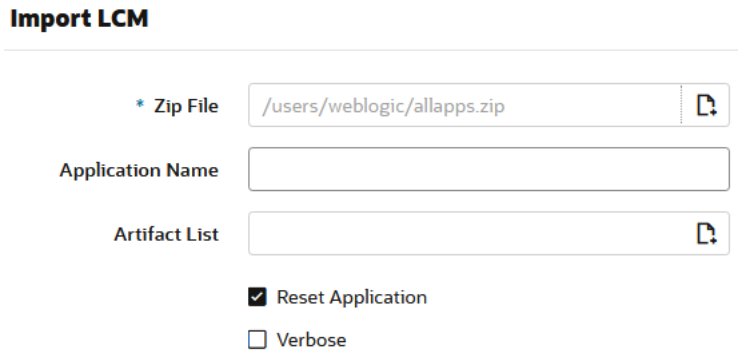
Copyright © 2019, 2026, Oracle and/or its affiliates.

Import LCM

The Import LCM job in the Essbase web interface enables you to restore Essbase cube artifacts that were backed up to an Essbase Lifecycle Management (LCM) ZIP file.

To run this job, you need at least user role with Application Manager permission, or, you must be the power user who created the application.

Before you start, locate the Lifecycle Management (LCM) ZIP file that was created using the Export LCM job (or the LcmExport: Back Up Cube Files CLI command).



To restore cube artifacts from a Lifecycle Management (LCM) ZIP file:

1. On the Home page, click **Jobs**.
2. From the **New Job** menu, select **Import LCM**.
3. Select the LCM export ZIP file.
4. Enter the target application name.
5. Select the **Artifact List**, if applicable.

   If server-level artifacts were included in the LCM export, you can select the artifact list to also include server-level artifacts on LCM import.
6. Select or clear **Reset Application**.

   Choosing to reset the application deletes the existing application and replaces it with the provided LCM file. If reset application is not selected, and the specified application name is the same as an existing application, the Import LCM job fails.
7. Select whether to use verbose descriptions.

   Choosing **Verbose** enables extended descriptions.
8. Click **Submit**.

Notes

To check the job status, click on the **Actions** menu to the right of the job and select **Job Details**.

After the LCM import completes, you may need to take further action to restore migrated connections to external sources. To do this, open the connection and enter the password.

LCM Import does not migrate location alias credentials. You must replace your location alias credentials, either by recreating location aliases using MaxL, or by editing the location alias credentials in the XML exported by LCM Export.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Lifecycle Management (LCM) import operations (and Migration Utility import) are not supported for migration of federated partitions. Federated partitions must be recreated manually on the target.

Rollback from a patch, to a version that's older than what was used to configure the Essbase instance, is not supported. In this scenario, importing applications from LCM in the Essbase web interface can fail after rollback.

See also: LcmImport: Restore Cube Files.

Load Data

The Load Data job in the Essbase web interface enables you to insert data values into an Essbase cube, using a source of data and a rule file. The source of data can be a file, a SQL Source, or a defined Datasource in Essbase.

To run this job, you must have at least Database Update permission.

Select a workflow:

- Load Data From File
- Load Data From SQL Source
- Load Data From Datasource

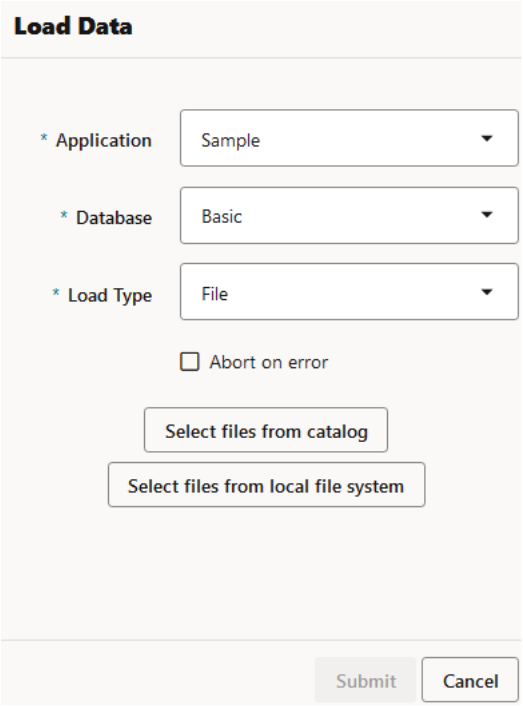
Load Data From File

This procedure describes how to load data using the **File** load type.

To load data from a file:

1. On the Home page, click **Jobs**.
2. From the **New Job** menu, select **Load Data**.
3. Select the application and the database.
4. For the **Load Type**, select **File**.

Copyright © 2019, 2026, Oracle and/or its affiliates.



5. Click **Select Files From Catalog**.
6. Navigate to the source data file and click **Select**.
7. If you are using a load rule, select the **Add file** option next to **Rule file**, browse to the rule file you want to use for that data file, select it, and click **Select**.
8. Select **Abort on Error** if you want to end the data load if it encounters an error. If Abort on Error is not selected, errors are written to an error file (err_*dbname*_*jobid*.txt) in the cube directory.
9. Click **Submit**.
10. To check the job status, click on the **Actions** menu to the right of the job and select **Job Details**. If you are performing a parallel data load (loading more than one data file), there is information in **Job Details** about each individual data load.

Load Data From SQL Source

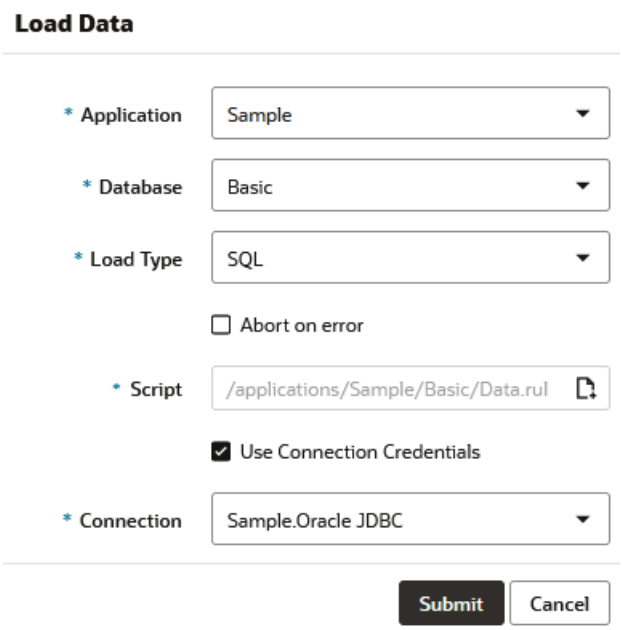
This procedure describes how to load data using the **SQL** load type. Use this type if the load rule itself queries an external source of data. To learn how to set up rules to access external sources of data, refer to Define Rules that Query External Sources.

1. On the Home page, click **Jobs**.
2. From the **New Job** menu, select **Load Data**.
3. Select the application and the database.
4. For the **Load Type**, select **SQL**.
5. For **Script**, browse the catalog and select the rule file.
6. Do one of the following:
   - If your load rule's connectivity to the external database is based on configured ODBC drivers or a connection string, enter the **User Name** and **Password** of a user authorized to access the external database.

Copyright © 2019, 2026, Oracle and/or its affiliates.

- If your load rule's connectivity to the external database is based on a global or application-level saved in Essbase, click **Use Connection Credentials** and select the named connection.

  Application-level connections are prefixed with the application name; for example, **SAMPLE.OracleDB**.



  Refer to Create a Global Connection and Datasource or Create an Application-Level Connection and Datasource.

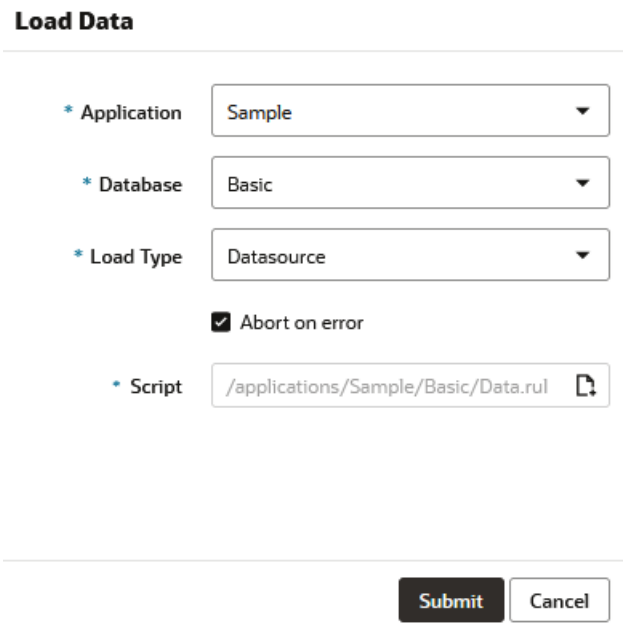
7. Select **Abort on Error** if you want to end the data load if it encounters an error. If Abort on Error is not selected, errors are written to an error file (err_*dbname_jobid*.txt) in the cube directory.
8. Click **Submit**.
9. To check the job status, click on the **Actions** menu to the right of the job and select **Job Details**. If you are performing a parallel data load (loading more than one data file), there is information in **Job Details** about each individual data load.

Load Data From Datasource

This procedure describes how to load data using the **Datasource** load type. This procedure assumes that your load rule's SQL Properties are pointing to a Datasource defined in Essbase, as illustrated in Access External Data Using a Connection and Datasource.

1. On the Home page, click **Jobs**.
2. From the **New Job** menu, select **Load Data**.
3. Select the application and the database.
4. For the **Load Type**, select **Datasource**.
5. For **Script**, browse the catalog and select the rule file.

Copyright © 2019, 2026, Oracle and/or its affiliates.



6. Select **Abort on Error** if you want to end the data load if it encounters an error. If Abort on Error is not selected, errors are written to an error file (err_*dbname*_*jobid*.txt) in the cube directory.
7. Click **Submit**.
8. To check the job status, click on the **Actions** menu to the right of the job and select **Job Details**. If you are performing a parallel data load (loading more than one data file), there is information in **Job Details** about each individual data load.

See Also

Parallel Data Load

Run MDX

The Run MDX job in the Essbase web interface enables you to run an MDX script. MDX is a query language for multidimensional databases that can be used to analyze and extract Essbase data and metadata, define formulas on aggregate storage cubes, and more.

Running MDX scripts requires at least Database Access permission.

To run an MDX script as a job from the Essbase web interface,

1. On the Home page, click **Jobs**.
2. From the **New Job** menu, select **Run MDX**.
3. For **Application**, choose an application.
4. For **Database**, choose a cube.
5. Select an MDX script.
6. Click **Submit**.

See Run MDX Scripts.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Vectorize Outline

The Run **Vectorize Outline** job in the Essbase web interface enables you to vectorize your database outline. This process converts outline elements into vector representations for use with AI features.

> **Note**
>
> If you update the vector embedding model in the AI connection or make any structural changes to the outline, you should rerun the **Vectorize Outline** job to ensure all changes are applied.

To run the **Vectorize Outline** job:

1. In the Essbase web interface, click **Jobs**.
2. From the **New Job** menu, select **Vectorize Outline**.
3. For **Application**, choose an application.
4. For **Database**, choose a cube.
5. Click **Submit** to run the job.

Copyright © 2019, 2026, Oracle and/or its affiliates.

9

Work with Connections and Datasources

Many operations call for connecting to source data external to the cube. Connections and Datasources, which you create and save as reusable objects in Oracle Essbase, provide a way to do this efficiently.

For example, you can set up a partition between a cube and RDBMS tables, share data between a cube and Oracle Database, develop security filters using variables to fetch members or user names from outside source data, and load data from REST API endpoints.

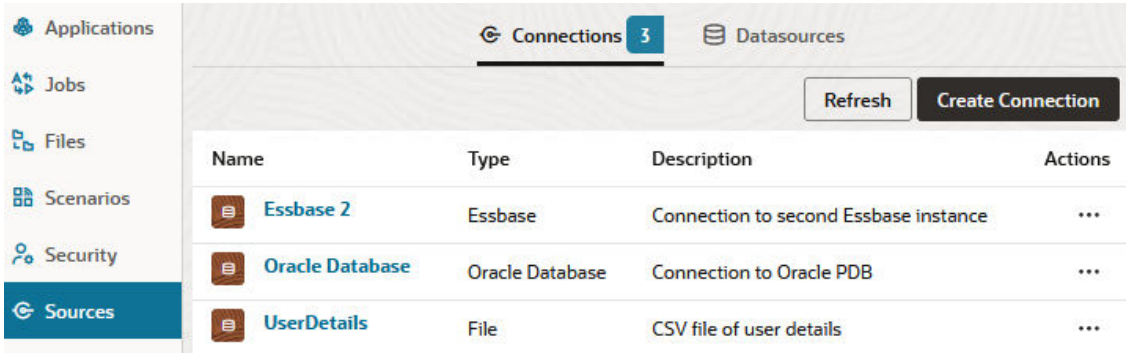
Many cube operations require connection information, such as login details, to access remote source data or hosts. You can define these as connections and Datasources once, and reuse them in various operations, so that you do not have to specify the details each time you perform a task.

You can implement saved connections and Datasources either globally or per application. These abstractions facilitate the following operations:

- Loading dimensions and data
- Importing cubes
- Defining variable security filters
- Connecting cubes using partitions, and accessing real-time data
- Drilling through to remote sources of data

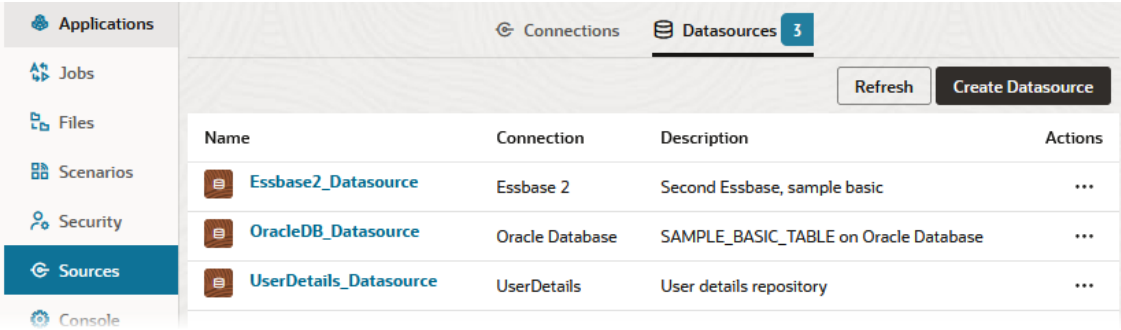
If you have network connectivity between an external source of data and Essbase, you can define connections and Datasources in Essbase to easily "pull" data from the external source. If you have no network connectivity between Essbase and the external source of data, then you should stream data loads or dimension builds using the CLI tool, first creating a local connection, and then issuing the dataload or dimbuild command with the stream option.

A **connection** stores information about an external server and the login credentials that are required to access it. By defining one connection that can be used by multiple processes and artifacts, you can simplify many aspects of your analytics. For example, when it’s time to change a system password, you only need to update one connection.



A **Datasource** is another object that you can define once and reuse, to help you manage data flow into and out of your cubes. You can define a Datasource to represent any external source of data, whether a relational system, a table, a file, or another cube.

Copyright © 2019, 2026, Oracle and/or its affiliates.



You can define one connection and use it to access multiple Datasources. For example, consider an external Oracle Database server that has separate tables for products, resellers, and sales territories. You need only one connection to access Oracle Database, but you might want to create unique Datasources to access each of the tables.

One use case in which you might define multiple Datasources per connection is as follows: if you use separate load rules to build each dimension in a cube, each rules file can be set up to access the relevant table in Oracle Database. For example, assume your cube has a Market dimension, and you regularly build dimensions using a Dim_Market load rule to populate the Market dimension from a SALES_TERRITORIES table. Likewise, you use a Dim_Product load rule to populate the Product dimension from a PRODUCT table. Both load rules can use the same connection, but because they draw from separate tables, you defined two different Datasources.

Historically, you needed to hard code connection and source data details into Essbase artifacts such as rule files, location aliases, and partitions. While hard coded information is still supported in these artifacts, you can work more efficiently if you define connections and Datasources globally (or, at the application level).

Create an Application-Level Connection and Datasource

When Essbase operations for a single application require access to to source data external to the cube, you can implement saved connections and Datasources for that application.

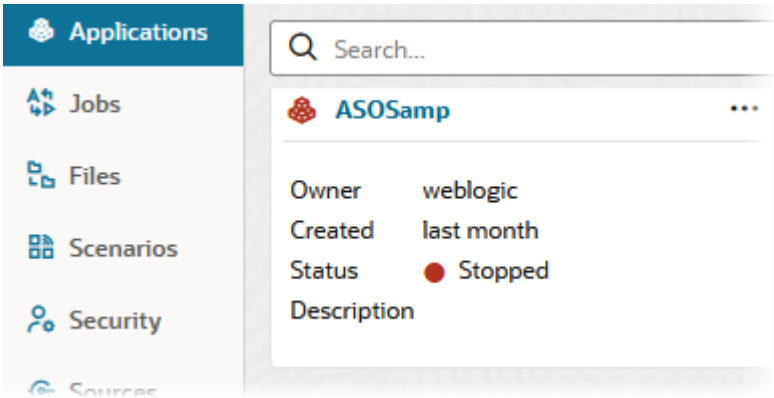
Before you can create connections to external source data from Essbase, you must get the connection details such as host names, user names, passwords, and any other service credentials from your Service Administrator.

- Redwood
- Classic

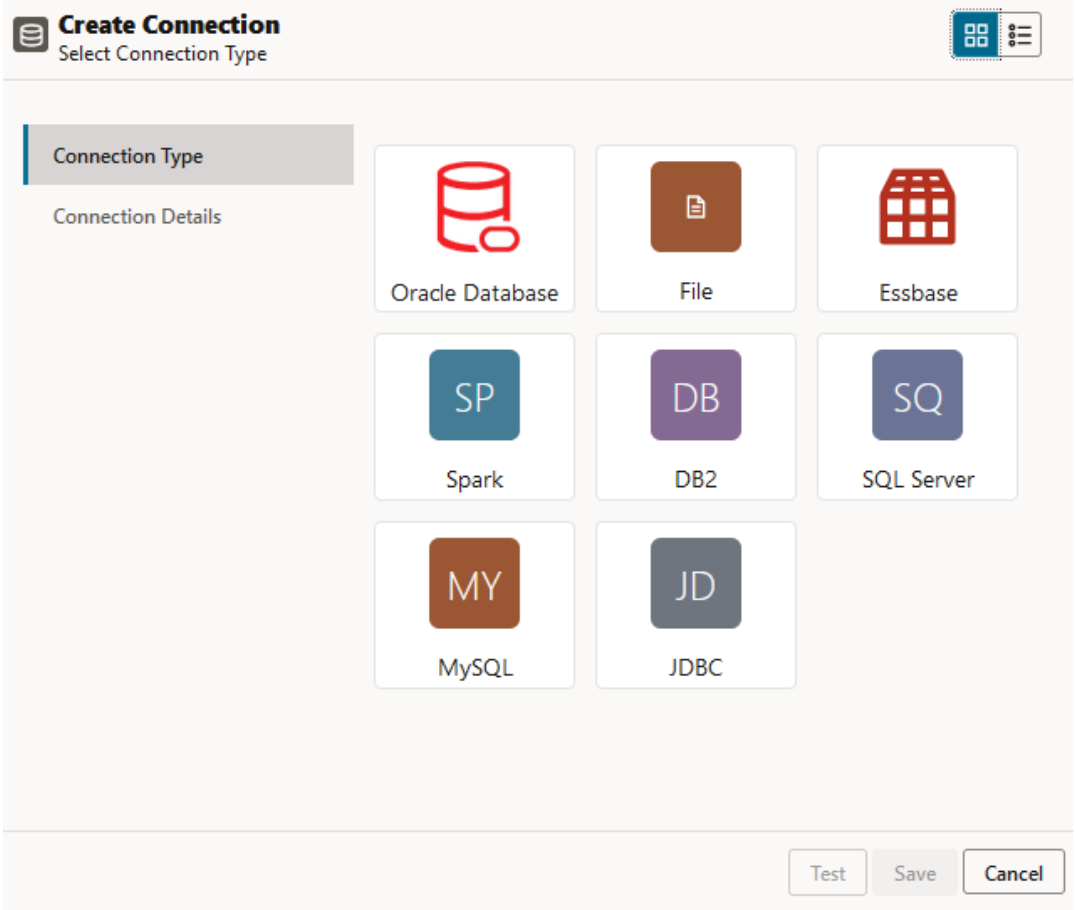
Redwood

1. Log in to the Essbase web interface as an Application Manager, or as a power user with application management permission to the specified application.
2. On the Home page, click an application name. For example, click **ASOSamp**.

Copyright © 2019, 2026, Oracle and/or its affiliates.

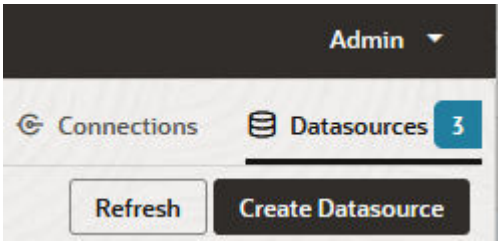


3. Click **Sources**, and then click **Create Connection**.
4. Select the type of source you need to connect to. Sources and versions included with Essbase are listed in the Database section of the certification matrix (see the **Platform SQL** table). If you want to use your own preferred JDBC driver that you will upload, refer for details to Create Connections and Datasources for Generic JDBC Drivers.



5. Complete the connection details and save the connection. The input details vary depending on the source type.
6. To create one or more Datasources that use the connection, click **Datasources**, and then click **Create Datasource**.

Copyright © 2019, 2026, Oracle and/or its affiliates.



The information you provide in steps above will vary depending on the type of source you are using. For some source-specific workflows, refer to the following topics:

- Create a Connection and Datasource to Access Oracle Database
- Create a Connection and Datasource to Access Another Cube
- Create a Connection and Datasource for a File
- Create Connections and Datasources for Generic JDBC Drivers

Classic

1. Log in to the Essbase web interface as an application manager, or as a power user with application management permission to the specified application.
2. On the **Applications** page, click the Actions menu to the right of the application name, and click **Inspect**.
3. Click the **Sources** tab.

4. Click **Create Connection** and select the source you need to connect to. Supported sources and versions included with Essbase are listed in the Database section of the certification matrix (see the **Platform SQL** table). If you want to use your own preferred JDBC driver that you will upload, refer for details to Create Connections and Datasources for Generic JDBC Drivers.
5. Complete the connection details and save the connection. Similarly, create one or more Datasources that use the connection. The input details vary depending on the source type.

Create a Global Connection and Datasource

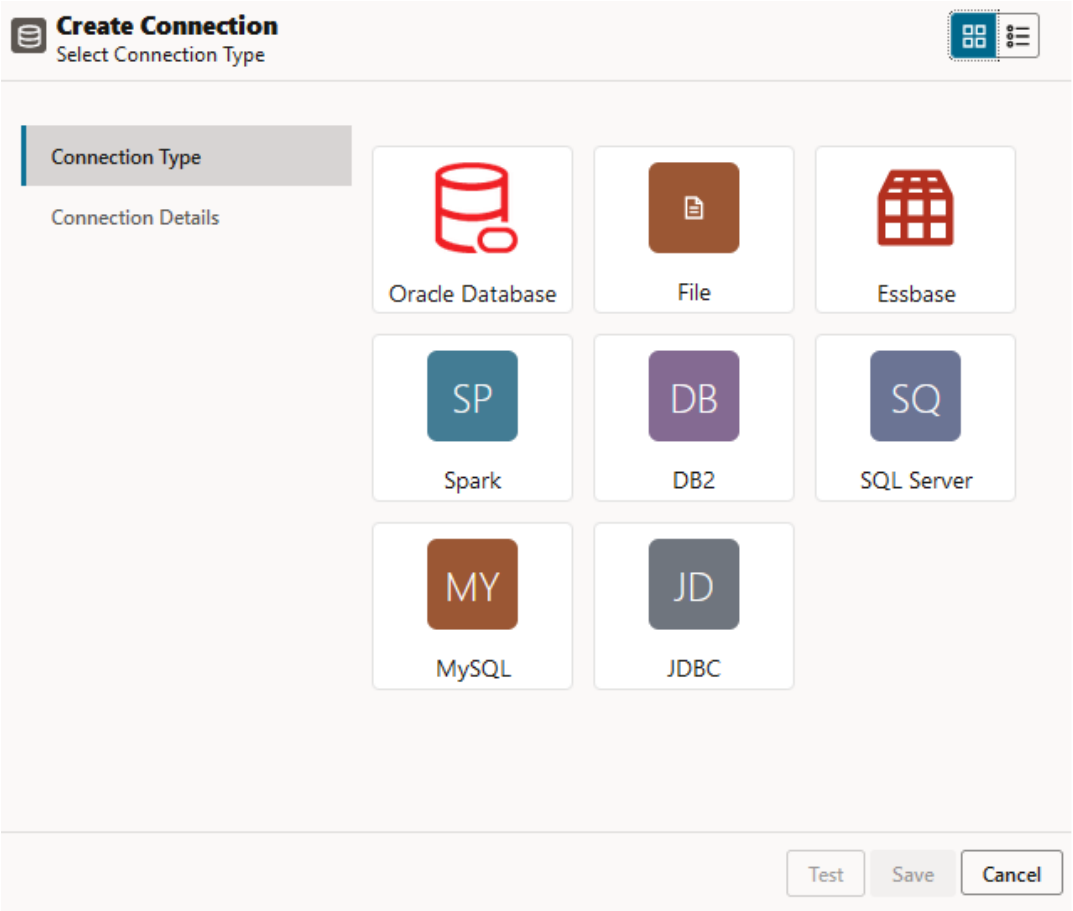
When Essbase operations for more than one application require access to to source data external to the cube, you can implement saved connections and Datasources globally. Global connection and Datasources are accessible to multiple applications. You must be a Service Administrator to create a global connection or Datasource.

- Redwood
- Classic

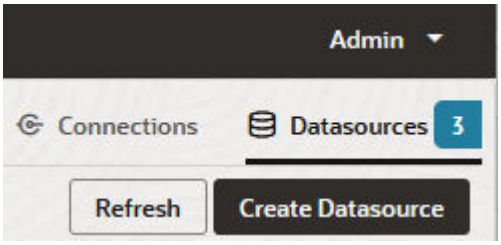
Copyright © 2019, 2026, Oracle and/or its affiliates.

Redwood

1. Log in to the Essbase web interface as a Service Administrator.
2. Click **Sources**.
3. Click **Create Connection** and select the type of source you need to connect to. Sources and versions included with Essbase are listed in the Database section of the certification matrix (see the **Platform SQL** table). If you want to use your own preferred JDBC driver that you will upload, refer for details to Create Connections and Datasources for Generic JDBC Drivers.



4. Complete the connection details. The input details vary depending on the source type. Optionally test, and then create, the connection.
5. To create one or more Datasources that use the connection, click **Datasources**, and then click **Create Datasource**.



Copyright © 2019, 2026, Oracle and/or its affiliates.

The information you provide in steps above will vary depending on the type of source you are using. For some source-specific workflows, refer to the following topics:

- Create a Connection and Datasource to Access Oracle Database
- Create a Connection and Datasource to Access Another Cube
- Create a Connection and Datasource for a File
- Create Connections and Datasources for Generic JDBC Drivers

Classic

1. Log in to the Essbase web interface as a Service Administrator.
2. Click **Sources**.

3. Click **Create Connection** and select the source you need to connect to. Sources and versions included with Essbase are listed in the Database section of the certification matrix (see the **Platform SQL** table). If you want to use your own preferred JDBC driver that you will upload, refer for details to Create Connections and Datasources for Generic JDBC Drivers.

4. Complete the connection details. The input details vary depending on the source type. Optionally test, and then create, the connection. Similarly, create one or more Datasources that use the connection.

Create a Connection and Datasource for a File

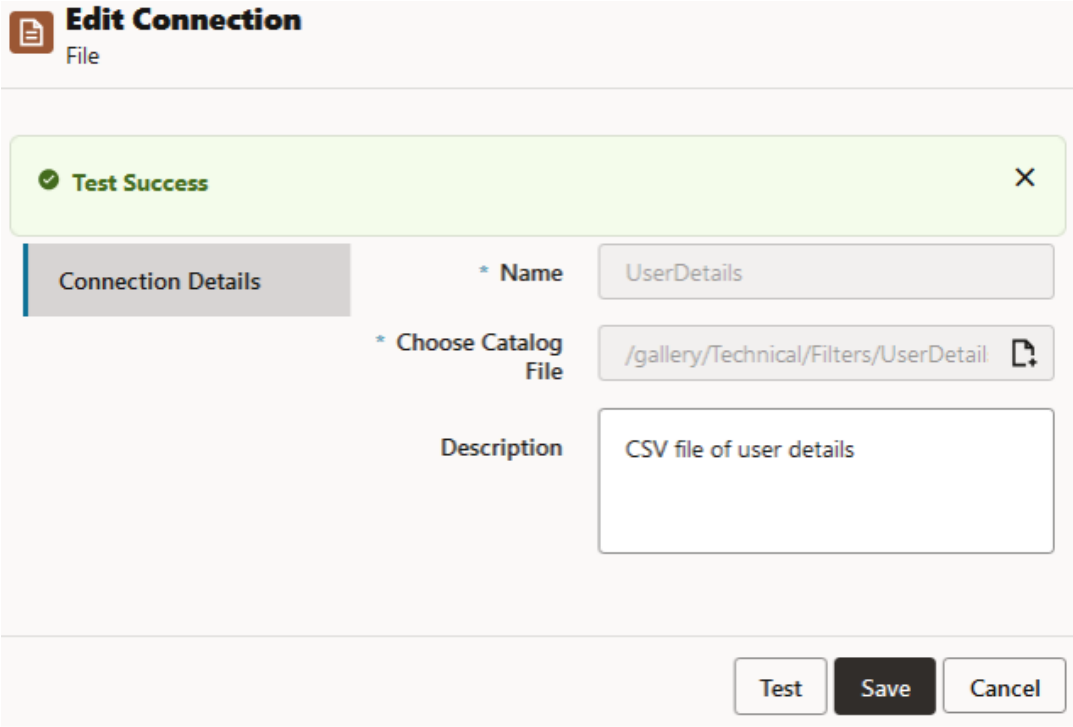
Define a global or local connection and Datasource over a source data file in the Essbase Server file catalog.

1. Upload the source data file to the file catalog on Essbase.
2. If you need a sample source data file for this task flow, you can use `UserDetails.csv` from the gallery section of the file catalog. It is a repository of 22 users, with their associated countries, cost centers, currency, managers, company, business units, and offices.
3. In the Essbase web interface, on the Home page, click **Sources**, then **Connections**.

   Or, to define the connection and Datasource at application level instead of globally, start on the Applications page instead of the Sources page. Click an application name, and then click **Sources**. The example used in this topic is an application level connection defined on Sample.
4. Click **Create Connection**, and for the connection type, select **File**.
5. Enter a name for the connection; for example, `UserDetails`.

Copyright © 2019, 2026, Oracle and/or its affiliates.

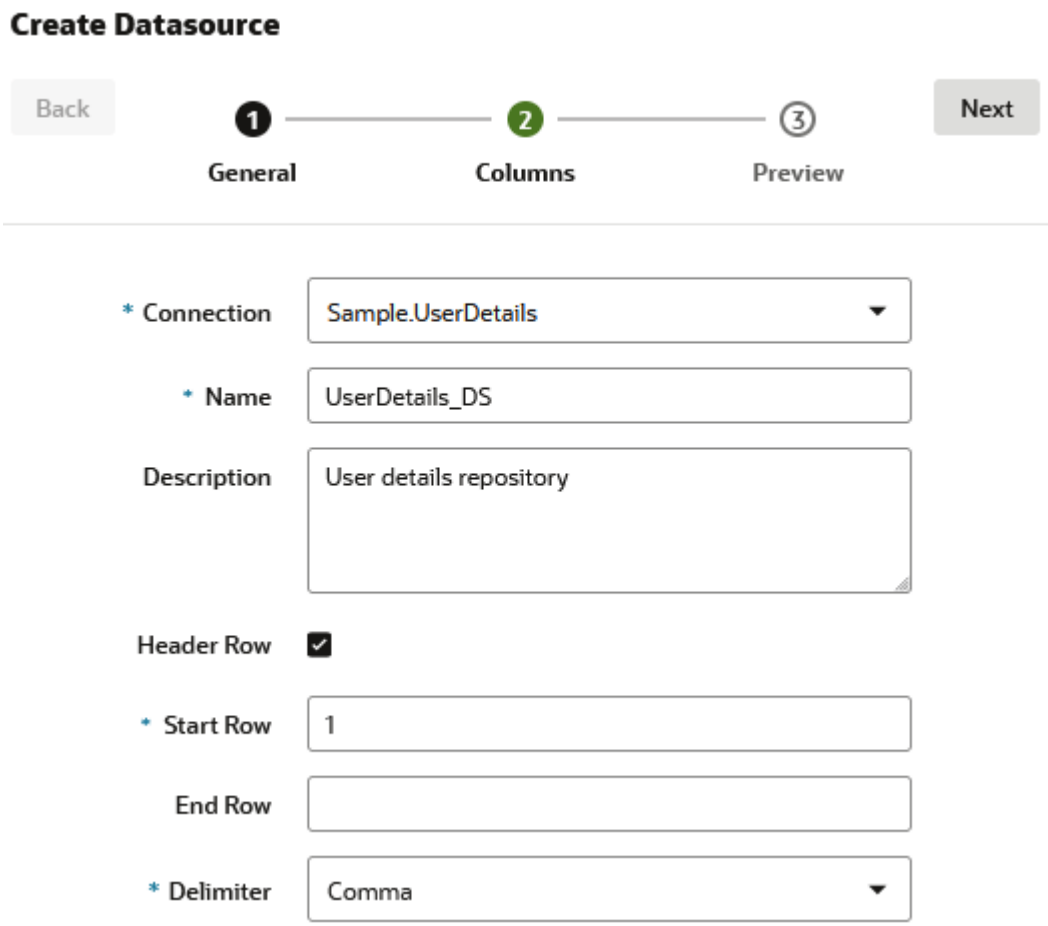
6. Browse to the catalog location of the source data file.
7. Enter an optional description; for example, `CSV file of user details`
8. Click **Test** to validate the connection, and if successful, click **Save**.



9. Verify that the connection was created successfully and appears in the list of connections.

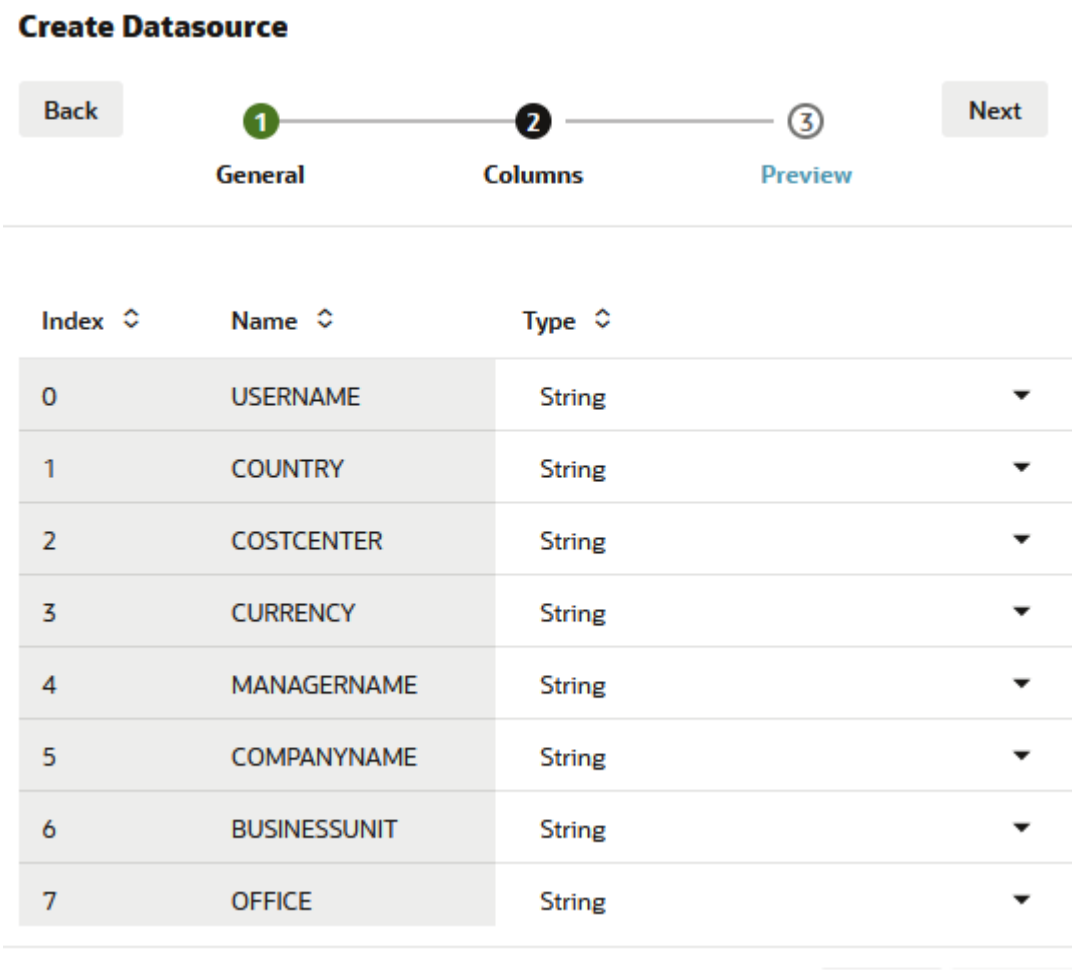
   Next, you will create a Datasource for the file connection.
10. Click **Datasources**, and click **Create Datasource**.
11. From the **Connection** drop-down box, select the name of the connection you just created; for example, UserDetails.
12. Enter a name for the Datasource, and an optional description.
13. Essbase detects and enters details about the source data; for example, whether it has a header row, and is comma-delimited. Click **Next**.

Copyright © 2019, 2026, Oracle and/or its affiliates.



14. If the SQL statement was correct to query a table, the queried columns are populated. Change any numeric columns to Double, and click **Next**.

Copyright © 2019, 2026, Oracle and/or its affiliates.



15. If the preview looks correct, click **Save** to finish creating the Datasource.

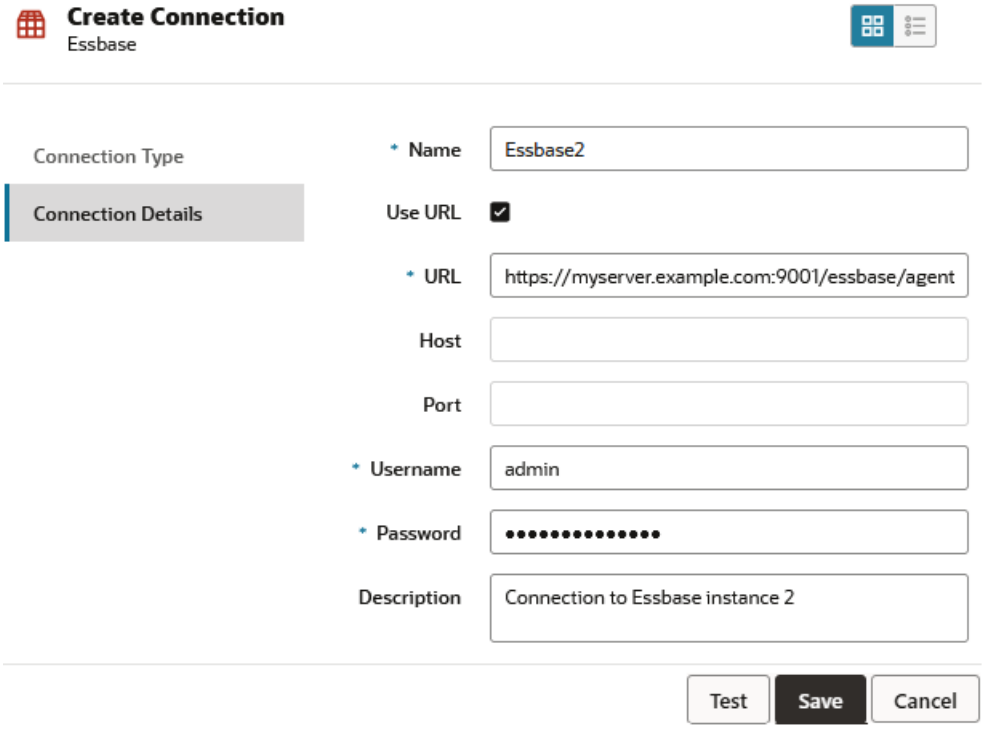
Create a Connection and Datasource to Access Another Cube

Define a connection and Datasource between two Essbase cubes (on different instances).

1. Log in to the Essbase web interface as a service administrator or an application manager.
2. Click **Sources**, then **Connections**.

   To define the connection and Datasource at application level instead of globally, start on the Applications page instead of the Sources page. Click an application name, and then click **Sources**.
3. Click **Create Connection**, and for the connection type, select **Essbase**.
4. Enter a connection name; for example, `Essbase2`.
5. Either enter the Host and Port information, or check the box to **Use URL**. Connection information can be provided by your Service Administrator.

Copyright © 2019, 2026, Oracle and/or its affiliates.



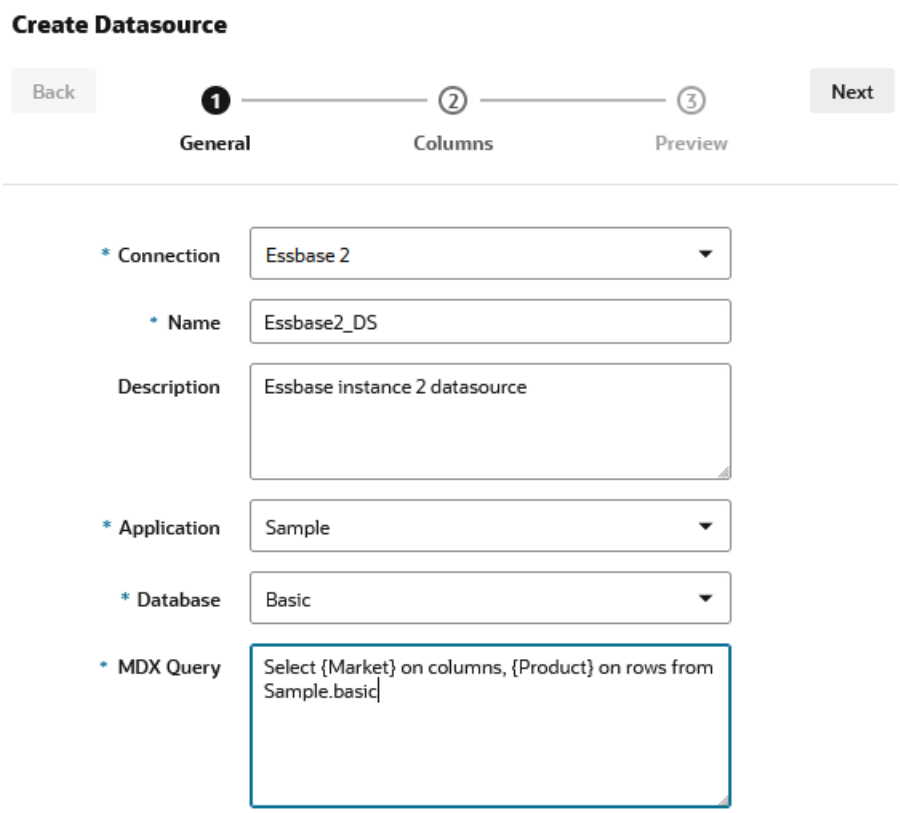
If you are using the URL, use the discovery URL format. A discovery URL is the URL provided by your Service Administrator, with `/agent` appended to the end. For example:

```
https://myserver.example.com:9001/essbase/agent
```

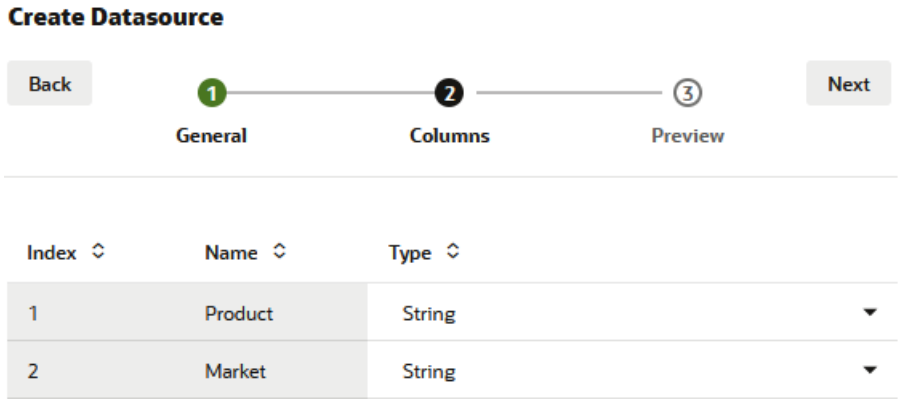
6. Click **Test** to validate the connection, and if successful, click **Save**.
7. Verify that the connection was created successfully and appears in the list of connections.

   Next, you will create a Datasource for the Essbase connection.
8. Click **Datasources**, and click **Create Datasource**.
9. From the **Connection** drop-down box, select the name of the connection you just created.
10. Enter a name for the Datasource, and an optional description.
11. Select the application and database that will be used for this Datasource.
12. Provide a valid MDX query that selects the cube data you want to make available in this Datasource.

Copyright © 2019, 2026, Oracle and/or its affiliates.

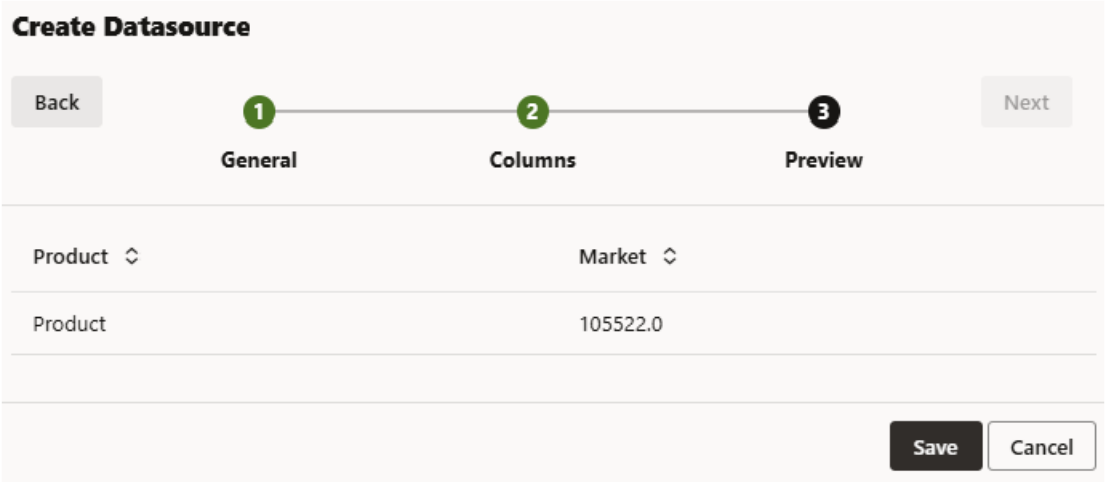


13. Click **Next**. If the MDX syntax was correct to query the remote cube, the queried columns are populated.
14. Change any numeric columns to Double, and click **Next**.



15. Change any additional source-specific parameters, if applicable, and click **Next**.
16. Review the preview panel. The results of the MDX query fetch columns of data from the other cube.

Copyright © 2019, 2026, Oracle and/or its affiliates.



17. If the preview is correct, click **Save** to finish creating the Datasource.

Create a Connection and Datasource to Access Oracle Database

Define a connection and Datasource between Essbase and Oracle Database.

If applicable, use one of the following subtopics instead of this one:

- Create a Connection and Datasource for Autonomous Data Warehouse
- Create a Connection for Federated Cubes

1. Log in to the Essbase web interface as a service administrator or an application manager.
2. Click **Sources**, then **Connections**.

   To define the connection and Datasource at application level instead of globally, start on the Applications page instead of the Sources page. Click an application name, and then click **Sources**.

3. Click **Create Connection**, and for the connection type, select **Oracle Database**.
4. Click **Create Connection** and select **Oracle Database**.
5. Enter a connection name, host, port number, user name, and password. When you enter the **User** name, enter the Oracle Database user name, without the role. Select **SID** (server ID) or **Service**, and enter server details.

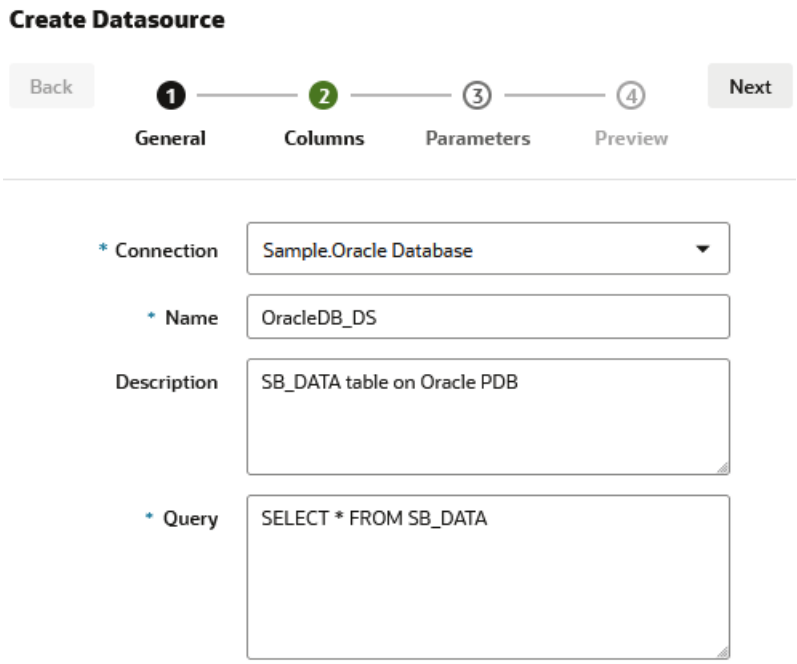
6. Click **Test** to validate the connection, and if successful, click **Save**.
7. Verify that the connection was created successfully and appears in the list of connections.

   Next, you will create a Datasource for the Oracle Database connection.

8. Click **Datasources**, and click **Create Datasource**.
9. From the **Connection** drop-down box, select the name of the connection you just created; for example, Sample.Oracle Database. Application-level connections are prefixed with the application name, in the format *appName.connectionName*.
10. Provide a name for the Datasource; for example, OracleDB_DS.

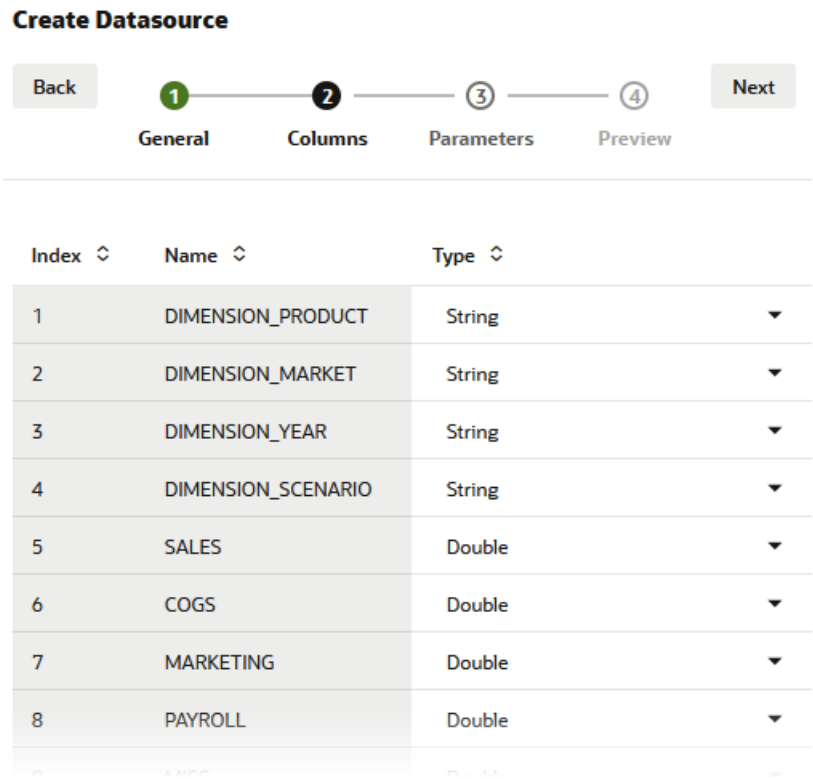
Copyright © 2019, 2026, Oracle and/or its affiliates.

11. Optionally enter a description of the Datasource; for example, `SAMPLE_BASIC_TABLE on Oracle Database`.
12. In the Query field, provide the appropriate SQL query that selects the Oracle Database data you want to make available in this Datasource.



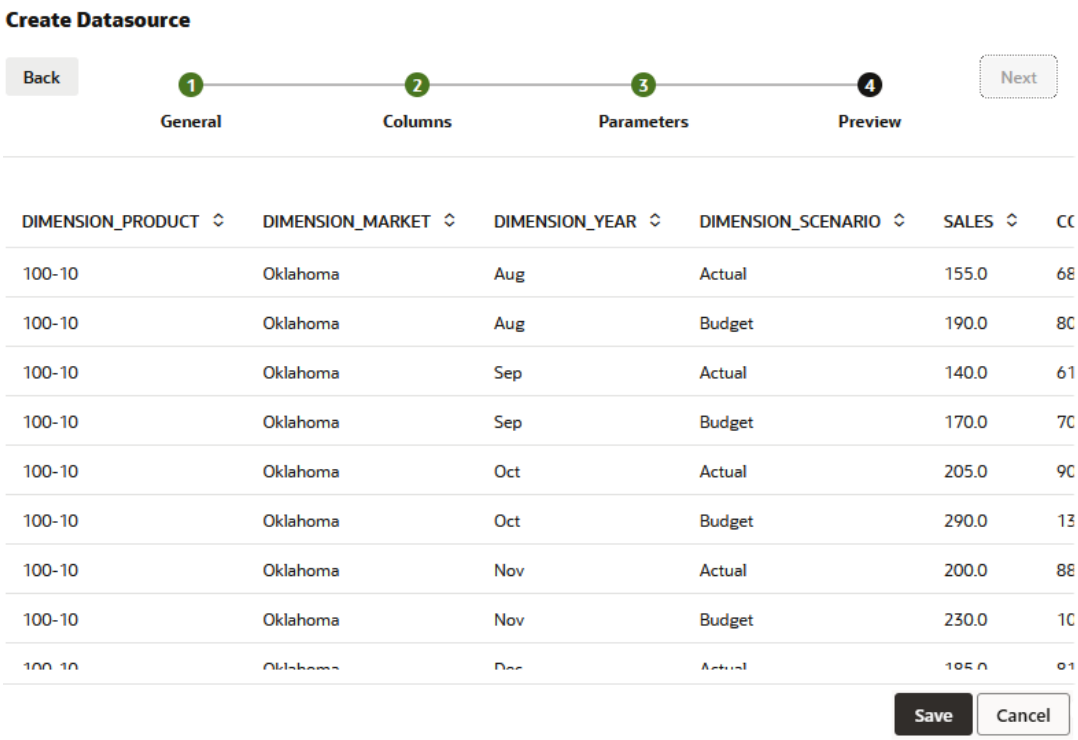
13. Click **Next**. If the SQL statement was correct to query an Oracle Database area, The preview of the data source should display up to 10 records of data.

Copyright © 2019, 2026, Oracle and/or its affiliates.



14. Change any numeric columns to Double, and click **Next**.
15. Change any additional source-specific parameters, if applicable, and click **Next**. For information about parameter use, see Implement Parameters for Datasources.
16. Review the preview panel. You should see the results of the SQL query fetching columns of data from Oracle Database.

Copyright © 2019, 2026, Oracle and/or its affiliates.



17. If the preview looks correct, click **Save** to finish creating the Datasource.

Create a Connection and Datasource for Autonomous Data Warehouse

Define a connection and Datasource between Essbase and Autonomous AI Lakehouse.

If you will create a federated partition between Essbase and Autonomous AI Lakehouse Serverless, use the following topic instead of this one: Create a Connection for Federated Cubes.

To create a global connection, you need to have the service administrator role. To create an application level connection, you need to have user role, plus application manager permission on the application.

- Redwood
- Classic

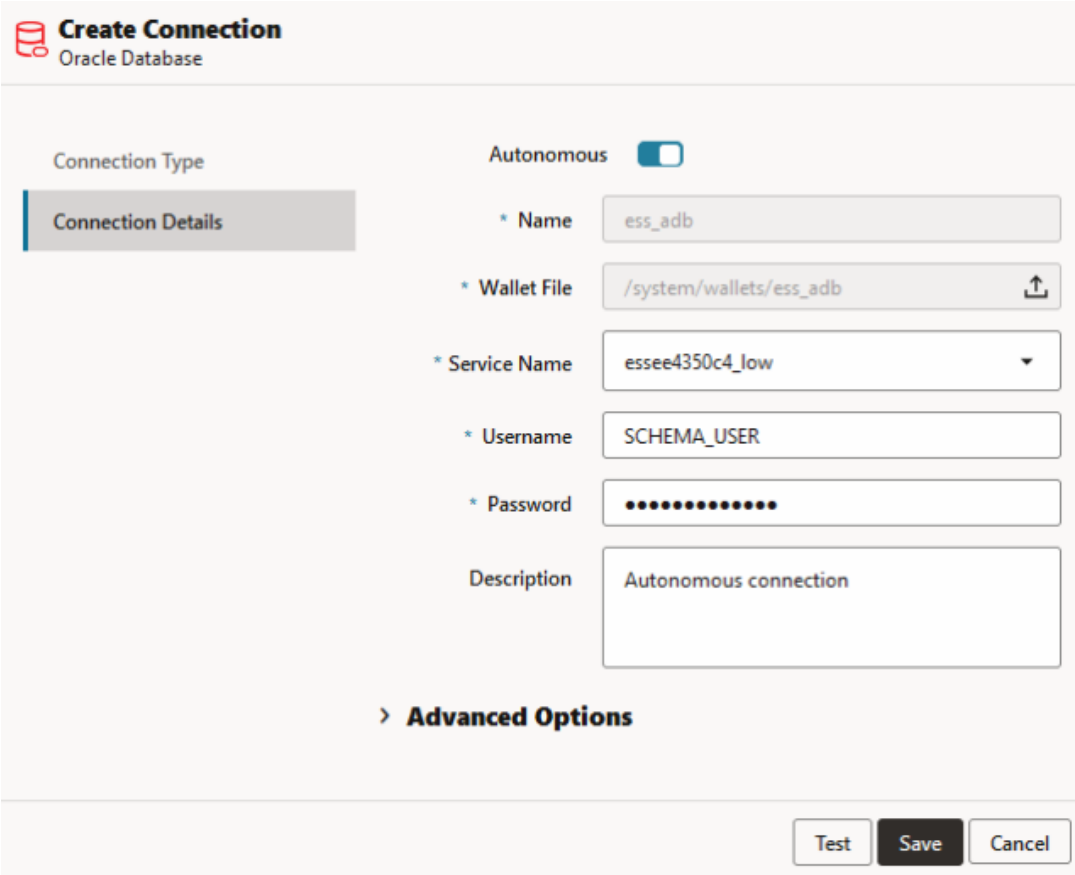
Redwood

1. Log in to the Essbase web interface as a service administrator.
2. Click **Sources**.

Copyright © 2019, 2026, Oracle and/or its affiliates.

To define the connection and Datasource at application level, instead of globally, start on the Applications page instead of the Sources page. Click an application name, and then click **Sources**.

3. Click **Create Connection** and select **Oracle Database**.
4. Select **Autonomous** using the toggle switch.



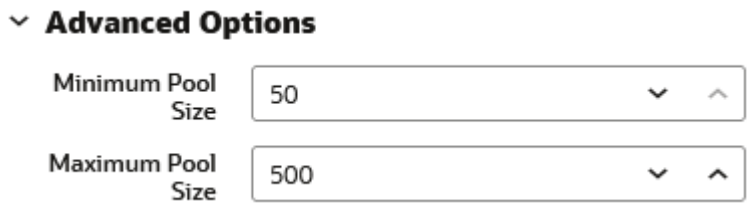
5. Enter a connection name.
6. If needed, drag and drop a wallet file, or click the **Wallet File** field to upload one.

   If you are using a connection which has already been made available to you (a repository connection), you do not need to upload a wallet, because it should already be in the repository. Select the **Repository Database** option.

   If you need to upload a wallet, obtain a wallet file by selecting **Download Client Credentials (Wallet)** from your Autonomous Data Warehouse Administration page in Oracle Cloud Infrastructure.
7. Select a service name.
8. Enter your Autonomous AI Lakehouse username and password, and optionally, a description.
9. Click **Test** to validate the connection, and if successful, click **Save**.

   If you get connection errors, you may need to expand **Advanced Options** to adjust the minimum and maximum connection pool sizes.

Copyright © 2019, 2026, Oracle and/or its affiliates.



Refer to About Controlling the Pool Size in UCP in *Universal Connection Pool Developer's Guide*.

10. Verify that the connection was created successfully and appears in the list of connections.
11. Next, you will create a Datasource for the Autonomous AI Lakehouse connection. Click **Datasources**, and click **Create Datasource**.
12. From the Connection drop-down box, select the name of the connection you just created; for example, essbaseadb_public. For application-level Datasources, select the application-level connection name, in the format *appName.connectionName*.
13. Provide a name for the Datasource; for example, essbaseadb_ds.
14. Optionally enter a description of the Datasource; for example, Autonomous AI Lakehouse Datasource.
15. In the **Query** field, provide the appropriate SQL query that selects the Autonomous AI Lakehouse data you want to make available in this Datasource.
16. Click **Next**. If the SQL statement was correct to query an Autonomous AI Lakehouse area, the queried columns should display with numeric indices, column names, and data types.
17. Change any additional source-specific data types, if applicable, and click **Next**.
18. Review the preview panel. The results of the SQL query should fetch some columns of data from Autonomous AI Lakehouse.
19. If the preview is correct, click **Save** to finish creating the Datasource.

Classic

1. Log in to the Essbase web interface as a service administrator.
2. Click **Sources**.

   To define the connection and Datasource at application level, instead of globally, start on the Applications page instead of the Sources page. From the Actions menu to the right of an application name, launch the inspector and click **Sources**.
3. Click **Create Connection** and select **Oracle Database**.
4. Select **Autonomous** using the toggle switch.
5. Enter a connection name.

Copyright © 2019, 2026, Oracle and/or its affiliates.

6. If needed, drag and drop a wallet file, or click the **Wallet File** field to upload one.

   If you are using a connection which has already been made available to you (a repository connection), you do not need to upload a wallet, because it should already be in the repository. Select the **Repository Database** option.

   If you need to upload a wallet, obtain a wallet file by selecting **Download Client Credentials (Wallet)** from your Autonomous Data Warehouse Administration page in Oracle Cloud Infrastructure.

7. Select a service name.
8. Enter your Autonomous AI Lakehouse username and password, and optionally, a description.
9. Click **Test** to validate the connection, and if successful, click **Create**.

   If you get connection errors, you may need to expand **Advanced Options** to adjust the minimum and maximum connection pool sizes.

   Refer to About Controlling the Pool Size in UCP in *Universal Connection Pool Developer's Guide*.

10. Verify that the connection was created successfully and appears in the list of connections.
11. Next, you will create a Datasource for the Autonomous AI Lakehouse connection. Click **Datasources**, and click **Create Datasource**.
12. From the Connection drop-down box, select the name of the connection you just created; for example, EssbaseADWS. For application-level Datasources, select the application-level connection name, in the format *appName.connectionName*.
13. Provide a name for the Datasource; for example, ADW_DS.
14. Optionally enter a description of the Datasource; for example, Autonomous AI Lakehouse Datasource.
15. In the **Query** field, provide the appropriate SQL query that selects the Autonomous AI Lakehouse data you want to make available in this Datasource.
16. Click **Next**. If the SQL statement was correct to query an Autonomous AI Lakehouse area, you should see the queried columns populated.
17. Change any additional source-specific parameters, if applicable, and click **Next**.
18. Review the preview panel. The results of the SQL query should fetch some columns of data from Autonomous AI Lakehouse.
19. If the preview looks correct, click **Create** to finish creating the Datasource.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Create Connections and Datasources for Generic JDBC Drivers

Use this workflow to enable Essbase to connect to any JDBC source of data, using drivers you upload to the Essbase Server.

If you are the Essbase deployment administrator, you can configure Essbase to use your preferred drivers you upload to the Essbase server machine. Oracle has tested Essbase JDBC connectivity using Oracle drivers. To use JDBC drivers from other vendors, check the driver documentation for requirements on specifying the URL and credentials of your JDBC data source. For any performance-related steps, refer to the vendor JDBC documentation.

> **Note**
>
> Ensure that the JDBC driver you use with Essbase honors the **setFetchSize** method for controlling memory used while processing the result set. For optimal performance of data load and dimension build processes, Essbase fetches 1000 records per network call.

To configure Essbase to use generic JDBC drivers,

1. Connect to the Essbase server machine using SSH.
2. Manually create a `drivers` directory in <Essbase Product Home> on the server instance.

   Ensure `drivers` is all lower case, as the path is case sensitive.
3. From your vendor site, download the JDBC driver JARs you want to use.

   The Oracle Database JDBC driver supported by Essbase is `ojdbc8.jar`.

   If you use Autonomous AI Lakehouse, you need to download the full archive (`ojdbc8-full.tar.gz`) containing the Oracle JDBC Thin driver and companion JARs.
4. Upload the JDBC driver jars to the `drivers` directory on the Essbase instance.

   Upload only one version of each database driver to the `drivers` directory. For example, do not upload both `sqljdbc41.jar` and `sqljdbc42.jar`, or else Essbase will use the older one (as it appears first in CLASSPATH).

   If you use Autonomous AI Lakehouse, extract the archive (`ojdbc8-full.tar.gz`) and move all of the contents directly into the `drivers` directory (not a subfolder).
5. Create connections to the JDBC drivers.
   a. In the Essbase web interface, on the Home page, click **Sources**, then **Connections**.

      Or, to define the connection and Datasource at application level instead of globally, start on the Home page instead of the Sources page, click an application name, and then click **Sources**.
   b. Click **Create Connection** and select **JDBC**.

      To find the JDBC driver, Essbase looks in the `drivers` folder. If no `jar` files are found, Essbase returns a Class Not Found (or failure to load driver) error when you test the connection.
   c. In the **Create Connection** screen,
      i. Provide a name for the JDBC connection. For example, Oracle JDBC.

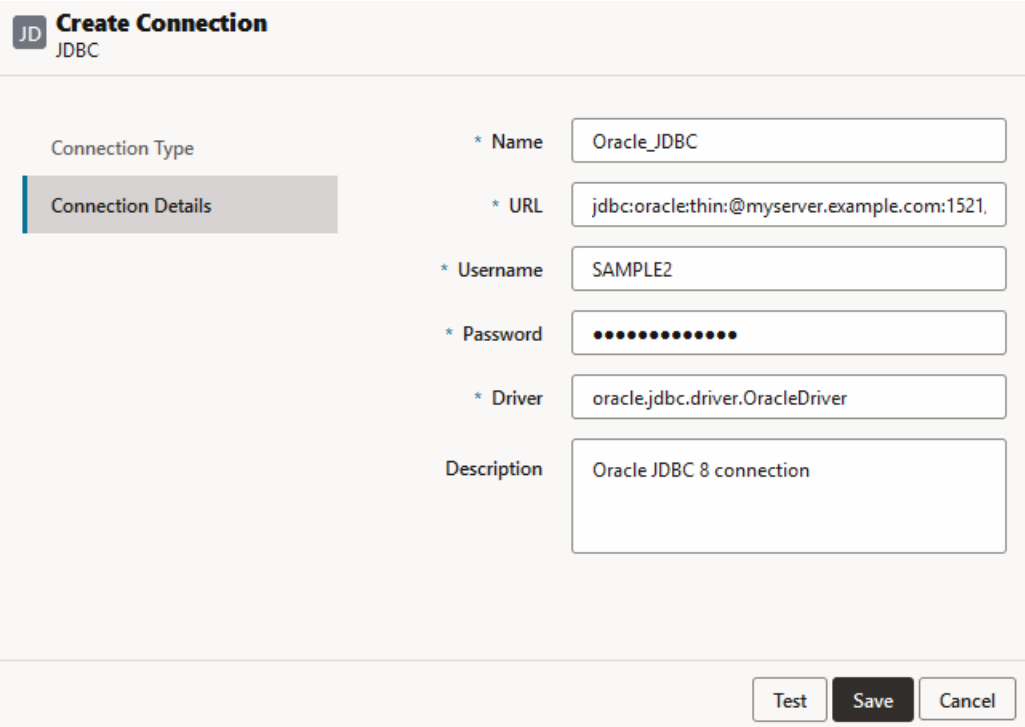
Copyright © 2019, 2026, Oracle and/or its affiliates.

ii. In the URL field, provide the JDBC connection string. For example, `jdbc:oracle:thin:@myserver.example.com:1521/orclpdb.example.com`. Obtain the JDBC connection string from the JDBC provider.

The syntax format above applies only for Oracle Database. See More Connection Examples for Generic JDBC Drivers if you are working with other providers.

iii. For User and Password fields, enter the credentials for a user who is authorized to access the database.

iv. In the Driver field, provide the fully qualified class name of the JDBC driver. For example, `oracle.jdbc.driver.OracleDriver`.



For Oracle drivers, specify the URL using the following syntax guidelines:

- If Oracle Database is registered with a listener, you can use Service Name in the URL instead of the SID, using short syntax `jdbc:oracle:thin:@<host>:<port>/<servicename>`. Example:

  ```
  jdbc:oracle:thin:@myserver.example.com:1521/orclpdb.example.com
  ```

- The following example uses Service Name with long syntax.

  ```
  jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS=(host=myserver.example.com)
  (protocol=tcp)(port=1521))
  (CONNECT_DATA=(SERVICE_NAME=orclpdb.example.com)))
  ```

- To use the Oracle System ID (SID) that uniquely identifies the database, use the syntax `jdbc:oracle:thin:@<host>:<port>:<SID>`. For example,

  ```
  jdbc:oracle:thin:@myhost:1521:orcl
  ```


Copyright © 2019, 2026, Oracle and/or its affiliates.

- If you are using Autonomous AI Lakehouse, in the URL syntax, you must include the TNS_ADMIN environment variable specifying the path to the wallet. The wallet can be anywhere on the Essbase server machine, but you must provide the full path, using syntax like `jdbc:oracle:thin:@database_service_name?TNS_ADMIN=walletpath`.

  Linux Example

  ```
  jdbc:oracle:thin:@adwsql_low?TNS_ADMIN=/scratch/oracle_home/dist/
  essbase/drivers/adwConn
  ```

  Windows Example

  ```
  jdbc:oracle:thin:@adwsql_low?TNS_ADMIN="C:\\Oracle123\\Middleware\
  \Oracle_Home\\essbase\\drivers\\adwConn"
  ```

  Example on an OCI Deployment

  ```
  jdbc:oracle:thin:@adwsql_low?TNS_ADMIN=/u01/data/essbase/catalog/
  users/firstname.lastname@example.com/adwconn
  ```

The examples above work only for Oracle Database. See [More Connection Examples for Generic JDBC Drivers] if you are working with other providers.

d. Click **Test** to validate the connection, and if successful, click **Save**.

e. Verify that the connection was created successfully and appears in the list of connections.

6. Create Datasources over the generic JDBC driver connections.

   a. Click **Datasources**, and click **Create Datasource**.

   b. From the **Connection** drop-down box, select the name of the connection you just created; for example, Oracle JDBC. Application-level connections are prefixed with the application name, in the format *appName.connectionName*.

   c. Provide a name for the Datasource; for example, OracleDB_Datasource.

   d. Optionally enter a description of the Datasource; for example, `SAMPLE_BASIC_TABLE on Oracle Database`.

   e. In the Query field, provide the appropriate SQL query that selects the data you want to make available in this Datasource.

   f. Click **Next**. If the SQL statement was correct to query a table, the queried columns are populated.

   g. Change any numeric columns to Double, and click **Next**.

   h. Change any additional source-specific parameters, if applicable, and click **Next**. For information about parameter use, see [Implement Parameters for Datasources].

   i. Review the preview panel. You should see the results of the query fetching columns of data from the external source.

   j. If the preview looks correct, click **Save** to finish creating the Datasource.

Copyright © 2019, 2026, Oracle and/or its affiliates.

More Connection Examples for Generic JDBC Drivers

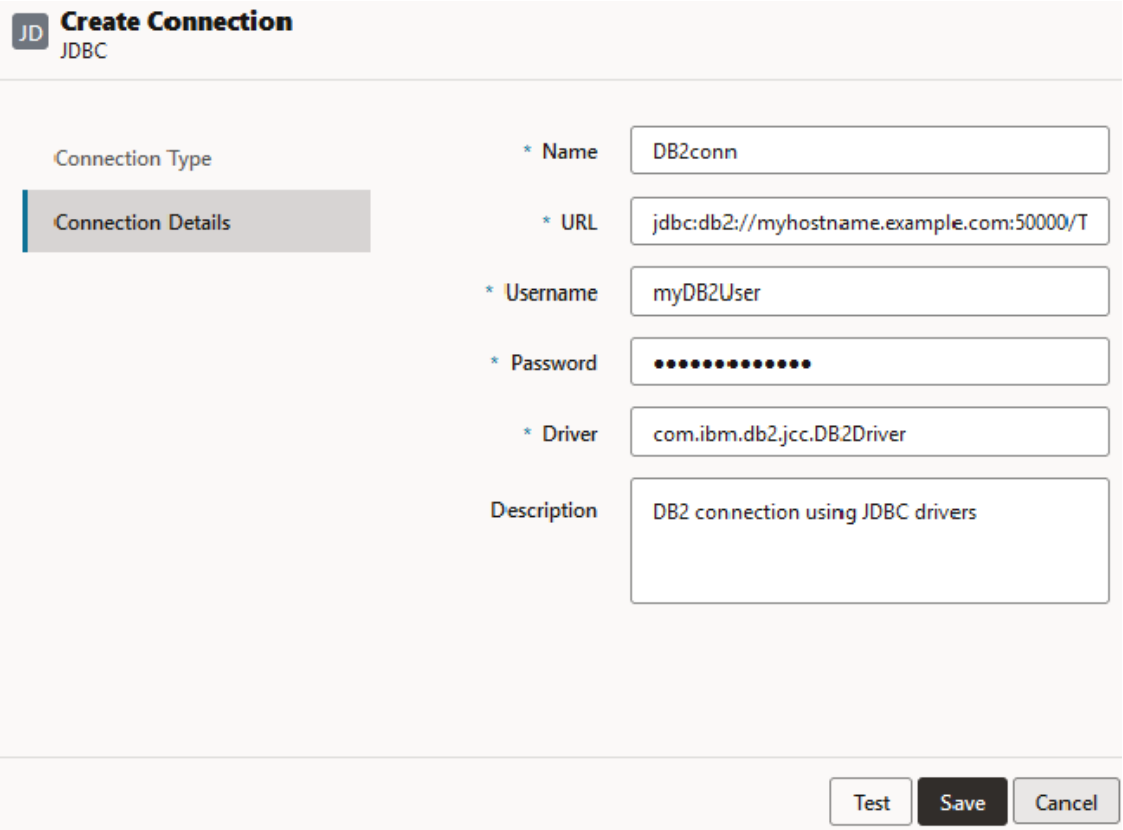
The following examples illustrate using the Essbase web interface to connect to non-Oracle JDBC sources of data, using generic drivers you uploaded to the Essbase Server.

The following examples are for non-Oracle sources. To create an Oracle Database connection using a generic JDBC driver, see Create Connections and Datasources for Generic JDBC Drivers.

Start by creating a global or application level connection of type **JDBC**, using the Essbase web interface. Complete the connection details that apply for the generic JDBC driver you uploaded.

JDBC Connection Example for DB2

For a generic JDBC connection to DB2, in the **Connection Details** screen,

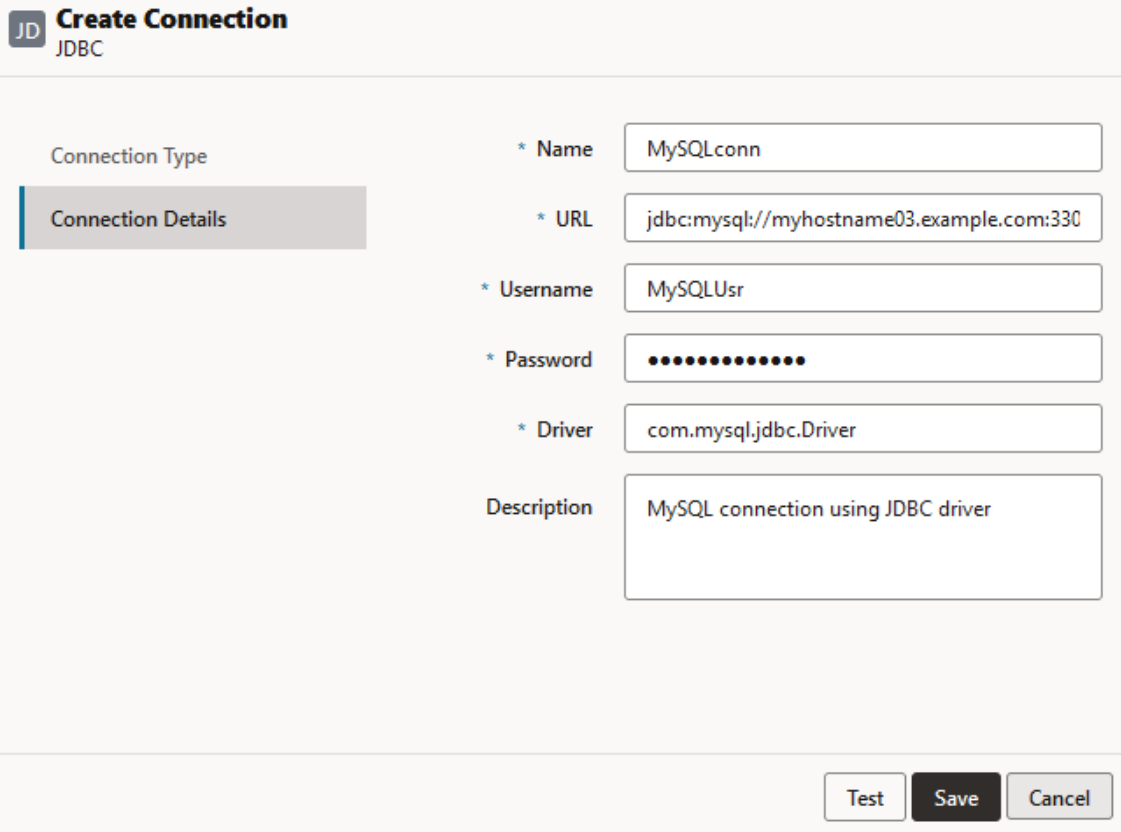


1. Provide a name for the JDBC connection. For example, DB2conn.
2. In the URL field, provide the JDBC connection string. For example, `jdbc:db2://myhostname.example.com:50000/TBC`. Obtain the JDBC connection string from the JDBC provider.
3. For User and Password fields, enter the credentials for a user who is authorized to access the DB2 database.
4. In the Driver field, provide the fully qualified class name of the JDBC driver. For example, `com.ibm.db2.jcc.DB2Driver`.

Copyright © 2019, 2026, Oracle and/or its affiliates.

JDBC Connection Example for MySQL

For a generic JDBC connection to MySQL, in the **Connection Details** screen,

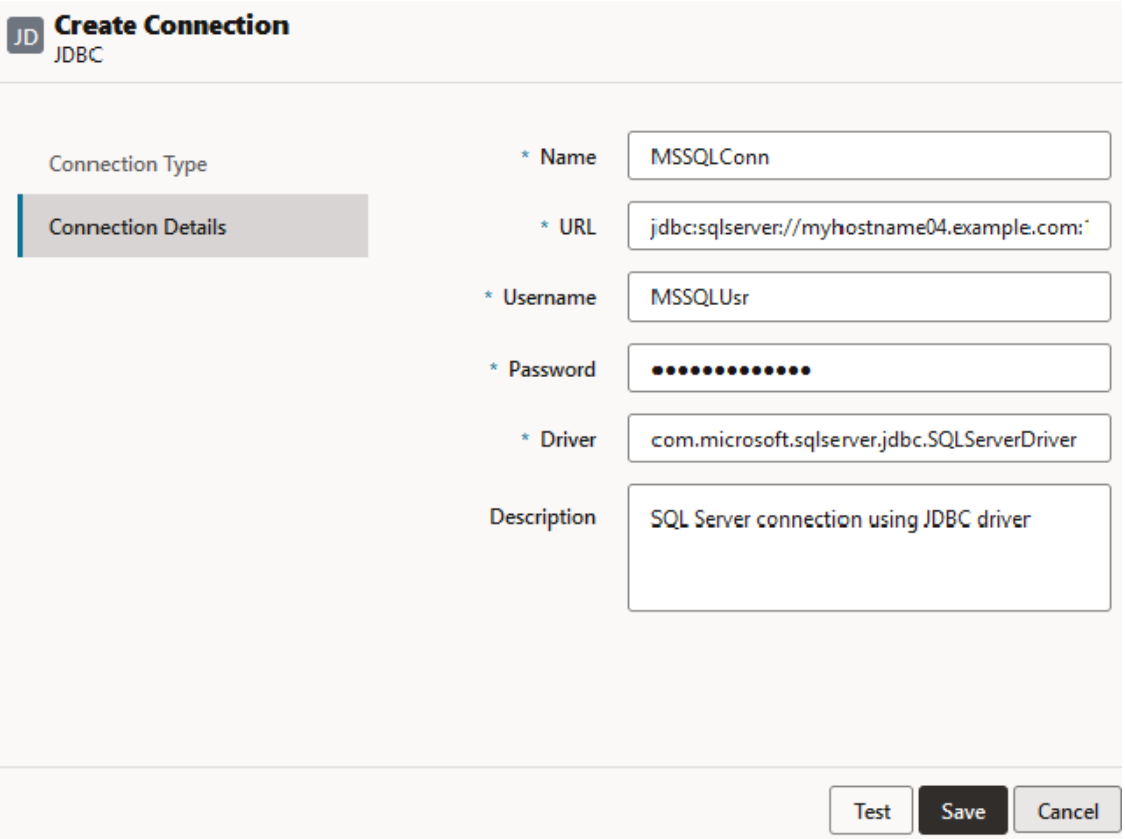


1. Provide a name for the JDBC connection. For example, MySQLconn.
2. In the URL field, provide the JDBC connection string. For example, `jdbc:mysql://myhostname03.example.com:3306/tbc`. Obtain the JDBC connection string from the JDBC provider.
3. For User and Password fields, enter the credentials for a user who is authorized to access the database.
4. In the Driver field, provide the fully qualified class name of the JDBC driver. For example, `com.mysql.jdbc.Driver`.

JDBC Connection Example for SQL Server

For a generic JDBC connection to Microsoft SQL Server, in the **Connection Details** screen,

Copyright © 2019, 2026, Oracle and/or its affiliates.

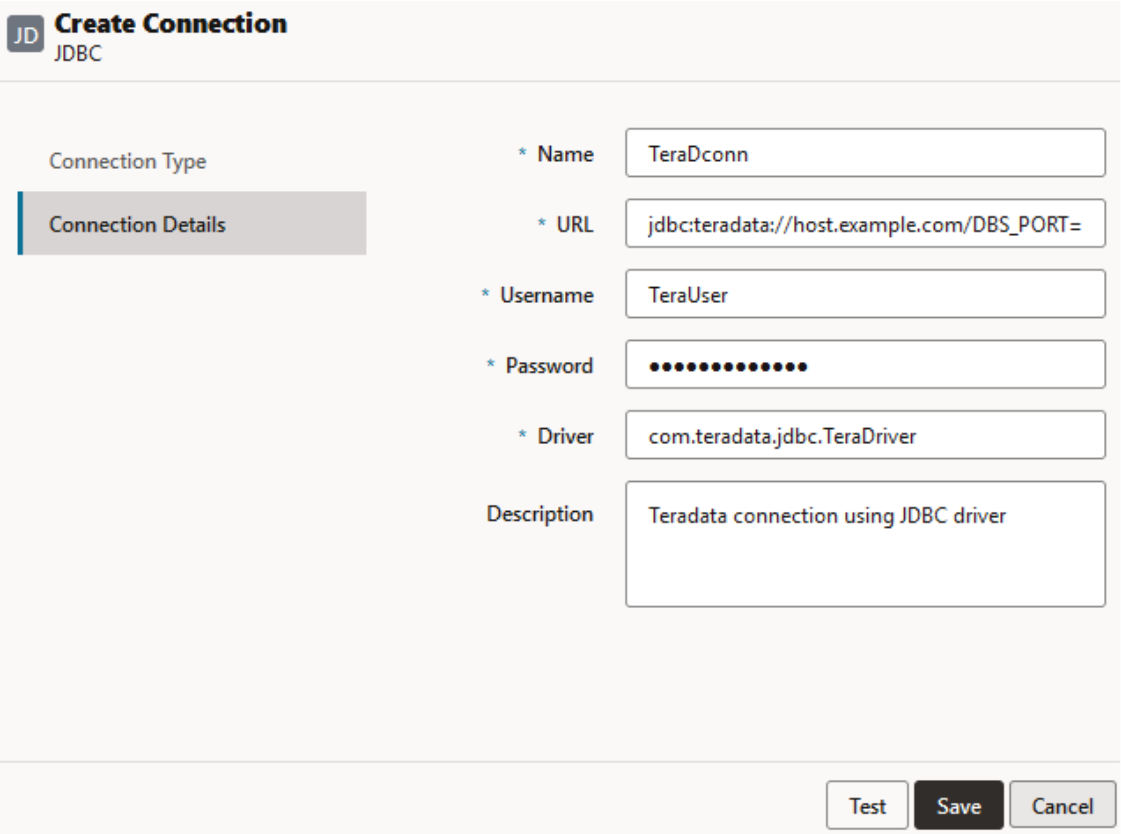


1. Provide a name for the JDBC connection. For example, MSSQLConn.
2. In the URL field, provide the JDBC connection string. For example, `jdbc:sqlserver://myhostname04.example.com:1433`. Obtain the JDBC connection string from the JDBC provider.
3. For User and Password fields, enter the credentials for a user who is authorized to access the database.
4. In the Driver field, provide the fully qualified class name of the JDBC driver. For example, `com.microsoft.sqlserver.jdbc.SQLServerDriver`.

JDBC Connection Example for Teradata

For a generic JDBC connection to Teradata, in the **Connection Details** screen,

Copyright © 2019, 2026, Oracle and/or its affiliates.



1. Provide a name for the JDBC connection. For example, TeraDconn.
2. In the URL field, provide the JDBC connection string. For example, `jdbc:teradata://host.example.com/DBS_PORT=1025`. Obtain the JDBC connection string from the JDBC provider.
3. For User and Password fields, enter the credentials for a user who is authorized to access the database.
4. In the Driver field, provide the fully qualified class name of the JDBC driver. For example, `com.teradata.jdbc.TeraDriver`.

Implement Parameters for Datasources

To make Datasources more flexible, you can implement runtime parameters, enabling Essbase to use variables in the query. Using variables in Datasources helps you streamline operations by requiring fewer Datasources to maintain.

These may be substitution variables defined in Essbase, runtime parameters defined by the grid context when Smart View users drill through to external data, or user defined functions written in an external source system.

Whenever you plan to use variables in Essbase Datasources, you first need to

1. Include variable syntax in the Datasource query. For example, the Datasource query must include ? in its syntax, where the ? is a placeholder for some variable to be defined at runtime.
2. Do one of the following:

Copyright © 2019, 2026, Oracle and/or its affiliates.

- Set a fixed, default parameter value in the Datasource for Essbase to use as a fallback in case the variable has an invalid context at runtime
- Set a substitution variable to be used by the Datasource
- Pass an external, user-defined function (or stored procedure) as a parameter

To define Datasources and implement parameters for them, you must be an Application Manager or higher.

Implementing variables to Datasources enables you to specify a runtime query context that will be applied whenever a user accesses a Datasource associated with an Essbase cube.

For example, assume the following use cases.

- A Database Manager oversees a recurring data load job that loads data to the cube on a monthly basis. The Database Manager can now use a substitution variable to load data for the current month, instead of maintaining a load rule for each month.
- An Application Manager maintains drill through report definitions for different business use cases. The Application Manager implements variables in the underlying Datasource that Smart View users pull from in their drill through operations. As a result, the Application Manager has fewer drill through report definitions to maintain and debug.

Set a Default Parameter in a Datasource

Set a default parameter in a Datasource if you want to enable the use of variables in the queries Essbase generates when it works with data stored outside the cube.

To set the default parameter,

1. Obtain or create a connection to the external source of data (for example, create a connection to Oracle Database).

   You can use a global connection, if one already exists in the Sources page of the Essbase web interface, or you can create an application level connection.

2. Create a Datasource over the connection you will use to access Oracle Database.

   You can define a Datasource globally if it should be available to all applications, or you can define it at the application level.

   a. To create a global Datasource, you must be a service administrator. On the Home page, click **Sources**, click the **Datasources** tab under your user name, and click **Create Datasource**.

      Or, to create an application level Datasource, you must be an application manager or a power user with application management permission to the specified application. On the Home page, click an application name. Then click **Sources**, click the **Datasources** tab under your user name, and click **Create Datasource**.

   b. In the **General** step, for **Connection**, select the Oracle Database connection you created.

   c. For **Name**, give a name to the Datasource.

   d. For the **Query**, provide a query (this example uses SQL). To make it a parameterized query, you must include a filter condition (WHERE clause) that maps a relational column in your source to a placeholder. You indicate the position of the variable by using a placeholder, **?**, in the query syntax. The placeholder is for a parameter you will pass in a later step.

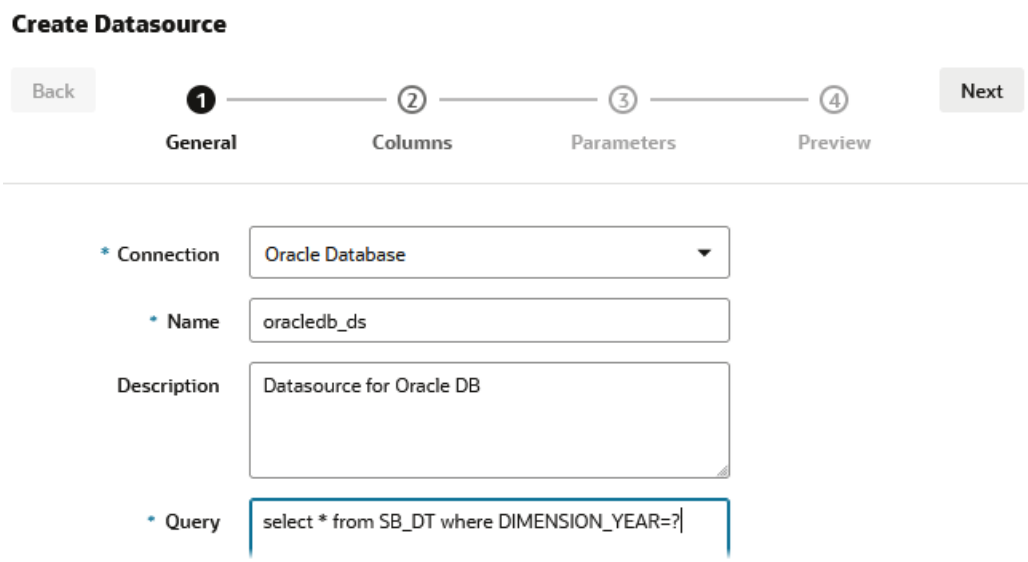
      ```
      select * from SB_DT where DIMENSION_YEAR=?
      ```


Copyright © 2019, 2026, Oracle and/or its affiliates.

For example, assume your relational database has the following table, named SB_DT. The table has DIMENSION_YEAR column with months as values:

	DIMENSION_PRODUCT	DIMENSION_MARKET	DIMENSION_YEAR	DIMENSION_SCENARIO	SALES	COGS	MARKETING	PAYROLL	MI
22	100-20	Louisiana	Jul	Budget	180	70	10	10	(nu
23	100-20	Louisiana	Aug	Actual	154	63	17	11	
24	100-20	Louisiana	Aug	Budget	190	70	10	10	(nu
25	100-20	Louisiana	Sep	Actual	126	51	14	11	
26	100-20	Louisiana	Sep	Budget	150	60	10	10	(nu
27	100-20	Louisiana	Oct	Actual	118	48	13	11	
28	100-20	Louisiana	Oct	Budget	160	70	10	0	(nu
29	100-20	Louisiana	Nov	Actual	78	31	8	11	
30	100-20	Louisiana	Nov	Budget	90	30	0	10	(nu
31	100-20	Louisiana	Dec	Actual	85	34	9	11	
32	100-20	Louisiana	Dec	Budget	100	40	0	10	(nu
33	100-20	New Mexico	Jan	Actual	99	88	27	23	
34	100-20	New Mexico	Jan	Budget	120	110	20	20	(nu
35	100-20	New Mexico	Feb	Actual	102	84	26	23	
36	100-20	New Mexico	Feb	Budget	120	100	20	20	(nu
37	100-20	New Mexico	Mar	Actual	106	88	27	23	
38	100-20	New Mexico	Mar	Budget	130	110	20	20	(nu
39	100-20	New Mexico	Apr	Actual	133	93	28	23	
40	100-20	New Mexico	Apr	Budget	160	110	20	20	(nu
41	100-10	Oklahoma	Aug	Actual	155	68	22	12	

To use a variable for the selection of month values from the DIMENSION_YEAR column, apply the following filter syntax in the query: `where DIMENSION_YEAR=?`

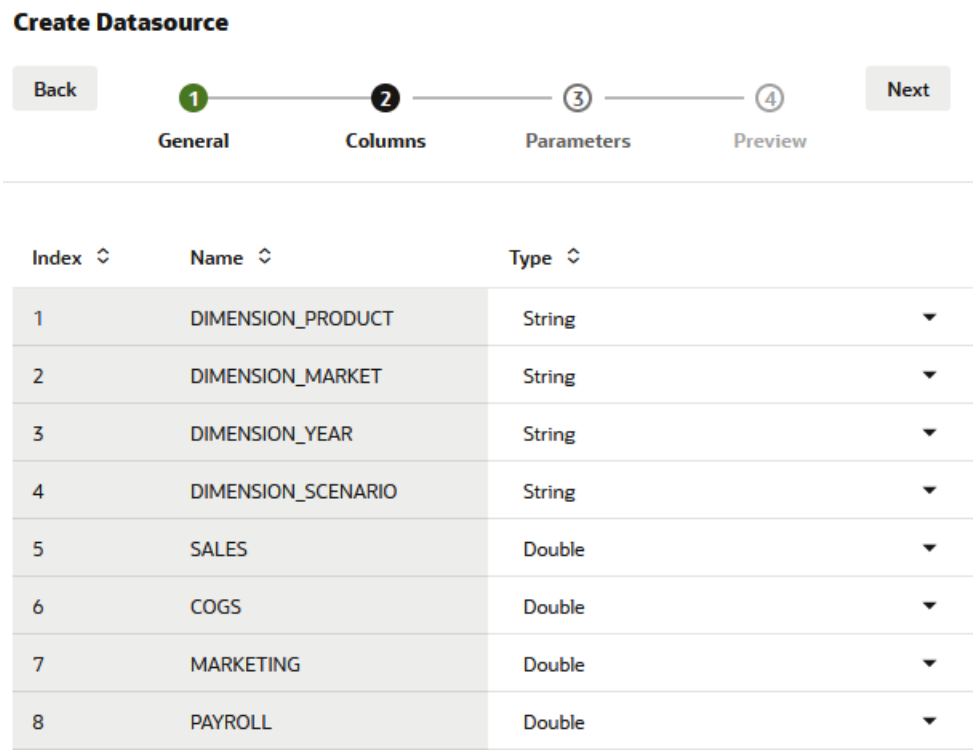


e. Click **Next**.

f. In the **Columns** step, apply the appropriate data type that Essbase should associate with each column from your relational source data.

   For example, set the numeric columns to type **Double**, and leave the alphanumeric columns as type **String**.

Copyright © 2019, 2026, Oracle and/or its affiliates.

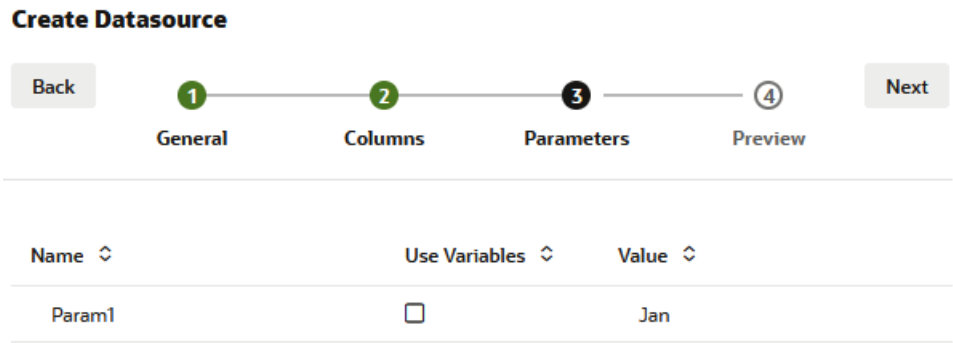


g. Click **Next**.

h. In the Parameters step, Param1 is created – this parameter exists because you used a ? in the query on the General step.

   Leave **Use Variables** unchecked, double-click the text field under **Value**, and type in a default value for the runtime parameter. The purpose of this default value is for Essbase to use as a fallback in case the parameter has an invalid context at runtime. This step is important if you intend to use runtime parameters as part of drill through report definitions.

   You can also rename Param1 to a name that is meaningful for your use case. For example, you can rename it to *param_G_month* to indicate that the parameter uses a global variable for the current month, or you can rename it to *param_<appName>_month* to indicate that the parameter uses an application-level variable for the current month. Customizing the parameter names can be helpful when debugging parameters using Essbase server log files.

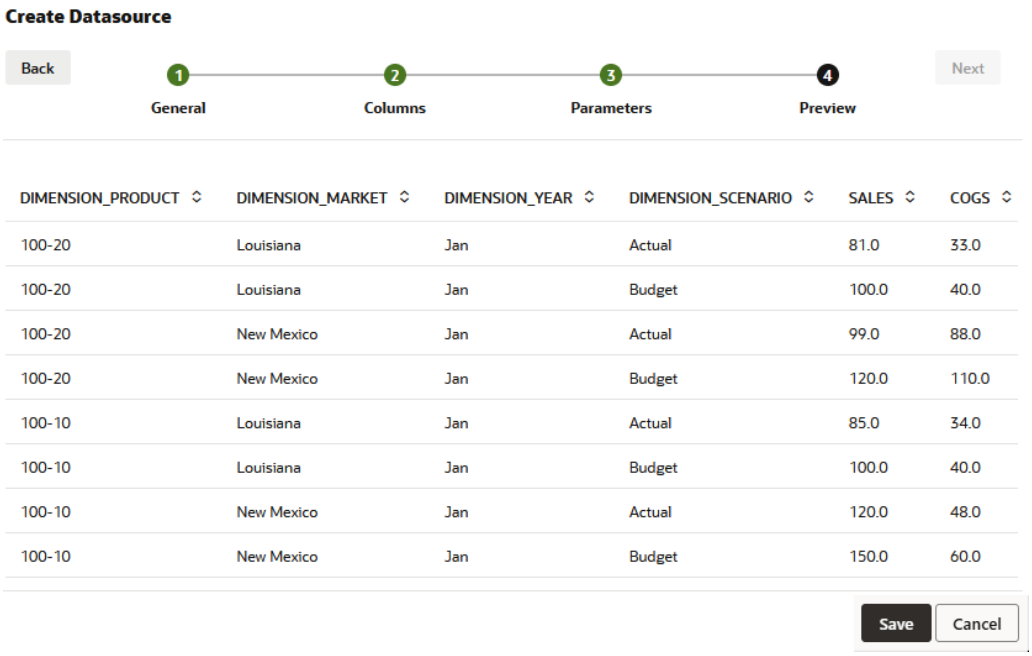


Copyright © 2019, 2026, Oracle and/or its affiliates.

If you want to customize the parameter to reference a substitution variable, then you do not have to provide a default value. See Use Substitution Variables in a Datasource instead of this topic.

i. Click **Next**.

j. In the **Preview**, notice that the default parameter has been applied to your query. As a result, the preview is populated only with external source records in which the value of DIMENSION_YEAR column is Jan.



DIMENSION_PRODUCT	DIMENSION_MARKET	DIMENSION_YEAR	DIMENSION_SCENARIO	SALES	COGS
100-20	Louisiana	Jan	Actual	81.0	33.0
100-20	Louisiana	Jan	Budget	100.0	40.0
100-20	New Mexico	Jan	Actual	99.0	88.0
100-20	New Mexico	Jan	Budget	120.0	110.0
100-10	Louisiana	Jan	Actual	85.0	34.0
100-10	Louisiana	Jan	Budget	100.0	40.0
100-10	New Mexico	Jan	Actual	120.0	48.0
100-10	New Mexico	Jan	Budget	150.0	60.0

Although the preview only displayed values with the default parameter applied, later, when you implement runtime parameters for drill through report definition, you will have access to more external data than what was visible in the preview.

k. Click **Save** to create the Datasource based on this query of your external source data. The Datasource is enabled for implementation of runtime parameters.

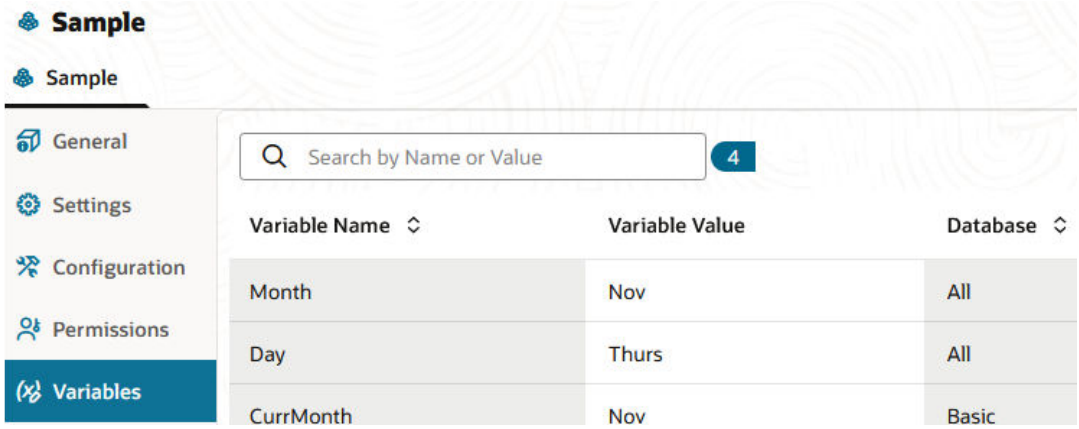
Use Substitution Variables in a Datasource

The following workflow illustrates how to create an Essbase Datasource from a query of external source data, using a substitution variable defined in Essbase. The substitution variable adds more flexibility to how you design the query that pulls from your source data.

In this example, you will use a substitution variable in Essbase to declare the current month. Instead of updating Datasources monthly to pull in data for the current month, you can leave the Datasources alone, and only update the substitution variable you defined.

1. Create a global or application level substitution variable.

Copyright © 2019, 2026, Oracle and/or its affiliates.



2. Obtain or create a connection to the external source of data (for example, create a connection to Oracle Database).

   You can use a global connection, if one already exists in the Sources page of the Essbase web interface,or you can create an application level connection.

3. Create a Datasource over the connection you will use to access Oracle Database.

   You can define a Datasource globally if it should be available to all applications, or you can define it at the application level.

   a. In the **General** step, for **Connection**, select the Oracle Database connection you created.

   b. For **Name**, give a name to the Datasource.

   c. For the **Query**, provide a query (this example uses SQL). To make it a parameterized query, you must include a filter condition (WHERE clause) that maps a relational column in your source to a placeholder. You indicate the position of the variable by using a placeholder, ?, in the query syntax. The placeholder is for a parameter you will pass in a later step.

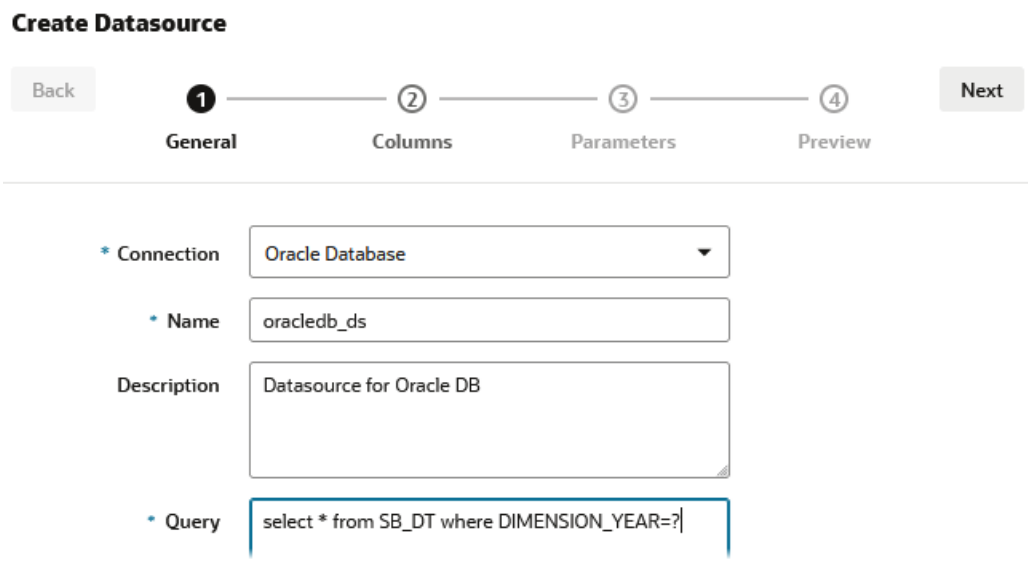
```
select * from SB_DT where DIMENSION_YEAR=?
```

   For example, assume your relational database has the following table, named SB_DT. The table has DIMENSION_YEAR column with months as values:

	DIMENSION_PRODUCT	DIMENSION_MARKET	DIMENSION_YEAR	DIMENSION_SCENARIO	SALES	COGS	MARKETING	PAYROLL	MI
22	100-20	Louisiana	Jul	Budget	180	70	10	10	(nu
23	100-20	Louisiana	Aug	Actual	154	63	17	11	
24	100-20	Louisiana	Aug	Budget	190	70	10	10	(nu
25	100-20	Louisiana	Sep	Actual	126	51	14	11	
26	100-20	Louisiana	Sep	Budget	150	60	10	10	(nu
27	100-20	Louisiana	Oct	Actual	118	48	13	11	
28	100-20	Louisiana	Oct	Budget	160	70	10	0	(nu
29	100-20	Louisiana	Nov	Actual	78	31	8	11	
30	100-20	Louisiana	Nov	Budget	90	30	0	10	(nu
31	100-20	Louisiana	Dec	Actual	85	34	9	11	
32	100-20	Louisiana	Dec	Budget	100	40	0	10	(nu
33	100-20	New Mexico	Jan	Actual	99	88	27	23	
34	100-20	New Mexico	Jan	Budget	120	110	20	20	(nu
35	100-20	New Mexico	Feb	Actual	102	84	26	23	
36	100-20	New Mexico	Feb	Budget	120	100	20	20	(nu
37	100-20	New Mexico	Mar	Actual	106	88	27	23	
38	100-20	New Mexico	Mar	Budget	130	110	20	20	(nu
39	100-20	New Mexico	Apr	Actual	133	93	28	23	
40	100-20	New Mexico	Apr	Budget	160	110	20	20	(nu
41	100-10	Oklahoma	Aug	Actual	155	68	22	12	

Copyright © 2019, 2026, Oracle and/or its affiliates.

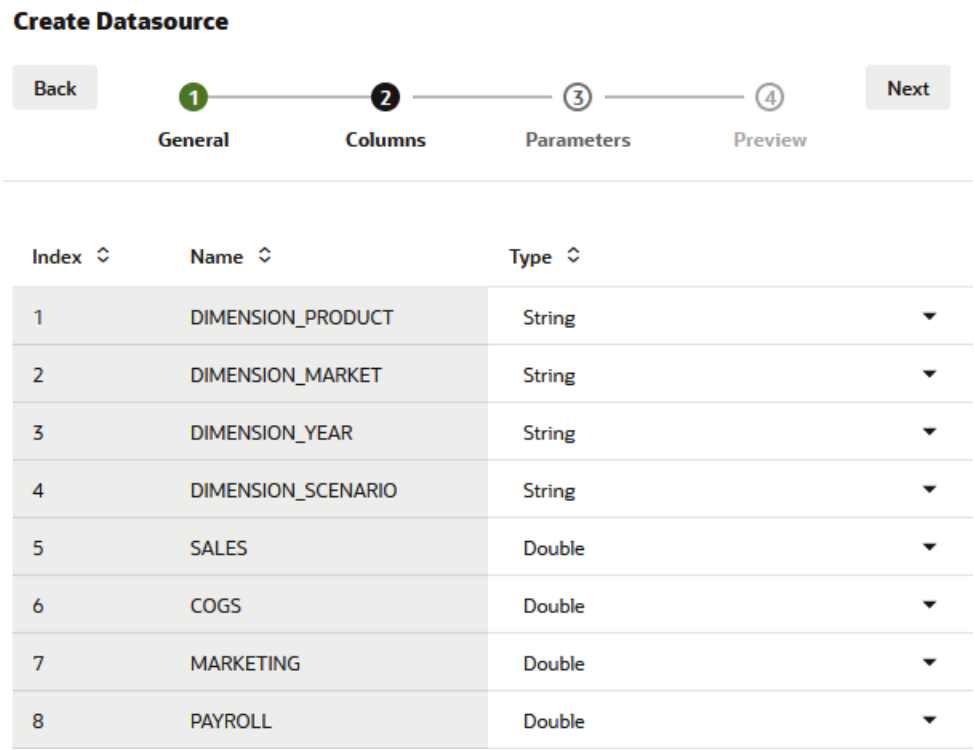
To use a variable for the selection of month values from the DIMENSION_YEAR column, apply the following filter syntax in the query: `where DIMENSION_YEAR=?`



d. Click **Next**.

e. In the **Columns** step, apply the appropriate data type that Essbase should associate with each column from your relational source data.

For example, set the numeric columns to type **Double**, and leave the alphanumeric columns as type **String**.



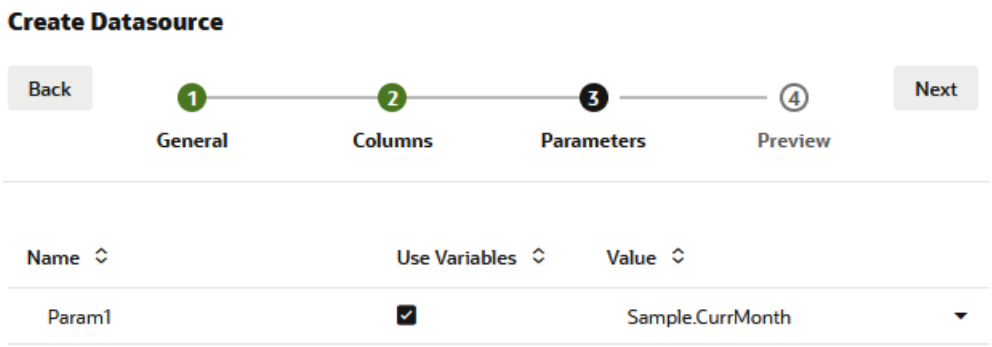
Copyright © 2019, 2026, Oracle and/or its affiliates.

f. Click **Next**.

g. In the Parameters step, Param1 is created – this parameter exists because you used a ? in the query on the General step. To customize **Param1** to reference a substitution variable, click **Use Variables**, and select a substitution variable from the **Value** drop-down list.

   If you are creating a Datasource within an application, both global and application-level substitution variables are available to select. The application-level variables are prefixed with the application name. If you are creating a global Datasource, only global substitution variables are available to select.

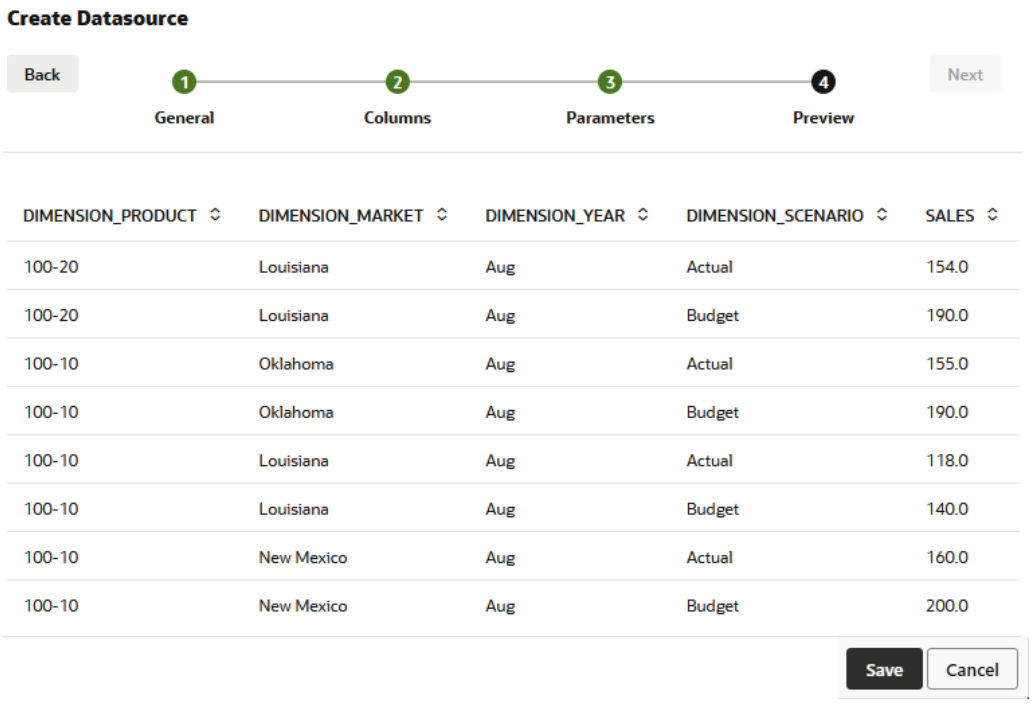
   You can rename **Param1** to a name that is meaningful for your use case. For example, you can rename it to *param_G_month* to indicate that the parameter uses a global variable for the current month, or you can rename it to *param_<appName>_month* to indicate that the parameter uses an application-level variable for the current month. Customizing the parameter names can be helpful when debugging parameters using Essbase server log files.



h. Click **Next**.

i. In the **Preview**, notice that the substitution variable is applied to your query. As a result, the preview is populated only with external source records in which the value of DIMENSION_YEAR column is Aug.

Copyright © 2019, 2026, Oracle and/or its affiliates.



DIMENSION_PRODUCT	DIMENSION_MARKET	DIMENSION_YEAR	DIMENSION_SCENARIO	SALES
100-20	Louisiana	Aug	Actual	154.0
100-20	Louisiana	Aug	Budget	190.0
100-10	Oklahoma	Aug	Actual	155.0
100-10	Oklahoma	Aug	Budget	190.0
100-10	Louisiana	Aug	Actual	118.0
100-10	Louisiana	Aug	Budget	140.0
100-10	New Mexico	Aug	Actual	160.0
100-10	New Mexico	Aug	Budget	200.0

j. Click **Save** to create a Datasource based on this query of your external source data.

Build Dimensions and Load Data

This information moved to *Database Administrator's Guide for Oracle Essbase*.

See:

- Understanding Data Loading and Dimension Building
- Work with Load Rules
- Performing and Debugging Data Loads or Dimension Builds
- Understanding Advanced Dimension Building Concepts

Create a GenAI Connection

Define a GenAI connection to your application.

1. Log in to the Essbase web interface as a service administrator.
2. Click **Sources**, then select **Create Connection**.
3. Select **GenAI** as the **Connection Type**.
4. Enter a descriptive name for your AI connection in the **Name** field.
5. The **Service Provider** is set to Oracle by default.
6. Click **Database connection** and choose the database for which you want to create an AI connection.
7. Fill in the parameters to complete the **Vector and chat credential details** section:
   a. **User OCID**: Find this value in the **Configuration file preview**.

Copyright © 2019, 2026, Oracle and/or its affiliates.

b. **Tenancy OCID**: Find this value in the **Configuration file preview**.

c. **OCI Compartment ID**: Go to **Identity** in the Oracle Cloud Infrastructure console, then select **Compartments**. You can find the compartment ID there.

d. **Private Key**: Refer to the Required Keys and OCIDs for instructions to create your private key.

e. **Fingerprint**: Find this value in the **Configuration file preview**.

> **Note**
>
> **To get your OCI credentials:**
>
> a. Sign in to the Oracle Cloud Infrastructure console.
>
> b. Click **Profile** icon, then select **User settings**.
>
> c. Go to **Tokens and keys**.
>
> d. Under the **Fingerprint** list, select your API key.
>
> e. Click the action menu.
>
> f. Select **View configuration file** to view the **Configuration file preview**.

> **Note**
>
> **If you haven't set up your OCI configuration:**
>
> a. Sign in to the Oracle Cloud Infrastructure console.
>
> b. Click **Profile** icon, then select **User settings**.
>
> c. Go to **Tokens and keys**.
>
> d. Click **Add API key**.
>
> e. Choose **Paste a public key**.
>
> f. On your VM, open your public key file and copy the public key content.
>
> g. Return to the Oracle Cloud Infrastructure page and paste your public key content.
>
> h. Click **Add**.
>
> i. After you add your API key, click action menu and then select **View configuration file** to view the **Configuration file preview**.

For more details, refer to the Required Keys and OCIDs.

8. Use the following details to complete the **Chat profile details** section and **Vectorization and MDX generator details** section:

   a. **Chat model:** Select any available model listed under Pretrained foundational models

   b. **OCI region:** Select a region where the GenAI service is currently available.

   c. **Generative AI service URL:** Get this from the inference API of any model listed under Pretrained foundational models.
   For example:

Copyright © 2019, 2026, Oracle and/or its affiliates.

`https://inference.generativeai.us-chicago-1.oci.oraclecloud.com/20231130/actions/embedText`

d. **Vector embedding model:** Select any available model listed under Pretrained foundational models

e. **Transfer timeout:** 3000

f. **Description:** (Optional)

Copyright © 2019, 2026, Oracle and/or its affiliates.

10
Analyze Data in the Web Interface

For convenience, you can perform analysis on cube data from the Essbase web interface.

To analyze data grids in the Essbase web interface,

1. Log in to Essbase with at least Database Access role for the application whose cube data you want to analyze.
2. Open **Analyze Data**:
   - In the Redwood Interface, on the Home page, open the application and then open the database (cube), and click **Analyze Data**.
   - In the Classic Web Interface, on the Applications page, expand the application, and highlight the row containing the cube name, and from the **Actions** menu to the right of the cube name, click **Analyze Data**.

A grid is displayed in the Ad Hoc Analysis tab. In this tab, you can:

- Perform ad hoc analysis against the cube you selected when you opened the Analyze Data view.
- Save a grid layout that you can refresh when you use the Ad Hoc Analysis tab in the future.

On the Reports tab, you can use MDX to write sophisticated data queries to populate the grid and to save as named reports.

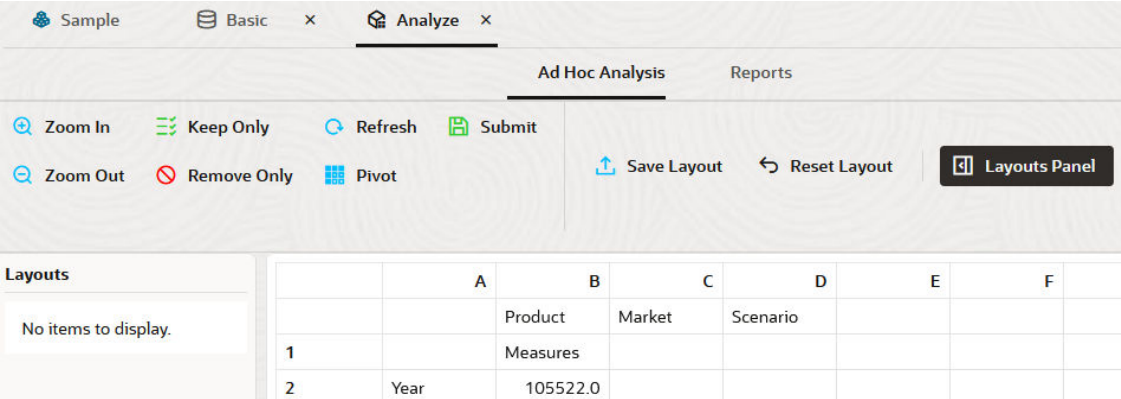
Perform Ad Hoc Analysis in the Web Interface

You can use Essbase web interface to perform analysis and navigate through your data. In the Ad Hoc Analysis tab of the Analyze Data view, a grid is displayed containing each of the base dimensions (non-attribute dimensions) from the cube.

You may or may not see data in the ad hoc grid, depending on your filter access and how data is stored in the cube. Data is not always stored at the topmost member for every dimension hierarchy.

Use the ad hoc navigation buttons at the top left of the Ad Hoc Analysis tab to navigate to data that you are allowed to see. If your filter grants you write permission on the cube, the **Submit** button enables you to update data for stored intersections within the scope of your filtered access.

Copyright © 2019, 2026, Oracle and/or its affiliates.



Work with Layouts

If you create an Essbase query grid in the Analyze view that you would like to use again in the future, you can save it as a Layout.

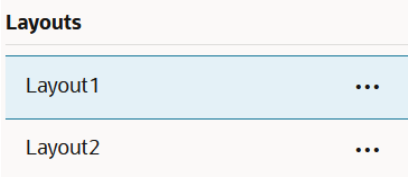
To create a named layout in the Essbase web interface,

1. In the Analyze Data view for your cube, on the Ad Hoc Analysis tab, create an ad hoc grid that you want to save.
2. Click **Save Layout**.
3. Enter a name for your layout, and optionally, a description.
4. If you want to see this grid each time you analyze data, instead of the database default ad hoc query, check the **Default** box.
5. Click **Save**.

The last ad hoc grid that was rendered during your session will be displayed the next time you log in, unless a default is set.

To view a grid previously saved as a layout,

1. If layouts are not listed by name in the Ad Hoc Analysis tab, click the **Layouts Panel** button to display the list.
2. Click the name of a stored layout to render it in the grid.



To return to the default ad hoc view, click the **Reset Layout** button.

To delete or edit layouts that you created, use the Actions menu next to the layout name. The Edit option allows you to select the layout as your default, update the description, or remove the default setting on a layout previously set as your default.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Access to Layouts

How you work with Essbase layouts depends on your cube access.

Clicking on a saved layout name causes it to render data in the Ad Hoc Analysis tab of the Analyze Data view.

Users with, at minimum, the application-level role of Database Manager can:

- See and render layouts created by others for this cube.
- Designate a layout to be the database default. This layout is shown to all cube users when they analyze data, unless they have previously created their own user default layouts.
- Delete layouts created by any user of this cube.

Layouts and reports are included when the cube is copied or moved using migration, export, and Lifecycle Mangement (LCM) tools.

Analyze and Manage Data with MDX

MDX (Multidimensional Expressions) is a powerful data manipulation and querying language.

With MDX, you can:

- Query and report against data and metadata in Essbase cubes
- Insert data into an Essbase cube
- Export data from an Essbase cube

An MDX query is a single MDX statement, having exactly one result set, that applies to a single cube.

An MDX report is a single MDX query, saved in the cube context. You can access MDX reports from Smart View and from the Essbase web interface.

An MDX script is a file, with an `.mdx` extension, that you can upload and then run from Jobs or in Smart View. Only MDX Insert and Export statements should be used in MDX scripts. To analyze grid data, use MDX reports rather than MDX scripts.

- Analyze Data with MDX Reports
- Insert and Export Data with MDX
- Run MDX Scripts

Analyze Data with MDX Reports

You can store and render queries in the Essbase web interface using MDX reports. The minimum permission required to create a report is Database Manager.

Defining Layouts using the Ad Hoc Analysis tab may not always be the most efficient way to create a sophisticated report. If you know exactly what you want to query, you can use MDX to create a query to populate the grid.

To create an MDX report:

1. Log in to the Essbase web interface as a Database Manager or higher role.
2. Navigate to **Analyze Data**:

Copyright © 2019, 2026, Oracle and/or its affiliates.

- On the Home page, navigate to the application, and select the Actions menu ⋯. Choose **Databases** > *Database Name* > **Analyze Data**.
- In the Classic Web Interface, expand the application, select a cube, click the Actions menu to the right of the cube name, and select **Analyze Data**.

3. In the Analyze view, select the **Reports** tab and click **Create**.
4. Enter a name for the report, and optionally, a description.
5. In the Query field, enter an MDX query relevant to the current cube. For example:

```
SELECT
  {([West].children)}
ON COLUMNS,
  {([Diet].children)}
ON ROWS
```

   Because the context of Analyze Data is the active cube, we recommend that you omit the optional cube specification (the FROM clause) from MDX reports. Omitting the FROM clause allows for more flexibility—if the cube is copied or renamed, the report will work in the new cube.

   Substitution variables are supported in MDX reports, but not runtime substitution variables. To use runtime substitution variables, save the MDX query as a script, and run it from Smart View using **Calculate** on the Essbase ribbon.
6. Click **Validate** to verify your MDX syntax, and then click **Save**.
7. From the Reports panel on the left, select the saved report to render a grid.

To learn more about MDX, see MDX and Writing MDX Queries.

Access to MDX Reports

How you work with Essbase MDX reports depends on your cube access.

Users with, at minimum, the application-level role of Database Access can render saved MDX reports created by others. The data a user sees displayed in the report depends on that user's filter access.

In addition to rendering saved reports, Database Access users can export result sets in various formats: HTML, CSV, Excel, and JSON.

Database Access users can also view the MDX query that defines the report, by clicking the **Actions** menu next to the report name and selecting **View**.

If you have at least Database Manager role, you can use reports in the same ways that Database Access users can. Additionally, you can edit and delete reports using the **Actions** menu.

If you are a Service Administrator, you can additionally use the **Execute As** button to impersonate other users and check their data access. This can be useful for testing filters assigned to various users.

Examples of MDX Reports

The MDX examples in this section demonstrate special types of analyses you can perform, using MDX reports. Save MDX reports in the Essbase web interface's Analyze view, on the Reports tab.

Copyright © 2019, 2026, Oracle and/or its affiliates.

The following examples are designed to work on the Sample Basic cube.

Metadata Report

The following example returns only metadata (member names, but no data):

```
SELECT
  {[Product].Levels(1).Members}
ON ROWS,
  {}
ON COLUMNS
```

returning the grid:

	A
1	100
2	200
3	300
4	400
5	Diet

Attribute Report

The following example uses, on columns, members from an attribute dimension:

```
SELECT
 [Product].Children
ON ROWS,
 [Ounces].Children
ON COLUMNS
WHERE {Sales}
```

returning the grid:

	A	B	C	D	E
1		Ounces_32	Ounces_20	Ounces_16	Ounces_12
2	100	#Missing	#Missing	12841.0	93293.0
3	200	#Missing	#Missing	49990.0	59096.0
4	300	#Missing	64436.0	#Missing	36969.0
5	400	84230.0	#Missing	#Missing	#Missing
6	Diet	#Missing	#Missing	38240.0	67438.0

Filtered Report

The following example uses a slicer (WHERE clause) to limit the query to Cola. Additionally, the Filter function limits the level 0 markets in the query to those that have a negative profit.

```
SELECT
  { Profit }
ON COLUMNS,
  Filter( [Market].levels(0).members, Profit < 0)
ON ROWS
WHERE {Cola}
```

returning the grid:

Copyright © 2019, 2026, Oracle and/or its affiliates.

	A	B
1		Profit
2	Oregon	-234.0
3	Utah	-31.0
4	Nevada	-210.0
5	Oklahoma	-102.0
6	Louisiana	-305.0
7	Ohio	-22.0
8	Wisconsin	-310.0
9	Missouri	-87.0
10	Iowa	-874.0

UDA Report

The following example shows Product data for Market dimension members that have a user defined attribute (UDA) of "Major Market." A slicer (WHERE clause) limits the query to include only Sales data.

```
SELECT
  [Product].Children
ON ROWS,
  {Intersect(UDA([Market], "Major Market"), [Market].Children)}
ON COLUMNS
WHERE {Sales}
```

returning the grid:

	A	B	C
1		East	Central
2	100	27740.0	33808.0
3	200	23672.0	29206.0
4	300	20241.0	33215.0
5	400	15745.0	33451.0
6	Diet	7919.0	42660.0

Insert and Export Data with MDX

In addition to being useful for Essbase grid-based analysis, MDX also enables you to copy and update subsets of multidimensional data.

The MDX Insert clause enables you to update the cube with data, using a calculated (non-physical) member that you define using MDX.

The MDX Export clause enables you to save and export query results as data subsets that you can view or import later.

Insert and Export MDX statements can be run as saved MDX scripts.

To learn more about MDX Insert and Export, see MDX Insert Specification and MDX Export Specification.

Run MDX Scripts

Use MDX scripts when you need to execute Insert or Export data operations.

Copyright © 2019, 2026, Oracle and/or its affiliates.

For analysis of grid data, use MDX reports. See Analyze Data with MDX Reports.

To use MDX scripts, select a workflow:

- Write, Upload, and Run an MDX Script
- Write an MDX Script in the Script Editor and Run It
- Create an MDX Script in Cube Designer and Run it

Write, Upload, and Run an MDX Script

Use this workflow to write MDX scripts in a text editor and upload them to Essbase.

1. Write the MDX script in a text editor, and save it with an `.mdx` extension.
2. Upload the MDX script to the application or cube directory under **Files** in the Essbase web interface.
3. Run the MDX script from **Jobs** or from Smart View, using **Calculate** on the Essbase ribbon.

Write an MDX Script in the Script Editor and Run It

Use this workflow to write MDX scripts in a script editor on the cube, and run them from **Jobs**.

- Redwood
- Classic

Redwood

1. On the Home page, open the application and open the database (cube).
2. Click **Scripts**, and click **MDX Scripts**.
3. Click **Create**, enter a name for the script and click **OK**.
4. Write the MDX script. A member tree and function list can help you.
5. Validate and save the script, then close the script editor.
6. Run the MDX script from **Jobs** (see Run MDX), or if using Smart View, using **Calculate** on the Essbase ribbon.

Classic

1. On the Applications page, expand an application and cube.
2. From the cube's Actions menu, click **Inspect**.
3. Click **Scripts**, and then click **MDX Scripts**.
4. Click **+** to open a script editor.
5. Write the MDX script. A member tree and function list can help you.
6. Validate and save the script, then close the script editor.
7. Run the MDX script from **Jobs** (see Run MDX), or if using Smart View, using **Calculate** on the Essbase ribbon.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Create an MDX Script in Cube Designer and Run it

Use this workflow to create MDX scripts using an application workbook, and run them from **Jobs**.

1. In an application workbook, create an MDX worksheet. See Work with MDX Worksheets in Cube Designer.
2. Add a file name in the **File Name** field.
3. Indicate, in the **Execute MDX** field, whether to execute the MDX at the time the cube is created. Valid entries are **Yes** and **No**.
4. Add the MDX script below the **Script** line.
5. Save the application workbook.
6. Build the cube. See Create an Application and Cube in Cube Designer.
7. Run the MDX script from **Jobs**, or if using Smart View, using **Calculate** on the Essbase ribbon.

Guidelines for MDX Scripts

Use MDX scripts to perform Insert or Export data operations. For grid analysis in the Essbase web interface, use MDX reports instead of MDX scripts. MDX scripts can optionally include runtime substitution variables, and you can run them from Smart View.

To be usable within Smart View, MDX scripts with runtime substitution variables must use the XML syntax within the SET RUNTIMESUBVARS calculation command, including `<RTSV_HINT>`.

To set a runtime substitution variable so that it calculates only the visible slice of data in Smart View, set the value of the runtime substitution variable to `POV`, and set the data type to `member`.

When run from the Essbase web interface, your MDX scripts may use substitution variables, but not runtime substitution variables. To use runtime substitution variables in MDX scripts, you must run the scripts from Smart View, using **Calculate** on the Essbase ribbon.

Use Substitution Variables

Examples of MDX Scripts

The following are examples of MDX Insert and Export scripts you can run on the Sample Basic cube, either from Jobs in the Essbase web interface, or from the **Calculate** operation Smart View.

MDX Insert

You can save this `.mdx` script and run it from **Jobs** or from the **Calculate** dialog in Smart View.

```
INSERT "([Measures].[Payroll])" TO "([Measures].[Revised_Payroll])"
INTO [Sample].[Basic]
FROM (
       SELECT
            {[Measures].[Payroll]} ON COLUMNS,
                {Crossjoin
                  (Crossjoin(Descendants([Year]),
```


Copyright © 2019, 2026, Oracle and/or its affiliates.

```
                    Crossjoin(Descendants([Scenario]),
                   Descendants([Product])),
                   Descendants([Market]))} ON ROWS
        FROM [Sample].[Basic]
);
```

The above example assumes you have previously added a Revised_Payroll measure to Sample Basic.

MDX Export

You can save this `.mdx` script and run it from **Jobs** or from the **Calculate** dialog in Smart View.

```
EXPORT INTO FILE "sample01" OVERWRITE
SELECT
 {[Mar],[Apr]}
ON COLUMNS,
 Crossjoin({[New York]},
  Crossjoin({[Actual],[Budget]},
     {[Opening Inventory],[Ending Inventory]}))
ON ROWS
FROM [Sample].[Basic]
WHERE ([100-10])
```

After you run the script, the following export file, `sample01.txt`, is saved in the cube directory of the file catalog:

```
Market,Scenario,Measures,Mar,Apr
New York,Actual,Opening Inventory,2041,2108
New York,Actual,Ending Inventory,2108,2250
New York,Budget,Opening Inventory,1980,2040
New York,Budget,Ending Inventory,2040,2170
```

MDX Export Using Runtime Substitution Variable

You can save this `.mdx` script and run it from the **Calculate** dialog in Smart View.

```
SET RUNTIMESUBVARS
{
 States = "Massachusetts"<RTSV_HINT><svLaunch>
                    <description>US States</description>
                    <type>member</type>
                    <allowMissing>false</allowMissing>
                    <dimension>Market</dimension>
                    <choice>multiple</choice>
                    </svLaunch></RTSV_HINT>;
};
EXPORT INTO FILE "sample002" OVERWRITE
SELECT
 {[Mar],[Apr]}
ON COLUMNS,
 Crossjoin({&States}, Crossjoin({[Actual],[Budget]},
 {[Opening Inventory],[Ending Inventory]}))
ON ROWS
```


Copyright © 2019, 2026, Oracle and/or its affiliates.

```
FROM [Sample].[Basic]
WHERE ([100-10])
```

After you run the script, the following export file, `sample002.txt`, is saved in the cube directory of the file catalog:

```
Market,Scenario,Measures,Mar,Apr
Massachusetts,Actual,Opening Inventory,-54,-348
Massachusetts,Actual,Ending Inventory,-348,-663
Massachusetts,Budget,Opening Inventory,-160,-520
Massachusetts,Budget,Ending Inventory,-520,-910
```

Access External Data with Drill Through Reports

Sometimes users may require more information than what exists in the Essbase cube. You can set up access to additional data from the source system using drill through reports.

- Introduction to Essbase Drill Through
- Access to Drill Through Reports
- Design Drill Through Reports
- Test Drill Through Reports
- Drill Through to a URL
- Drill Through from Multiple Cells
- Debug Drill Through using Essbase Server Platform Log

Introduction to Essbase Drill Through

When Smart View users need more information than what is available in the Essbase cube, drill through reports can provide customized access to external source data.

Typically, given the detailed granularity of data kept in data warehouses and other source data systems, the volume of external data can be too large to be useful for analysis. To populate an Essbase cube with the optimal amount of data for analysis, a common practice is to aggregate the source data (for example, aggregating daily transactional values to weekly or monthly totals), and load this aggregate data to the Essbase cube.

During subsequent analyses of the data in Essbase, if Smart View users find anomalies worthy of investigation, drill through provides them a way to quickly view the underlying source data to search for the cause. For example, if August data is unexpectedly different from July, users can drill through to the source system to find which record(s) may have been responsible.

To provide analytical Smart View users additional information about what constitutes one or more data values in the cube, a user with Database Manager permission or higher can implement drill through reports to give more depth of insight into the source data before it is rolled up and loaded into Essbase.

Drill through reports are one way you can build a data exchange interface between Essbase and external source systems.

Consider a relational table, SB_DT, with all records selected. The SQL query is:

```
SELECT * FROM SB_DT
```


Copyright © 2019, 2026, Oracle and/or its affiliates.

The query result (in SQL Developer) is truncated for length in this illustration, as there are thousands of records in the table:

Query Result

SQL | Fetched 100 rows in 0.189 seconds

	DIMENSION_PRODUCT	DIMENSION_MARKET	DIMENSION_YEAR	DIMENSION_SCENARIO	SALES	COGS	MARKETING	PAY
1	100-20	Oklahoma	Sep	Actual	72	29	8	
2	100-20	Oklahoma	Sep	Budget	90	30	0	
3	100-20	Oklahoma	Oct	Actual	66	27	7	
4	100-20	Oklahoma	Oct	Budget	90	30	0	
5	100-20	Oklahoma	Nov	Actual	88	36	10	
6	100-20	Oklahoma	Nov	Budget	100	40	10	
7	100-20	Oklahoma	Dec	Actual	82	33	9	
8	100-20	Oklahoma	Dec	Budget	90	30	0	
9	100-20	Louisiana	Jan	Actual	81	33	9	
10	100-20	Louisiana	Jan	Budget	100	40	0	
11	100-20	Louisiana	Feb	Actual	115	47	13	
12	100-20	Louisiana	Feb	Budget	140	50	10	
13	100-20	Louisiana	Mar	Actual	121	49	13	
14	100-20	Louisiana	Mar	Budget	150	60	10	
15	100-20	Louisiana	Apr	Actual	121	49	13	
16	100-20	Louisiana	Apr	Budget	150	60	10	
17	100-20	Louisiana	May	Actual	130	53	14	
18	100-20	Louisiana	May	Budget	160	60	10	
19	100-20	Louisiana	Jun	Actual	144	59	16	
20	100-20	Louisiana	Jun	Budget	180	70	10	
21	100-20	Louisiana	Jul	Actual	144	59	16	
22	100-20	Louisiana	Jul	Budget	180	70	10	
23	100-20	Louisiana	Aug	Actual	154	63	17	
24	100-20	Louisiana	Aug	Budget	190	70	10	
25	100-20	Louisiana	Sep	Actual	126	51	14	
26	100-20	Louisiana	Sep	Budget	150	60	10	
27	100-20	Louisiana	Oct	Actual	118	48	13	
28	100-20	Louisiana	Oct	Budget	160	70	10	
29	100-20	Louisiana	Nov	Actual	78	31	8	
30	100-20	Louisiana	Nov	Budget	90	30	0	
31	100-20	Louisiana	Dec	Actual	85	34	9	

Consider the same table with fewer records selected. For example, if the SQL selection is narrowed down to named columns, measures are aggregated, and a filter (WHERE clause) is applied,

```
select DIMENSION_PRODUCT, DIMENSION_MARKET, YEAR_PARENT, DIMENSION_SCENARIO,
sum(SALES) as SALES, sum(COGS) as COGS
         from SB_DT where DIMENSION_SCENARIO ='Actual' AND DIMENSION_MARKET
='California' AND YEAR_PARENT ='Qtr4' group by DIMENSION_PRODUCT,
DIMENSION_MARKET,
        YEAR_PARENT, DIMENSION_SCENARIO
```

then the query result is aggregated and filtered:

Copyright © 2019, 2026, Oracle and/or its affiliates.

Query Result

SQL | All Rows Fetched: 12 in 0.066 seconds

	DIMENSION_PRODUCT	DIMENSION_MARKET	YEAR_PARENT	DIMENSION_SCENARIO	SALES	COGS
1	300-10	California	Qtr4	Actual	1535	705
2	400-20	California	Qtr4	Actual	443	180
3	400-10	California	Qtr4	Actual	894	364
4	300-30	California	Qtr4	Actual	673	275
5	100-20	California	Qtr4	Actual	468	551
6	200-30	California	Qtr4	Actual	1402	700
7	100-10	California	Qtr4	Actual	1972	788
8	200-20	California	Qtr4	Actual	1706	732
9	300-20	California	Qtr4	Actual	281	122
10	100-30	California	Qtr4	Actual	327	362
11	200-10	California	Qtr4	Actual	909	381
12	400-30	California	Qtr4	Actual	539	214

You can harness the power of RDBMS queries in Essbase, using Datasources, data load, and drill through reports. Drill through reports provide filtered access to an external source of data directly from an Essbase query in a Smart View worksheet.

- Drill Through Terminology
- Workflow for Drill Through Report Design
- How Drill Through Works
- Drill Through Report Definition
- Drill Through Use Case Example

Drill Through Terminology

This topic explains the meaning of terms related to Essbase drill through.

Drill through (verb)

To drill through is to access external data from one or more Essbase cell intersections in a Smart View worksheet. The drill through action provides additional information not contained in the Essbase cube. The need to drill through arises when Essbase contains aggregated (“rolled up”) values and the external source system has more granular data that can be made available.

- If a query is performed upon drill through, the results display in a new worksheet that opens -- this is the drill through report. The report contains information pulled from the external source data.
- If a URL is launched upon drill through, it opens in a Web browser. Parameters can be passed to the URL, to execute a customized search on the website.

Drill through report

A drill through report is the result of a drill through operation, performed from a Smart View grid, to additional data from a source system external to Essbase.

Drill through report definition

A drill through report definition is the way, if you are a Database Manager or higher, to define the access your users should have to external information. You create drill through report definitions in the Essbase web interface or REST API. They are associated with your cube. As part of the definition, you specify:

Copyright © 2019, 2026, Oracle and/or its affiliates.

- A **column mapping**. This specifies which external columns you want displayed in the reports, and how much hierarchical (generational) depth of access you want to provide (for example, do you want to reveal daily, monthly, or quarterly information from the external source?)
- A **drillable region**. This specifies which cell intersections of your cube offer access to drill through reports (or a URL) containing additional external data. In the examples to follow, drillable regions in the POV of the Smart View worksheet are color coded as blue, using cell styles. You specify drillable regions using any of the Member Set Functions available in Essbase. In the examples to follow, the drillable region is @DESCENDANTS("Measures") on Sample Basic.
- A mapping for **runtime parameters**, if a parameterized query is implemented in the underlying Datasource query (optional).

Drill through report definitions usually rely on predefined connections and Datasources in Essbase (unless you are defining access to a file uploaded to Essbase). A connection stores the authentication details to the external source. One or more Datasources you define over the connection enable you to specify an initial query to fetch from the external source (for example, selecting all from a particular table). The query you specify in the Datasource can fetch as large or small a subset of data as you want to start with. Later, you narrow down how much data access to provide, when you create or edit the drill through report definition.

Workflow for Drill Through Report Design

As a database manager, use the following workflow to design and test drill through for your cube.

1. Prepare data access
   a. Upload a data file, OR
   b. Obtain authorization information needed to access an external source system
   c. Define a connection and one or more Datasources to the data file or to the external source. Refer to Create a Global Connection and Datasource or Create an Application-Level Connection and Datasource.
2. Create drill through report definitions on the cube
   a. define the column mapping
   b. define the drillable region
   c. map runtime parameters, if used
3. Test the drill through reports
   a. Prepare Smart View
      - install the latest version
      - connect to the cube
      - enable cell styles to show drillable regions
   b. Drill through from different cell intersections in Smart View
      - drill through from one intersection
      - drill through from multiple intersections
      - drill through from cells at different generations
   c. Check the drill through report outputs and the platform log

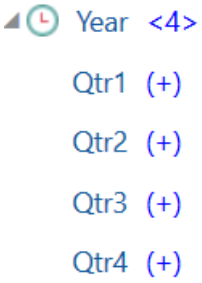
Copyright © 2019, 2026, Oracle and/or its affiliates.

This workflow, and the rest of this introduction, emphasize what you need to know to design and test drill through access to external source data in an RDBMS. If you are more interested in drill through URL implementation, see Drill Through to a URL.

To understand the different access requirements for drill through design and use, see Access to Drill Through Reports.

How Drill Through Works

In this example, assume the Sample Basic cube in Essbase has Qtr1-Qtr4 as the lowest levels of the time dimension.



There are no months in the outline hierarchy, but the monthly data is available in the external source system, in a table column named DIMENSION_YEAR:

DIMENSION_YEAR
Aug
Sep
Oct
Nov
Dec
Jan
Feb
Mar

When a Smart View user drills through on a cell intersection of a Sales value for Qtr1:

	A	B	C	D	E	F	G	H	I
1		Actual	New York	Cola					
2		Sales	COGS	Margin	Total Expenses	Profit	Inventory	Ratios	Measures
3	Qtr1	1998	799	1199	433	766	2101	60.01001001	766
4	Qtr2	2358	942	1416	488	928	2108	60.05089059	928
5	Qtr3	2612	1044	1568	518	1050	2654	60.03062787	1050
6	Qtr4	1972	788	1184	430	754	2548	60.04056795	754
7	Year	8940	3573	5367	1869	3498	9411	60.03355705	3498

The drill through report Essbase generates is

Copyright © 2019, 2026, Oracle and/or its affiliates.

	A	B	C	D	E	F	G
1	DIMENSION_PRC	DIMENSION_MAF	DIMENSION_YEA	DIMENSION_SCE	SALES	COGS	YEAR_PARENT
2	100-10	New York	Jan	Actual	678	271	Qtr1
3	100-10	New York	Feb	Actual	645	258	Qtr1
4	100-10	New York	Mar	Actual	675	270	Qtr1

The drill through report shows additional information, from the source database, about Sales for Qtr1. Notice that the Jan, Feb, and Mar values add up to the value for Qtr1: 678+645+675=**1998**.

The query Essbase internally uses to build the above drill through report is:

```
SELECT
        "DIMENSION_PRODUCT", "DIMENSION_MARKET", "DIMENSION_YEAR",
"DIMENSION_SCENARIO",
        "SALES", "COGS", "YEAR_PARENT"  FROM <Query defined in Datasource>
WHERE
        "YEAR_PARENT" = 'Qtr1' AND "DIMENSION_PRODUCT" = '100-10' AND
"DIMENSION_MARKET" =
        'New York' AND "DIMENSION_SCENARIO" = 'Actual'
```

From the platform log, administrators can access the exact queries behind each drill through report.

Drill Through Report Definition

A drill through report definition is the way, if you are a Database Manager or higher, to define the access your users should have to information that is stored externally to Essbase.

> **Note**
>
> Do not rename drill through report definitions. Drill through report definitions that are renamed may not be editable and may not work as expected.

To enable the drill through report generated in How Drill Through Works, the Database Manager created a drill through report definition associated with Sample Basic. The drill through report references a predefined Datasource that uses a query to pull external data from SB_DT (the hypothetical source system table mentioned in Introduction to Essbase Drill Through).

In the drill through report definition, the Database Manager specified the following column mapping:

External column	Include in report	Essbase dimension	Gen/Lev filter
DIMENSION_PRODUCT	Y	Product	Product SKU [Generation]
DIMENSION_MARKET	Y	Market	State [Generation]
DIMENSION_YEAR	Y	Year	None
DIMENSION_SCENARIO	Y	Scenario	Level0 [Level]
SALES	Y	None	--
COGS	Y	None	--

Copyright © 2019, 2026, Oracle and/or its affiliates.

External column	Include in report	Essbase dimension	Gen/Lev filter
YEAR_PARENT	Y	Year	Quarter [Generation]

The column mapping defines which external source columns should be included in the report, which Essbase dimensions those columns map to, and (optionally) a generation/level filter condition indicating how much depth of access to provide.

In the mapping template shown above, the Database Manager:

- Mapped the external DIMENSION_PRODUCT column to the generation named Product SKU in the Essbase Product dimension. This type of column mapping is called generation mapping.
- Mapped the external DIMENSION_MARKET column to the generation named State in the Essbase Market dimension (this is another example of generation mapping).
- Mapped the external DIMENSION_YEAR column to the Year dimension, with no further filter. This type of column mapping is called dimension mapping.
- Mapped the external DIMENSION_SCENARIO column to the lowest level (level 0) of the Essbase Scenario dimension. This type of column mapping is called level 0 mapping.
- Did not map the columns SALES and COGS to anything, but selected to include these columns in the report. It is not typically necessary to map columns to the Essbase accounts dimension.
- Mapped the external YEAR_PARENT column to the generation named Quarter in the Year dimension.

Column Mapping Options

Because the query that Essbase generates to pull data from your Datasource is highly dependent on your defined column mapping, it is helpful to understand the different ways of mapping columns and in which cases each method is useful. The types of column mapping are:

- Dimension mapping
- Generation mapping
- Level 0 mapping

Dimension Mapping

With Dimension mapping, you map a source data column directly to a dimension name in the Essbase cube. This type of mapping is most useful when the source data column contains all the layers of data represented in the corresponding dimension of your cube.

For example, if a source data column MONTH contains a mixture of all the same generations/levels that the dimension has, as shown,

Copyright © 2019, 2026, Oracle and/or its affiliates.

MONTH
Jan
Qtr3
Feb
Qtr1
Year
Qtr1
Aug
...

then it makes sense to map the MONTH column to the Essbase Year dimension, with no further filter:

External column	Include in report	Essbase dimension	Gen/Lev filter
MONTH	Y	Year	None

When you use a Dimension mapping for MONTH as shown above, the filter condition (the WHERE clause of the SQL query) is *not* predefined for the MONTH column:

```
SELECT "MONTH"
FROM <Query defined in Datasource>
WHERE "MONTH" = '<Grid context>'
```

and the drill through result for MONTH will return the current Smart View cell intersection.

Learn more about how dimension mapping works in Drill Through Use Case Example.

Generation Mapping

With Generation mapping, you map a source data column to a named generation in an Essbase dimension. This type of mapping is useful when the source data column contains only the data layer represented in a specific generation of a dimension in the cube. For example, if the source data column MONTH contains only months, and the Year dimension has months at generation 3,

Copyright © 2019, 2026, Oracle and/or its affiliates.

then the best choice is to map the source column to generation 3 (Months) of the Year dimension:

External column	Include in report	Essbase dimension	Gen/Lev filter
MONTH	Y	Year	Months [Generation]

When you use a Generation mapping for MONTH as shown above, the query filter condition will be predefined for the MONTH column:

```
SELECT "MONTH"
FROM <Query defined in Datasource>
WHERE "MONTH" = '<Generation filter>'
```

and the drill through result for MONTH will return values down to the Months generation of the Year dimension. No data (if it exists) will be returned for any level lower than Months.

Generation mapping is not ideal for asymmetric (ragged) hierarchies. A generation mapping has no effect on the drill through query in a dimension with a ragged hierarchy unless the drill through is performed on a member in a direct ancestral line to the generation where the column mapping is defined. To avoid unexpected results, Oracle recommends using level 0 mapping rather than generation mapping for drilling through on asymmetric hierarchies.

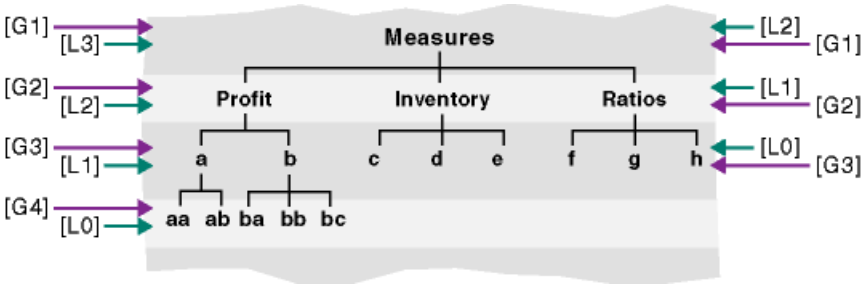
Learn more about how generation mapping works in Drill Through Use Case Example.

Level 0 Mapping

With level 0 mapping, Essbase adds to the filter condition all leaf level members from the hierarchy below the cell intersection (whichever member is selected in the current Smart View grid context at runtime).

Level 0 mapping is useful when working with asymmetric (ragged) hierarchies. In a ragged hierarchy, same-level (L) members do not all share the same generational (G) depth in the outline.

Copyright © 2019, 2026, Oracle and/or its affiliates.



A common example of a ragged hierarchy is an employee organization structure.

The Product dimension of the Essbase cube outline for Sample Basic would be a ragged hierarchy if a few child products were added below 100-10 (Cola):

If the Database Manager maps the PRODUCT source column to level 0 of the Product dimension, as shown:

External column	Include in report	Essbase dimension	Gen/Lev filter
PRODUCT	Y	Product	Level0 [Level]

then the query filter condition will be predefined for the PRODUCT column:

```
SELECT "PRODUCT"
FROM <Query defined in Datasource>
WHERE  "PRODUCT" = <Level0> below <Grid context>
```

and the drill through result for PRODUCT will return all level 0 members below the Product member selected in the current Smart View cell intersection.

Learn more about how level 0 mapping works in Drill Through Use Case Example.

Drill Through Use Case Example

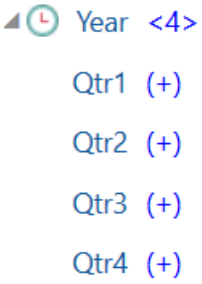
For this example use case, we will examine the following factors that you need to consider when designing drill through report access for Smart View users:

Copyright © 2019, 2026, Oracle and/or its affiliates.

- An Essbase cube for analysis
- An external source system for drill through
- A drill through report definition with column mapping defined by the database manager
- Drill through report results from Smart View

Essbase Cube

In this example, the basis is a cube similar to the Essbase demo cube, Sample Basic, but with only quarterly level data for the time dimension present in the cube (months were removed). Assume the Year dimension has Qtr1-Qtr4 as its lowest (level 0) members:



Though months are lacking in this outline hierarchy, the monthly data is available externally by drilling through to the source system (assume it is Oracle Database), and accessing information from a column named DIMENSION_YEAR:

DIMENSION_YEAR
Aug
Sep
Oct
Nov
Dec
Jan
Feb
Mar

For the remaining dimensions, assume they are the same as in the Essbase demo cube Sample Basic that is available in the gallery section of the Files catalog. A quick review of these is in order:

The Measures dimension tracks key performance indicators for accounts, using dynamic calculations with Essbase calc formulas.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Measures <3>		Label only
Profit <2> (+)	+ (Add)	Dynamic calcul...
Margin <2> (+)	+ (Add)	Dynamic calcul...
Sales (+)	+ (Add)	Store data
COGS (-)	- (Subtra...	Store data
Total Expenses <3> (-)	- (Subtra...	Dynamic calcul...
Inventory <4> (~)	~ (Ignore)	Dynamic calcul...
Ratios <3> (~)	~ (Ignore)	Label only

The Product dimension tracks the active inventory, descending two generations in depth, to generation 2 named Category (populated by 100 [alias Colas], 200, 300, 400, and Diet) and generation 3/level 0 named Product SKU (populated by 100-10 [alias Cola], 100-20, etc).

- Product <5> {Ca
  - 100 <3> (+)
    - 100-10 (+)
    - 100-20 (+)
    - 100-30 (+)
  - 200 <4> (+)
  - 300 <3> (+)
  - 400 <3> (+)
  - Diet <3> (~)

The Market dimension provides geographical separation with two additional generations below the dimension name. Generation 2 is Region (East, West, etc) and generation 3 is State.

Copyright © 2019, 2026, Oracle and/or its affiliates.

- Market <4> {Population}
  - East <5> (+)
    - New York (+)
    - Massachusetts (+)
    - Florida (+)
    - Connecticut (+)
    - New Hampshire (+)
  - West <5> (+)
  - South <4> (+)
  - Central <6> (+)

The Scenario dimension adds financial reporting analysis to the cube with its two stored and two Dynamic Calc members:

Scenario <4>		Label only
Actual (+)	+ (Add)	Store data
Budget (~)	~ (Ignore)	Store data
Variance (~)	~ (Ignore)	Dynamic calcul...
Variance % (~)	~ (Ignore)	Dynamic calcul...

External Source System

For the source system in this example, assume it is Oracle Database. The predefined Datasource in Essbase includes a SQL query that pulls information from a table in Oracle Database.

Our task as the database manager is to design a drill through report definition, based on this Datasource, that provides Smart View users the correct access to source system data pulled through the Datasource.

The query in the Datasource can be as simple as

```
SELECT * FROM TABLENAME
```

or it can be refined to pull any aggregation or assortment of the external data that you want to use as a basis.

The selection from our hypothetical table in Oracle Database includes external columns like those illustrated in our introduction. We will map some of these external columns to Essbase dimensions when we design the drill through report definition.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Column Mapping Definition

The column mapping in this example utilizes dimension mapping for Products, generation mapping for Year and Scenario, and level 0 mapping for Market.

External column	Include in report	Essbase dimension	Gen/Lev filter
DIMENSION_PRODUCT	Y	Product	None
DIMENSION_MARKET	Y	Market	Level0 [Level]
YEAR_PARENT	Y	Year	Quarter [Generation]
DIMENSION_SCENARIO	Y	Scenario	Scen [Generation]
SALES	Y	None	--
COGS	Y	None	--
MARKETING	Y	None	--
PAYROLL	Y	None	--
MISC	Y	None	--

Drill Through Report Examples by Column Mapping Type

The following drill through report examples illustrate the query results for each column mapping type that the database manager specifies as part of the drill through report definition.

Dimension Mapping Example 1

Using **dimension mapping** for Product with no hierarchical filter,

External column	Include in report	Essbase dimension	Gen/Lev filter
DIMENSION_PRODUCT	Y	Product	None

drill through performed from a cell intersection will be unbound to any specific generation or level.

Therefore, drilling through from (Year, Sales, West, Actual, Cola):

	A	B	C	D	E
1		Year	West	Actual	
2		Sales	COGS	Margin	Measures
3	Cola	14862	6059	8803	4593
4	Diet Cola	8923	5216	3707	-534
5	Caffeine Free Cola	4521	2892	1629	-510
6	Colas	28306	14167	14139	3549
7	Root Beer	34200	15144	19056	9727
8	Cream Soda	35391	15442	19949	10731
9	Fruit Soda	35034	18152	16882	5854
10	Diet Drinks	36423	17031	19392	8087
11	Product	132931	62905	70026	29861

Copyright © 2019, 2026, Oracle and/or its affiliates.

returns a drill through report filtered by the current grid context for Product, which happens to be 100-10 (100-10 is the Product SKU associated with the alias name Cola). All values pulled from the DIMENSION_PRODUCT column in the source system will be records where DIMENSION_PRODUCT = 100-10.

	A	B	C	D	E	F	G	H	I
1	DIMENSION_PROD	DIMENSION_MAF	YEAR_PAREN	DIMENSION_SCEN	SALES	COGS	MARKETI	PAYROLL	MISC
2	100-10	Utah	Qtr1	Actual	384	163	53	81	1
3	100-10	Utah	Qtr3	Actual	311	133	42	81	2
4	100-10	California	Qtr1	Actual	1998	799	278	153	2
5	100-10	California	Qtr3	Actual	2612	1044	364	153	0
6	100-10	Oregon	Qtr4	Actual	370	154	49	129	2
7	100-10	Washington	Qtr3	Actual	589	240	75	66	1
8	100-10	Nevada	Qtr3	Actual	259	114	42	99	2
9	100-10	California	Qtr4	Actual	1972	788	275	153	3
10	100-10	Oregon	Qtr1	Actual	464	194	63	129	1
11	100-10	Nevada	Qtr1	Actual	225	100	36	99	2
12	100-10	Nevada	Qtr4	Actual	239	106	38	99	1
13	100-10	Oregon	Qtr2	Actual	347	144	46	135	2
14	100-10	Washington	Qtr1	Actual	422	172	53	66	1
15	100-10	Utah	Qtr2	Actual	340	145	46	81	2
16	100-10	California	Qtr2	Actual	2358	942	328	159	1
17	100-10	Oregon	Qtr3	Actual	345	143	45	129	2
18	100-10	Washington	Qtr2	Actual	537	219	69	69	2
19	100-10	Washington	Qtr4	Actual	499	203	64	66	2
20	100-10	Utah	Qtr4	Actual	349	149	48	81	1
21	100-10	Nevada	Qtr2	Actual	242	107	39	99	0
22					Sum = 14862				
23									

To validate drill through reports as you test them, check that the sum for the measure in the report matches the cell intersection upon which drill through was performed. In the example above, the drill through report is validated, because the cell drilled upon matches the value (14862) of the sum of the mapped column in the drill through report.

The query Essbase uses to build the above drill through report is:

```
SELECT "DIMENSION_PRODUCT", "DIMENSION_MARKET", "YEAR_PARENT",
"DIMENSION_SCENARIO", "SALES", "COGS", "MARKETING", "PAYROLL", "MISC"
FROM <Query defined in Datasource>
WHERE (
"YEAR_PARENT" = 'Qtr3' OR
"YEAR_PARENT" = 'Qtr4' OR
"YEAR_PARENT" = 'Qtr1' OR
"YEAR_PARENT" = 'Qtr2')
AND
"DIMENSION_PRODUCT" = '100-10'
AND (
"DIMENSION_MARKET" = 'Oregon' OR
"DIMENSION_MARKET" = 'California' OR
"DIMENSION_MARKET" = 'Washington' OR
"DIMENSION_MARKET" = 'Utah' OR
"DIMENSION_MARKET" = 'Nevada')
AND
"DIMENSION_SCENARIO" = 'Actual'
```


Copyright © 2019, 2026, Oracle and/or its affiliates.

Dimension Mapping Example 2

Continuing from the previous example, let's explore what happens when drilling through on Product at a higher level.

Drilling through from (Year, Sales, West, Actual, Colas):

	A	B	C	D	E
1		Year	West	Actual	
2		Sales	COGS	Margin	Measures
3	Cola	14862	6059	8803	4593
4	Diet Cola	8923	5216	3707	-534
5	Caffeine Free Cola	4521	2892	1629	-510
6	Colas	28306	14167	14139	3549
7	Root Beer	34200	15144	19056	9727
8	Cream Soda	35391	15442	19949	10731
9	Fruit Soda	35034	18152	16882	5854
10	Diet Drinks	36423	17031	19392	8087
11	Product	132931	62905	70026	29861

returns a drill through report filtered by the current grid context for Product, which now happens to be 100 (100 is the Product category associated with the alias name Colas). All values pulled from the DIMENSION_PRODUCT column in the source system will be records where DIMENSION_PRODUCT = 100.

	A	B	C	D	E	F	G	H	I
1	DIMENSION_PRO	DIMENSION_MAI	YEAR_PAREN	DIMENSION_SCEN	SALES	COGS	MARKETING	PAYROLL	MISC
2	100	Utah	Qtr1	Actual	1454	646	218	243	5
3	100	Utah	Qtr3	Actual	1168	520	174	243	7
4	100	California	Qtr1	Actual	2767	1553	520	348	5
5	100	California	Qtr3	Actual	3401	2070	696	348	1
6	100	Oregon	Qtr4	Actual	1051	434	224	282	5
7	100	Washington	Qtr3	Actual	1426	590	391	153	4
8	100	Nevada	Qtr3	Actual	496	222	74	162	4
9	100	California	Qtr4	Actual	2767	1701	570	348	6
10	100	Oregon	Qtr1	Actual	1257	521	265	282	4
11	100	Nevada	Qtr1	Actual	413	184	60	162	3
12	100	Nevada	Qtr4	Actual	440	197	64	162	2
13	100	Oregon	Qtr2	Actual	1010	416	219	291	2
14	100	Washington	Qtr1	Actual	1059	438	294	153	4
15	100	Utah	Qtr2	Actual	1317	587	197	243	3
16	100	California	Qtr2	Actual	3161	1919	645	363	4
17	100	Oregon	Qtr3	Actual	932	382	194	282	7
18	100	Washington	Qtr2	Actual	1249	516	338	156	4
19	100	Washington	Qtr4	Actual	1203	498	331	153	4
20	100	Utah	Qtr4	Actual	1294	575	194	243	3
21	100	Nevada	Qtr2	Actual	441	198	65	162	1
22									
23					Sum = 23806				

The drill through report is validated, because the cell drilled upon matches the value (23806) of the sum of the mapped column in the drill through report.

Copyright © 2019, 2026, Oracle and/or its affiliates.

The query Essbase uses to build the above drill through report is:

```
SELECT "DIMENSION_PRODUCT", "DIMENSION_MARKET", "YEAR_PARENT",
"DIMENSION_SCENARIO", "SALES", "COGS", "MARKETING", "PAYROLL", "MISC"
FROM <Query defined in Datasource>
WHERE (
"YEAR_PARENT" = 'Qtr3' OR
"YEAR_PARENT" = 'Qtr4' OR
"YEAR_PARENT" = 'Qtr1' OR
"YEAR_PARENT" = 'Qtr2')
AND
"DIMENSION_PRODUCT" = '100'
AND (
"DIMENSION_MARKET" = 'Oregon' OR
"DIMENSION_MARKET" = 'California' OR
"DIMENSION_MARKET" = 'Washington' OR
"DIMENSION_MARKET" = 'Utah' OR
"DIMENSION_MARKET" = 'Nevada')
AND
"DIMENSION_SCENARIO" = 'Actual'
```

Generation Mapping Example 1

Using the **generation mapping** for Year with a filter on the generation named Quarter,

External column	Include in report	Essbase dimension	Gen/Lev filter
YEAR_PARENT	Y	Year	Quarter [Generation]

drill through performed from a cell intersection will be bound to the specified generation of Year.

Drilling through from (Qtr2, Sales, Market, Actual, Cola):

	A	B	C	D	E
1		Qtr2	Market	Actual	
2		Sales	COGS	Margin	Measures
3	Cola	16048	6136	9912	5892
4	Diet Cola	7957	3871	4086	1534
5	Caffeine Free Cola	3182	1606	1576	446
6	Colas	27187	11613	15574	7872
7	Root Beer	27401	12194	15207	7030
8	Cream Soda	25736	11649	14087	6769
9	Fruit Soda	21355	9906	11449	5436
10	Diet Drinks	26787	11967	14820	7336
11	Product	101679	45362	56317	27107

returns a drill through report filtered by the mapped generation context for Year, which is Quarters. Because Qtr2 is in the selected grid context, all values pulled from the YEAR_PARENT column in the source system will be records where YEAR_PARENT = Qtr2.

Copyright © 2019, 2026, Oracle and/or its affiliates.

	A	B	C	D	E	F	G	H	I
1	DIMENSION_PRC	DIMENSION_MAR	YEAR_PARE	DIMENSION_SCE	SALES	COGS	MARKETING	PAYROLL	MISC
2	100-10	Iowa	Qtr2	Actual	199	91	26	63	1
3	100-10	Ohio	Qtr2	Actual	303	123	38	69	3
4	100-10	Connecticut	Qtr2	Actual	799	318	104	93	0
5	100-10	Oregon	Qtr2	Actual	347	144	46	135	2
6	100-10	Texas	Qtr2	Actual	1500	688	211	63	2
7	100-10	Missouri	Qtr2	Actual	520	233	87	99	2
8	100-10	New Mexico	Qtr2	Actual	413	164	53	93	2
9	100-10	New Hampshire	Qtr2	Actual	413	164	53	93	3
10	100-10	Colorado	Qtr2	Actual	558	244	79	36	0
11	100-10	New York	Qtr2	Actual	2358	942	328	159	1
12	100-10	Louisiana	Qtr2	Actual	292	118	32	33	3
13	100-10	Washington	Qtr2	Actual	537	219	69	69	2
14	100-10	Wisconsin	Qtr2	Actual	712	297	269	87	1
15	100-10	Massachusetts	Qtr2	Actual	1719	186	60	93	2
16	100-10	Nevada	Qtr2	Actual	242	107	39	99	0
17	100-10	California	Qtr2	Actual	2358	942	328	159	1
18	100-10	Illinois	Qtr2	Actual	1399	586	193	135	1
19	100-10	Florida	Qtr2	Actual	735	293	96	93	3
20	100-10	Utah	Qtr2	Actual	340	145	46	81	2
21	100-10	Oklahoma	Qtr2	Actual	304	132	43	36	1
22									
23									

Sum = 16048

The drill through report is validated, because the cell drilled upon matches the value (16048) of the sum of the mapped column in the drill through report.

The query Essbase uses to build the above drill through report is:

```
SELECT "DIMENSION_PRODUCT", "DIMENSION_MARKET", "YEAR_PARENT",
"DIMENSION_SCENARIO", "SALES", "COGS", "MARKETING", "PAYROLL", "MISC"
FROM <Query defined in Datasource>
WHERE
"YEAR_PARENT" = 'Qtr2'
AND
"DIMENSION_PRODUCT" = '100-10'
AND (
"DIMENSION_MARKET" = 'Oregon' OR
"DIMENSION_MARKET" = 'New York' OR
"DIMENSION_MARKET" = 'Oklahoma' OR
"DIMENSION_MARKET" = 'California' OR
"DIMENSION_MARKET" = 'Florida' OR
"DIMENSION_MARKET" = 'Washington' OR
"DIMENSION_MARKET" = 'Utah' OR
"DIMENSION_MARKET" = 'Iowa' OR
"DIMENSION_MARKET" = 'New Mexico' OR
"DIMENSION_MARKET" = 'Massachusetts' OR
"DIMENSION_MARKET" = 'Texas' OR
"DIMENSION_MARKET" = 'Illinois' OR
"DIMENSION_MARKET" = 'Colorado' OR
"DIMENSION_MARKET" = 'Connecticut' OR
"DIMENSION_MARKET" = 'New Hampshire' OR
"DIMENSION_MARKET" = 'Missouri' OR
"DIMENSION_MARKET" = 'Louisiana' OR
"DIMENSION_MARKET" = 'Ohio' OR
```


Copyright © 2019, 2026, Oracle and/or its affiliates.

```
"DIMENSION_MARKET" = 'Wisconsin' OR
"DIMENSION_MARKET" = 'Nevada')
AND
"DIMENSION_SCENARIO" = 'Actual'
```

Generation Mapping Example 2

Continuing from the previous example, let's explore what happens when drilling through on Year dimension at a higher level.

Drilling through from (Year, Sales, Market, Actual, Cola):

	A	B	C	D	E
1		Year	Market	Actual	
2		Sales	COGS	Margin	Measures
3	Cola	62824	24198	38626	22777
4	Diet Cola	30469	14784	15685	5708
5	Caffeine Free Cola	12841	6366	6475	1983
6	Colas	106134	45348	60786	30468
7	Root Beer	109086	48500	60586	27954
8	Cream Soda	101405	46405	55000	25799
9	Fruit Soda	84230	39083	45147	21301
10	Diet Drinks	105678	47136	58542	28826
11	Product	400855	179336	221519	105522

returns a drill through report filtered by the mapped Quarters generation of Year. Values pulled from the YEAR_PARENT column in the source system will be Qtr1, Qtr2, Qtr3, and Qtr4.

Copyright © 2019, 2026, Oracle and/or its affiliates.

	A	B	C	D	E	F	G	H	I
1	DIMENSION_PRO	DIMENSION_MAR	YEAR_PARE	DIMENSION_SCE	SALES	COGS	MARKETING	PAYROLL	MISC
2	100-10	Utah	Qtr1	Actual	384	163	53	81	1
3	100-10	Utah	Qtr3	Actual	311	133	42	81	2
4	100-10	Iowa	Qtr1	Actual	188	84	24	63	0
5	100-10	Colorado	Qtr2	Actual	558	244	79	36	0
6	100-10	Ohio	Qtr3	Actual	277	111	33	66	1
7	100-10	Ohio	Qtr4	Actual	322	130	40	66	2
8	100-10	New York	Qtr1	Actual	1998	799	278	153	2
9	100-10	Massachusetts	Qtr3	Actual	1905	164	53	93	3
10	100-10	Florida	Qtr3	Actual	821	327	106	93	1
11	100-10	Connecticut	Qtr2	Actual	799	318	104	93	0
12	100-10	Connecticut	Qtr3	Actual	708	283	91	93	0
13	100-10	Connecticut	Qtr4	Actual	927	370	120	93	2
14	100-10	Missouri	Qtr4	Actual	514	229	86	99	1
15	100-10	Iowa	Qtr2	Actual	199	91	26	63	1
16	100-10	Iowa	Qtr4	Actual	201	91	26	63	1
17	100-10	California	Qtr1	Actual	1998	799	278	153	2
18	100-10	Massachusetts	Qtr2	Actual	1719	186	60	93	2
19	100-10	Louisiana	Qtr2	Actual	292	118	32	33	3
20	100-10	Louisiana	Qtr3	Actual	336	136	37	33	2
21	100-10	California	Qtr3	Actual	2612	1044	364	153	0
22	100-10	Oregon	Qtr4	Actual	370	154	49	129	2
23	100-10	Washington	Qtr3	Actual	589	240	75	66	1
24	100-10	Texas	Qtr1	Actual	1384	634	196	63	2
25	100-10	Colorado	Qtr4	Actual	281	122	39	36	1
26	100-10	New Hampshire	Qtr2	Actual	413	164	53	93	3
27	100-10	Illinois	Qtr3	Actual	1421	596	195	129	1
28	100-10	Illinois	Qtr4	Actual	1313	[illegible]	[illegible]	129	0
29	100-10	Ohio	Qtr1	Actual	389			66	1

Sum = 62824

The drill through report is validated, because the cell drilled upon matches the value (62824) of the sum of the mapped column in the drill through report.

The query Essbase uses to build the above drill through report is:

```
SELECT "DIMENSION_PRODUCT", "DIMENSION_MARKET", "YEAR_PARENT",
"DIMENSION_SCENARIO", "SALES", "COGS", "MARKETING", "PAYROLL", "MISC"
FROM <Query defined in Datasource>
WHERE (
"YEAR_PARENT" = 'Qtr3' OR
"YEAR_PARENT" = 'Qtr4' OR
"YEAR_PARENT" = 'Qtr1' OR
"YEAR_PARENT" = 'Qtr2')
AND
"DIMENSION_PRODUCT" = '100-10'
AND (
"DIMENSION_MARKET" = 'Oregon' OR
"DIMENSION_MARKET" = 'New York' OR
"DIMENSION_MARKET" = 'Oklahoma' OR
"DIMENSION_MARKET" = 'California' OR
"DIMENSION_MARKET" = 'Florida' OR
"DIMENSION_MARKET" = 'Washington' OR
"DIMENSION_MARKET" = 'Utah' OR
"DIMENSION_MARKET" = 'Iowa' OR
"DIMENSION_MARKET" = 'New Mexico' OR
```


Copyright © 2019, 2026, Oracle and/or its affiliates.

```
"DIMENSION_MARKET" = 'Massachusetts' OR
"DIMENSION_MARKET" = 'Texas' OR
"DIMENSION_MARKET" = 'Illinois' OR
"DIMENSION_MARKET" = 'Colorado' OR
"DIMENSION_MARKET" = 'Connecticut' OR
"DIMENSION_MARKET" = 'New Hampshire' OR
"DIMENSION_MARKET" = 'Missouri' OR
"DIMENSION_MARKET" = 'Louisiana' OR
"DIMENSION_MARKET" = 'Ohio' OR
"DIMENSION_MARKET" = 'Wisconsin' OR
"DIMENSION_MARKET" = 'Nevada')
AND
"DIMENSION_SCENARIO" = 'Actual'
```

Level 0 Mapping Example

Using the **level 0 mapping** for Market with a filter on all level 0 (leaf) members,

External column	Include in report	Essbase dimension	Gen/Lev filter
DIMENSION_MARKET	Y	Market	Level0 [Level]

drill through performed from a cell intersection will return all the lowest members for Market.

Drilling through from (Year, Sales, Market, Actual, Cola):

	A	B	C	D	E
1		Year	Market	Actual	
2		Sales	COGS	Margin	Measures
3	Cola	62824	24198	38626	22777
4	Diet Cola	30469	14784	15685	5708
5	Caffeine Free Cola	12841	6366	6475	1983
6	Colas	106134	45348	60786	30468
7	Root Beer	109086	48500	60586	27954
8	Cream Soda	101405	46405	55000	25799
9	Fruit Soda	84230	39083	45147	21301
10	Diet Drinks	105678	47136	58542	28826
11	Product	400855	179336	221519	105522

returns a drill through report that includes the lowest level members of Market dimension, which happen to be States. All values pulled from the DIMENSION_MARKET column in the source system will be records containing States.

Copyright © 2019, 2026, Oracle and/or its affiliates.

	A	B	C	D	E	F	G	H	I
1	DIMENSION_PR	DIMENSION_MARKET	YEAR_PARE	DIMENSIC	SALES	COGS	MARKETING	PAYROLL	MISC
2	100-10	Utah	Qtr1	Actual	384	163	53	81	1
3	100-10	Utah	Qtr3	Actual	311	133	42	81	2
4	100-10	Iowa	Qtr1	Actual	188	84	24	63	0
5	100-10	Colorado	Qtr2	Actual	558	244	79	36	0
6	100-10	Ohio	Qtr3	Actual	277	111	33	66	1
7	100-10	Ohio	Qtr4	Actual	322	130	40	66	2
8	100-10	New York	Qtr1	Actual	1998	799	278	153	2
9	100-10	Massachusetts	Qtr3	Actual	1905	164	53	93	3
10	100-10	Florida	Qtr3	Actual	821	327	106	93	1
11	100-10	Connecticut	Qtr2	Actual	799	318	104	93	0
12	100-10	Connecticut	Qtr3	Actual	708	283	91	93	0
13	100-10	Connecticut	Qtr4	Actual	927	370	120	93	2
14	100-10	Missouri	Qtr4	Actual	514	229	86	99	1
15	100-10	Iowa	Qtr2	Actual	199	91	26	63	1
16	100-10	Iowa	Qtr4	Actual	201	91	26	63	1
17	100-10	California	Qtr1	Actual	1998	799	278	153	2
18	100-10	Massachusetts	Qtr2	Actual	1719	186	60	93	2
19	100-10	Louisiana	Qtr2	Actual	292	118	32	33	3
20	100-10	Louisiana	Qtr3	Actual	336	136	37	33	2
21	100-10	California	Qtr3	Actual	2612	1044	364	153	0
22	100-10	Oregon	Qtr4	Actual	370	154	49	129	2
23	100-10	Washington	Qtr3	Actual	589	240	75	66	1
24	100-10	Texas	Qtr1	Actual	1384	634	196	63	2
25	100-10	Colorado	Qtr4	Actual	281	122	39	36	1
26	100-10	New Hampshire	Qtr2	Actual	413	164	53	93	3
27	100-10	Illinois	Qtr3	Actual	1421	596	195	129	1
28	100-10	Illinois	Qtr4	Actual	1313	551	180	129	0
29	100-10	Ohio	Qtr1	Actual	389	Sum = 62824		66	1
30	100-10	New York	Qtr3	Actual	2612	1044	364	153	1

The drill through report is validated, because the cell drilled upon matches the value (62824) of the sum of the mapped column in the drill through report.

The query Essbase uses to build the above drill through report is:

```
SELECT "DIMENSION_PRODUCT", "DIMENSION_MARKET", "YEAR_PARENT",
"DIMENSION_SCENARIO", "SALES", "COGS", "MARKETING", "PAYROLL", "MISC"
FROM <Query defined in Datasource>
WHERE (
"YEAR_PARENT" = 'Qtr3' OR
"YEAR_PARENT" = 'Qtr4' OR
"YEAR_PARENT" = 'Qtr1' OR
"YEAR_PARENT" = 'Qtr2')
AND
"DIMENSION_PRODUCT" = '100-10'
AND (
"DIMENSION_MARKET" = 'Oregon' OR
"DIMENSION_MARKET" = 'New York' OR
"DIMENSION_MARKET" = 'Oklahoma' OR
"DIMENSION_MARKET" = 'California' OR
"DIMENSION_MARKET" = 'Florida' OR
"DIMENSION_MARKET" = 'Washington' OR
"DIMENSION_MARKET" = 'Utah' OR
```


Copyright © 2019, 2026, Oracle and/or its affiliates.

```
"DIMENSION_MARKET" = 'Iowa' OR
"DIMENSION_MARKET" = 'New Mexico' OR
"DIMENSION_MARKET" = 'Massachusetts' OR
"DIMENSION_MARKET" = 'Texas' OR
"DIMENSION_MARKET" = 'Illinois' OR
"DIMENSION_MARKET" = 'Colorado' OR
"DIMENSION_MARKET" = 'Connecticut' OR
"DIMENSION_MARKET" = 'New Hampshire' OR
"DIMENSION_MARKET" = 'Missouri' OR
"DIMENSION_MARKET" = 'Louisiana' OR
"DIMENSION_MARKET" = 'Ohio' OR
"DIMENSION_MARKET" = 'Wisconsin' OR
"DIMENSION_MARKET" = 'Nevada')
AND
"DIMENSION_SCENARIO" = 'Actual'
```

Access to Drill Through Reports

How you work with Essbase drill through reports depends on your level of access.

A user role of Database Manager is required to create drill through report definitions on a cube. If the drill through report definition is based on one or more Datasources defined at the application level, a prerequisite assumption is that a connection and Datasource were already defined at the application level, by at least an Application Manager.

The Application Manager who creates the connection and Datasource must additionally have appropriate credentials to access the external source system; for example, if the external source data is a SQL source, the Application Manager must have credentials to log in to the SQL source, in order to create the connection.

Power User is the minimum permission to create the application and cube in the first place. A Power User has implicit Application Manager permission for the applications they created, but not for all applications.

Any user with Database Access can access the drill through report, as long as the user's filter does not restrict access to the cells within the drillable region defined for the drill through report. A drillable region is a specification that indicates the cell intersections from which the drill through report is accessible from Smart View.

Design Drill Through Reports

If you are an Essbase Database Manager or higher, you can create drill through report definitions to enable user access to more data that is stored in an external, relational source. Design factors include column mappings, drillable regions, and optionally, parameterization, if you want to add the flexibility of variables.

- General Considerations for Designing Drill Through Reports
- Define Column Mappings for Drill Through Reports
- Define Drillable Regions for Drill Through Reports
- Implement Parameters for Drill Through Reports

Copyright © 2019, 2026, Oracle and/or its affiliates.

General Considerations for Designing Drill Through Reports

Drill through reports pull data from an external source system or a file, or link to web applications. To create a drill through report definition, you must be a Database Manager or higher. Create it in the the database's Scripts section, in the Essbase web interface.

Every drill through report definition must include a column mapping and a drillable region. Optionally, you can define a mapping for runtime parameters, if there is a parameterized query in the Datasource that supplies external source data connectivity for your drill through reports.

You can create drill through reports that pull data from an external source system (RDBMS) or from a file.

> **Note**
>
> If the Datasource for the drill through report is connected to Oracle Database, you can check the **Use Temporary Tables** option to improve performance for queries that have a large number of values in the SQL “IN” clause.

You can also design drill through reports to access Web URLs.

Define Column Mappings for Drill Through Reports

The column mapping part of an Essbase drill through report definition defines which external source columns should be included in the report, which Essbase dimensions those columns map to, and (optionally) a generation/level filter condition indicating how much depth of access to provide.

To understand column mappings in depth, see Drill Through Report Definition.

To define column mappings,

- Redwood
- Classic

Redwood

1. On the Home page, open the application and open the cube.
2. Select **Scripts** in the left hand panel.
3. Click **Drill Through Reports**.
4. Click **Create** and select **Datasource** from the menu.
5. In the **Name** field, add a name for the report.
6. In the **Datasource** field, select the Datasource you want to use.
7. Create column mappings:
   a. Select the columns from the external Datasource that you want to include in the report.
   b. Select the dimension to which you want to map each column.

Copyright © 2019, 2026, Oracle and/or its affiliates.

c. Select a filter conditions for the mappings; for example, Level 0, Generation, or None (for dimension mapping).

Column Name	☐ Report Columns	Dimension/Generation (Filter Condition)	
DIMENSION_PRODUCT	☑	Product	Product SKU [Generation]
DIMENSION_MARKET	☑	Market	State [Generation]
DIMENSION_YEAR	☑	Year	None
DIMENSION_SCENARIO	☑	Scenario	Level0 [Level]
SALES	☑	None	
COGS	☐	None	
MARKETING	☐	None	

d. When you are finished, click **Drillable Regions**. You cannot save the drill through report definition without defining a region, as described in Define Drillable Regions for Drill Through Reports.

Classic

1. On the Applications page, expand the application.
2. From the **Actions** menu, to the right of the cube name, click **Inspect**.
3. In the application inspector, select the **Scripts** tab.
4. Select **Drill Through Reports**.
5. Click **Create** and select **Datasource** from the drop down menu.
6. In the **Name** field, add a name for the report.
7. In the **Datasource** field, select the Datasource you want to use.
8. Create column mappings:
   a. Select the columns from the external Datasource that you want to include in the report.
   b. Select the dimension to which you want to map each column.
   c. Select a filter conditions for the mappings; for example, Level 0, Generation, or None (for dimension mapping).

9. When you are finished, click **Drillable Regions**. You cannot save the drill through report definition without defining a region, as described in Define Drillable Regions for Drill Through Reports.

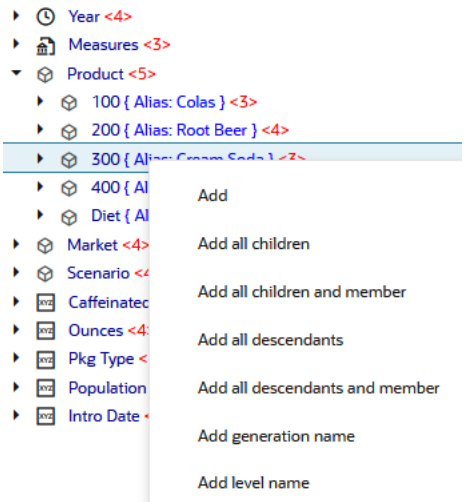
Copyright © 2019, 2026, Oracle and/or its affiliates.

Define Drillable Regions for Drill Through Reports

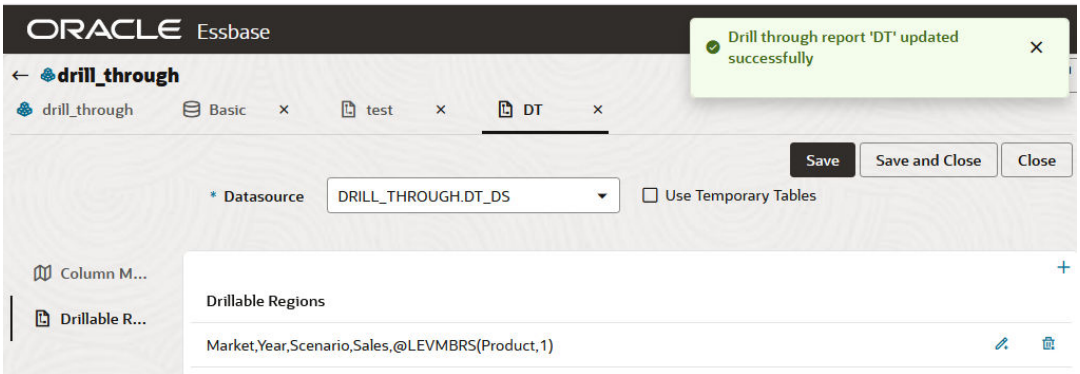
The "drillable region" part of an Essbase drill through report definition is where you specify one or more areas / data intersections in the Essbase cube from which you want to provide drill through access to additional detail that is available in the external source of data.

To specify the drillable region using the Essbase web interface,

1. After you complete the column mappings of your drill through report, click **Drillable Regions**.
2. Click the plus sign and choose one of the following methods:
   - Double click in the empty row and define the region using calculation syntax: member names and member set functions.
   - Click the pencil icon to open the **Drillable Region Selector**, select the member you want to add, right click, and choose from the menu options to define the region.



3. Click **Save**. A message is displayed indicating if the drill through report was updated successfully.



Drillable Region Examples

The following examples show drillable regions in Smart View, highlighted in blue.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Example 1

To define a drillable region at the top of the Market, Year, and Scenario dimensions, the member Sales, and all level 1 members of the product dimension, use:

Market,Year,Scenario,Sales,@LEVMBRS(Product,1)

When you drill into a Smart View sheet, the resulting grid looks like this:

		Market	Scenario
		Sales	COGS
Colas	Year	106134	45348
Old Fashioned	Year	41537	18995
Diet Root Beer	Year	38240	16659
Sasparilla	Year	17559	7647
Birch Beer	Year	11750	5199
Root Beer	Year	109086	48500
Dark Cream	Year	46956	20747
Vanilla Cream	Year	17480	9965
Diet Cream	Year	36969	15693
Dark Cream	Year	46956	20747
Vanilla Cream	Year	17480	9965
Diet Cream	Year	36969	15693
Cream Soda	Year	101405	46405
Grape	Year	35799	15267
Orange	Year	32670	14277
Strawberry	Year	15761	9539
Fruit Soda	Year	84230	39083

Example 2

To define a drillable region for the descendants of Market, use:

@DESCENDANTS(Market)

When you drill into a Smart View sheet, the resulting grid looks like this:

Copyright © 2019, 2026, Oracle and/or its affiliates.

			Scenario
			Measures
Colas	East	Year	12656
Colas	West	Year	3549
Colas	South	Year	4773
Colas	Central	Year	9490
Colas	Market	Year	30468
Root Beer	East	Year	2534
Root Beer	West	Year	9727
Root Beer	South	Year	6115
Root Beer	Central	Year	9578
Root Beer	Market	Year	27954
Cream Soda	East	Year	2627
Cream Soda	West	Year	10731
Cream Soda	South	Year	2350
Cream Soda	Central	Year	10091
Cream Soda	Market	Year	25799
Fruit Soda	East	Year	6344
Fruit Soda	West	Year	5854
Fruit Soda	South	Year	#Missing
Fruit Soda	Central	Year	9103
Fruit Soda	Market	Year	21301

Example 3

To define a drillable region for the members of generation 3 in the Product dimension, at the top of the Market, Year, Scenario dimensions and the member, Sales, use:

Market,Year,Scenario,Sales,@GENMBRS(Product,3)

When you drill into a Smart View sheet, the resulting grid looks like this:

Copyright © 2019, 2026, Oracle and/or its affiliates.

		Market	Scenario
		Sales	COGS
Cola	Year	62824	24198
Diet Cola	Year	30469	14784
Caffeine Free Cola	Year	12841	6366
Colas	Year	106134	45348
Old Fashioned	Year	41537	18995
Diet Root Beer	Year	38240	16659
Sasparilla	Year	17559	7647
Birch Beer	Year	11750	5199
Root Beer	Year	109086	48500
Dark Cream	Year	46956	20747
Vanilla Cream	Year	17480	9965
Diet Cream	Year	36969	15693
Cream Soda	Year	101405	46405
Grape	Year	35799	15267
Orange	Year	32670	14277
Strawberry	Year	15761	9539
Fruit Soda	Year	84230	39083
Shared Diet Cola	Year	30469	14784
Diet Root Beer	Year	38240	16659
Diet Cream	Year	36969	15693

Implement Parameters for Drill Through Reports

To make drill through reports to external source data more flexible, you can implement parameters, enabling Essbase to dynamically build drill through reports based on variables you pass into the SQL query.

If you are a Database Manager or higher, you can implement parameterized Datasource queries in drill through reports.

Prerequisite: an Application Manager or higher has provided you access to a Datasource that uses a fixed (default) value, a substitution variable, or an external user-defined function as a parameter.

Essbase discerns the value of parameters at runtime, inserting their current value into the query that Essbase generates when a drill through is performed.

In the drill through report definition, you can optionally add dimension, generation, and level mapping for runtime parameters. This enables you to further customize the results of drill through reports based on the current variable context.

If you want Essbase to dynamically build drill through reports based on variables, follow this workflow:

1. Implement parameters in the underlying Datasource query. This must be done by an Application Manager or higher.
2. Create a drill through report definition associated with the Datasource.
3. Optionally, provide runtime customizations to parameter use, within the drill through report definition. An example is provided below.

Copyright © 2019, 2026, Oracle and/or its affiliates.

4. Test and validate expected behavior by running drill through reports and checking the platform log.

Runtime Parameters and User Defined Functions

The following example use cases are based on prerequisite assumptions:

- A user defined function, **getMonths**, is defined in the external source system. The function returns a comma-separated list of months.
- The Datasource used for the drill through report definition is defined using a query that calls the **getMonths** function, as follows:

```
select * from SampleBasic where month in (getMonths(?))
```

Level 0 (Recursive) Mapping of Runtime Parameter

In the drill through report definition, when the runtime parameter is bound to Year at Level 0,

then if a Smart View user drills through on Qtr1 of the Year dimension, the drill through report will include Jan, Feb, Mar. If a Smart View user drills through on Year, the drill through report will include Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec.

Generation Mapping of Runtime Parameter

In the drill through report definition, when the runtime parameter is bound to Year at the Quarter generation,

then if a Smart View user drills through on Year dimension member, the drill through report will include Qtr1, Qtr2, Qtr3, Qtr4.

> **Note**
>
> It is invalid to set a runtime parameter's Dimension/Generation Binding that overlaps with a column mapping. The generated query in this case is always a null set.

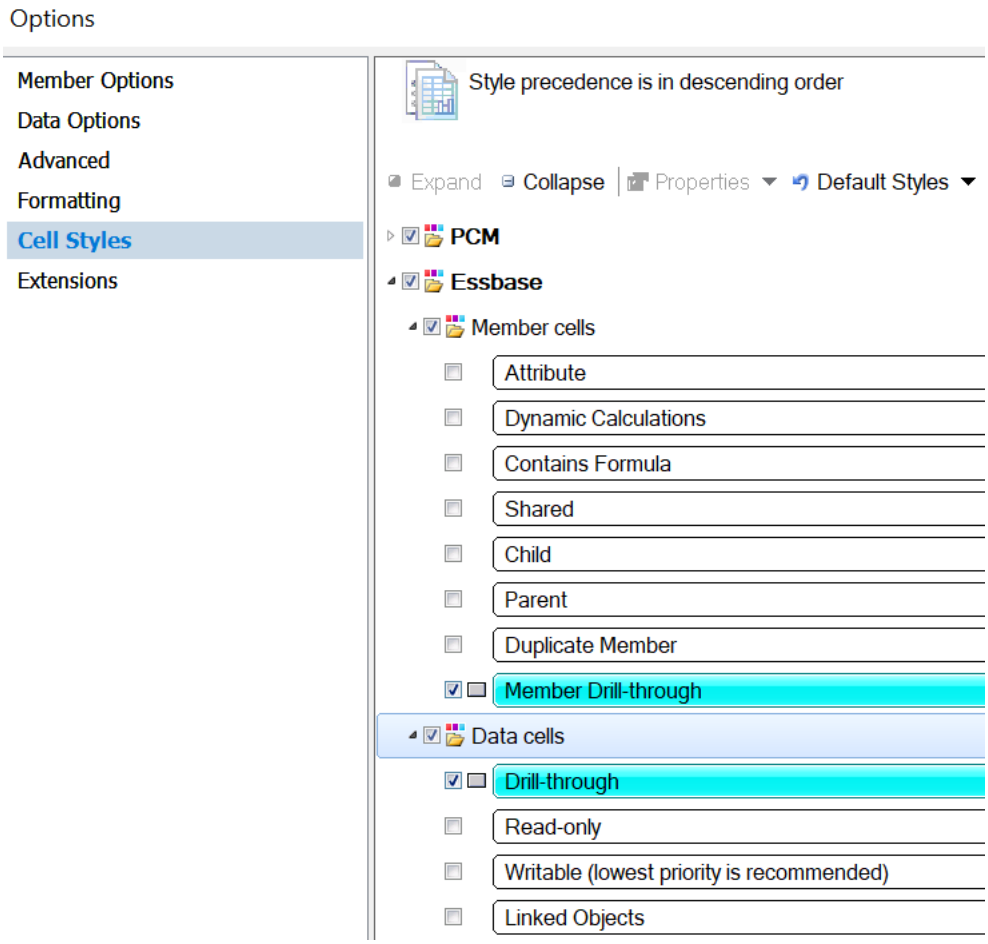
Copyright © 2019, 2026, Oracle and/or its affiliates.

Test Drill Through Reports

To test drill through reports, prepare Smart View, perform drill through operations, check the output, and check the platform log if you do not get the results you expect.

Prepare Smart View

1. Install the latest Smart View version.
2. Connect to the cube.
3. Enable cell styles to show drillable regions in the Smart View sheet.
   a. On the Smart View ribbon, click **Options**.
   b. Under **Formatting**, ensure that **Use Cell Styles** is selected.
   c. Expand **Essbase**, then **Member Cells**. Check **Member Drill-through**, then right click it and choose a style (for example, a blue background).
   d. Expand **Data Cells**. Check **Drill-through**, then right click it and choose the same style.



Drill Through to an External Source of Data

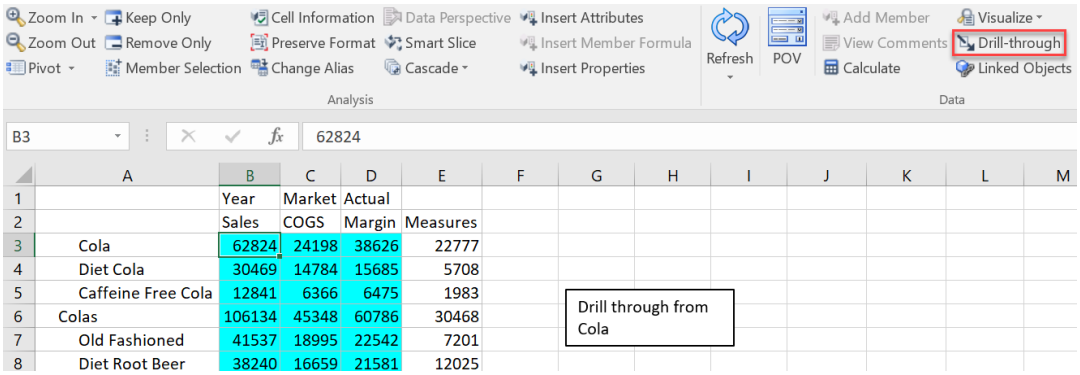
Drill through from one or more drillable cells to your external source of data.

1. Connect to your cube in Smart View.

Copyright © 2019, 2026, Oracle and/or its affiliates.

2. Drill down into the drillable regions you specified in your drill through report definition. Drillable cells are highlighted using the cell style you selected when preparing Smart View.

3. Select one or more cells and click **Drill-through** on the Essbase ribbon.
The image shows drill through from a single cell. You can also drill through from multiple cells. The cells can include a range or multiple ranges. They can be contiguous or non-contiguous, and can include different generations in the hierarchy.



	A	B	C	D	E
1		Year	Market	Actual	
2		Sales	COGS	Margin	Measures
3	Cola	62824	24198	38626	22777
4	Diet Cola	30469	14784	15685	5708
5	Caffeine Free Cola	12841	6366	6475	1983
6	Colas	106134	45348	60786	30468
7	Old Fashioned	41537	18995	22542	7201
8	Diet Root Beer	38240	16659	21581	12025

4. View the results.
When you drill through, a new sheet is opened showing the results of the drill through operation. Results depend upon the context of the cell or cells from which you drill through. See How Drill Through Works.

DIMENSION_PRODUCT	DIMENSION_MARKET	YEAR_PARENT	DIMENSION_SCENARIO	SALES	COGS	MARKETING	PAYROLL	MISC
100-10	Utah	Qtr1	Actual	384	163	53	81	1
100-10	Utah	Qtr3	Actual	311	133	42	81	2
100-10	Iowa	Qtr1	Actual	188	84	24	63	0
100-10	Colorado	Qtr2	Actual	558	244	79	36	0
100-10	Ohio	Qtr3	Actual	277	111	33	66	1
100-10	Ohio	Qtr4	Actual	322	130	40	66	2
100-10	New York	Qtr1	Actual	1998	799	278	153	2
100-10	Massachusetts	Qtr3	Actual	1905	164	53	93	3
100-10	Florida	Qtr3	Actual	821	327	106	93	1
100-10	Connecticut	Qtr2	Actual	799	318	104	93	0
100-10	Connecticut	Qtr3	Actual	708	283	91	93	0
100-10	Connecticut	Qtr4	Actual	927	370	120	93	2

Check the Drill Through Report Outputs and the Platform Log

Check drill through report outputs by performing drill through operations and analyzing the results.

If you are not seeing the results you expect, check the Debug Drill Through using Essbase Server Platform Log to debug your reports.

> **Note**
>
> If a drill through report fails and you find an error, `'ERROR: relation <member name> does not exist'` in the platform log, see Expand Limit for SQL IN Clauses in Drill Through Reports.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Drill Through to a URL

You can create a drill through report that enables Essbase users to drill through to a URL directly from a cell in a Smart View query worksheet.

> **Note**
>
> You can only drill through to a URL from a single cell. Drilling through to a URL from multiple cells is not supported.

Drill through target URLs can be static or dynamic. An example of a static target URL is https://docs.oracle.com.

Dynamic target URLs use variables. All dynamic target URLs have the following variable structure in common:

```
$$<dimension-name>-VALUE$$
```

When a user drills through from a cell, Essbase makes the substitutions necessary to generate the target URL in the context of the selected drill-through intersection.

For example, if the drillable region includes a member from the market dimension, the variable for the value from Market will look like this:

```
$$Market-VALUE$$
```

When you select the drill-through cell from an intersection containing a member from the Market dimension, Essbase substitutes the appropriate Market value into the URL syntax; for example:

East

In the following example, the variable for the value from the Product dimension will look like this:

```
$$Product-VALUE$$
```

When you select the drill-through cell from an intersection containing a member from the Product dimension, Essbase substitutes the appropriate Product value in the URL syntax; for example:

Cola

Create a Drill Through Report to a URL

To create a drill through URL report, define a new drill through report on Sample Basic.

- Redwood
- Classic

Copyright © 2019, 2026, Oracle and/or its affiliates.

Redwood

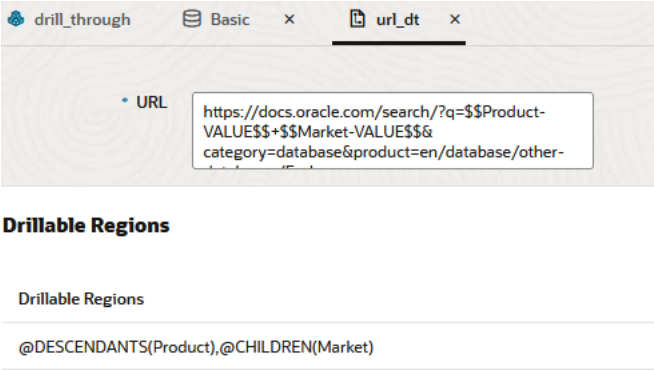
1. On the Home page, open the application and open the cube.
2. Select **Scripts** in the left hand panel.
3. Click **Drill Through Reports**.
4. Click **Create**, and then select **URL** from the drop down menu.
5. Give the drill through report a name, such as URL_dt.
6. In the **URL** field, add the following URL:

   ```
   https://docs.oracle.com/search/?q=$$Product-VALUE$$+$$Market-
   VALUE$$&category=database&product=en/database/other-databases/Essbase
   ```

   > **Note**
   >
   > The variable syntax comes right after the ?q=

7. Add a new drillable region:

   ```
   @DESCENDANTS(Product),@CHILDREN(Market)
   ```



Classic

1. On the Applications page, expand the application.
2. From the **Actions** menu, to the right of the cube name, click **Inspect**.
3. On the cube inspector, select the **Scripts** tab and then select **Drill Through Reports**.
4. Click **Create**, and then select **URL** from the drop down menu.
5. Give the drill through report a name, such as URL_dt.
6. In the **URL** field, add the following URL:

   ```
   https://docs.oracle.com/search/?q=$$Product-VALUE$$+$$Market-
   VALUE$$&category=database&product=en/database/other-databases/Essbase
   ```


Copyright © 2019, 2026, Oracle and/or its affiliates.

> **Note**
>
> The variable syntax comes right after the ?q=

7. Add a new drillable region:

```
@DESCENDANTS(Product),@CHILDREN(Market)
```

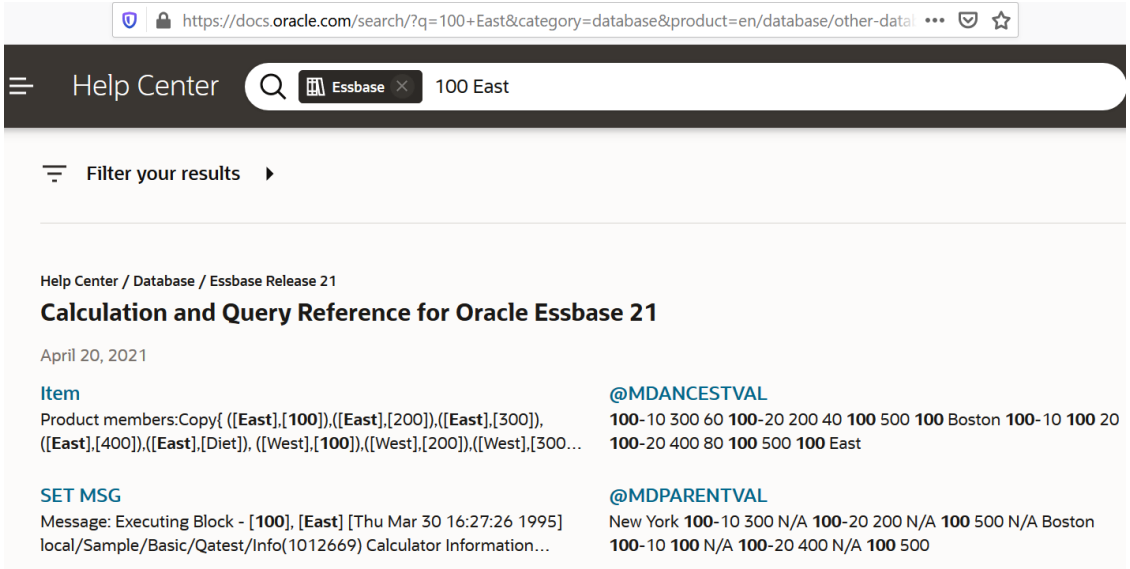
With this drill through report, when you drill through from a cell intersection for any Product generation member in any child of Market dimension, a browser is launched, and Essbase documentation is searched for the parameter represented by VALUE.

Log into the cube from Smart View.

In this example, we've selected a blue background for the drillable regions. Drill through on Actual, Colas, East, Qtr1.

	A	B	C	D	E
1					Measures
2	Actual	Colas	East	Qtr1	2747
3	Actual	Colas	East	Qtr2	3352
4	Actual	Colas	East	Qtr3	3740
5	Actual	Colas	East	Qtr4	2817

The browser is launched and the Essbase documentation is searched using the parameter values **100** and **East** (recall that Colas is the alias for Product category 100).



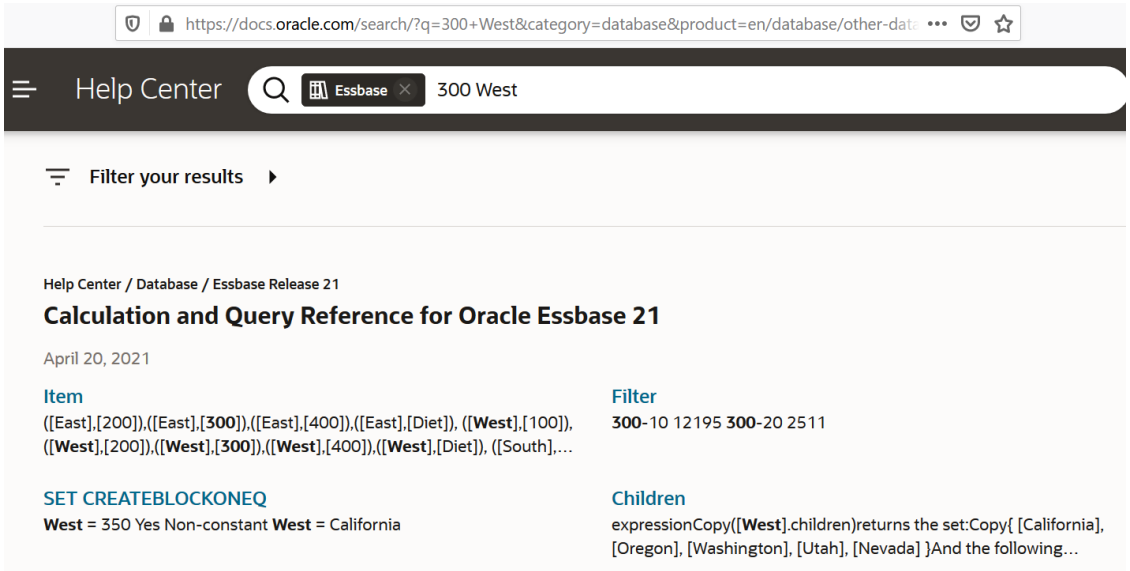
Copyright © 2019, 2026, Oracle and/or its affiliates.

Drill through on a different cell to see how the parameter passed to the drill through URL changes depending on cell intersection context in Smart View.

Drill through on Actual, Cream Soda, West, Qtr4:

	A	B	C	D	E
1					Measures
2	Actual	Cream Soda	West	Qtr1	2363
3	Actual	Cream Soda	West	Qtr2	2739
4	Actual	Cream Soda	West	Qtr3	2937
5	Actual	Cream Soda	West	Qtr4	2692
6	Actual	Cream Soda	West	Year	10731

The Essbase documentation is now searched for **300** and **West**:



Drill Through from Multiple Cells

You can drill through to a source external to Essbase from multiple Smart View cells, and the resulting drill through report will reflect the context of all the cells from which you drilled through.

You can drill through from non-contiguous cells, from a contiguous range of cells, from separate ranges, or from a range of cells covering different generations in the hierarchy.

The following examples show query sheets for several different drill through scenarios and the resulting output.

> **Note**
>
> Only the drill through reports that are common to all of the ranges will be available.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Example 1: drill through from more than one non-contiguous cells

Drilling through from Colas and Cream Soda returns a drill through report filtered by the grid context for product, with products 100 and 300 (100 and 300 are the product SKUs associated with Colas and Cream Soda).

	A	B	C	D	E	F	G
1			Sales				
2			East				
3			Actual				
4	Colas	Qtr1	6292				
5	Root Beer	Qtr1	5726				
6	Cream Soda	Qtr1	4868				
7	Fruit Soda	Qtr1	3735				
8	Diet Drinks	Qtr1	1884				
9	Product	Qtr1	20621				

Drill through on Colas + Cream Soda

	A	B	C	D	E
1	DIMENSION_PRODUCT	DIMENSION_MARKET	YEAR_PARENT	DIMENSION_SCENARIO	SALES
2	300	Connecticut	Qtr1	Actual	1070
3	300	New Hampshire	Qtr1	Actual	225
4	300	New York	Qtr1	Actual	2033
5	300	Massachusetts	Qtr1	Actual	391
6	300	Florida	Qtr1	Actual	1149
7	100	Connecticut	Qtr1	Actual	944
8	100	New Hampshire	Qtr1	Actual	654
9	100	New York	Qtr1	Actual	1998
10	100	Massachusetts	Qtr1	Actual	1456
11	100	Florida	Qtr1	Actual	1240

Example 2: drill through from a contiguous range of cells

Drilling through from Colas, Root Beer, Cream Soda, and Fruit Soda returns a drill through report filtered by the grid context for product, with products 100, 200, 300, and 400 (these are the product SKUs associated with the alias names Colas, Root Beer, Cream Soda, and Fruit Soda).

	A	B	C	D	E	F	G	H
1			Sales					
2			East					
3			Actual					
4	Colas	Qtr1	6292					
5	Root Beer	Qtr1	5726					
6	Cream Soda	Qtr1	4868					
7	Fruit Soda	Qtr1	3735					
8	Diet Drinks	Qtr1	1884					
9	Product	Qtr1	20621					

Drill through on Colas + Root Beer + Cream Soda + Fruit Soda

Copyright © 2019, 2026, Oracle and/or its affiliates.

	A	B	C	D	E
1	DIMENSION_PRODUCT	DIMENSION_MARKET	YEAR_PARENT	DIMENSION_SCENARIO	SALES
2	200	New York	Qtr1	Actual	1778
3	200	Massachusetts	Qtr1	Actual	1385
4	300	Massachusetts	Qtr1	Actual	391
5	300	New Hampshire	Qtr1	Actual	225
6	400	Florida	Qtr1	Actual	558
7	400	New Hampshire	Qtr1	Actual	264
8	100	New Hampshire	Qtr1	Actual	654
9	400	Massachusetts	Qtr1	Actual	428
10	200	Florida	Qtr1	Actual	1185
11	200	Connecticut	Qtr1	Actual	869
12	100	New York	Qtr1	Actual	1998
13	300	New York	Qtr1	Actual	2033
14	100	Florida	Qtr1	Actual	1240
15	100	Connecticut	Qtr1	Actual	944
16	300	Florida	Qtr1	Actual	1149
17	400	New York	Qtr1	Actual	1896
18	400	Connecticut	Qtr1	Actual	589
19	200	New Hampshire	Qtr1	Actual	509
20	100	Massachusetts	Qtr1	Actual	1456
21	300	Connecticut	Qtr1	Actual	1070

Example 3: drill through from separate ranges of cells

Drilling through from the children of Colas and the children of Cream Soda returns a drill through report filtered by the grid context for product, with products 100-10, 100-20, 100-30, 300-10, 300-20 and 300-30 (these are the product SKUs associated with the alias names for the children of Colas and the children of Cream Soda).

	A	B	C	D	E	F	G
1			Sales				
2			East				
3			Actual				
4	Cola	Qtr1	5371				
5	Diet Cola	Qtr1	620				
6	Caffeine Free Cola	Qtr1	301				
7	Colas	Qtr1	6292				
8	Root Beer	Qtr1	5726				
9	Dark Cream	Qtr1	3037				
10	Vanilla Cream	Qtr1	1499				
11	Diet Cream	Qtr1	332				
12	Cream Soda	Qtr1	4868				
13	Fruit Soda	Qtr1	3735				
14	Diet Drinks	Qtr1	1884				
15	Product	Qtr1	20621				

Drill through on children of Colas + Children of Cream Soda

Copyright © 2019, 2026, Oracle and/or its affiliates.

	A	B	C	D	E
1	DIMENSION_PRODUCT	DIMENSION_MARKET	YEAR_PARENT	DIMENSION_SCENARIO	SALES
2	300-30	Florida	Qtr1	Actual	332
3	300-20	Connecticut	Qtr1	Actual	498
4	300-20	New York	Qtr1	Actual	542
5	300-20	Florida	Qtr1	Actual	459
6	300-10	Connecticut	Qtr1	Actual	572
7	300-10	New Hampshire	Qtr1	Actual	225
8	300-10	New York	Qtr1	Actual	1491
9	300-10	Massachusetts	Qtr1	Actual	391
10	300-10	Florida	Qtr1	Actual	358
11	100-30	New Hampshire	Qtr1	Actual	301
12	100-20	Florida	Qtr1	Actual	620
13	100-10	Connecticut	Qtr1	Actual	944
14	100-10	New Hampshire	Qtr1	Actual	353
15	100-10	New York	Qtr1	Actual	1998
16	100-10	Massachusetts	Qtr1	Actual	1456
17	100-10	Florida	Qtr1	Actual	620

Example 4: drill through from a range of cells covering different generations in a hierarchy

Drilling through from Root Beer, and Cream Soda, and the children of Colas returns a drill through report filtered by the grid context for product, with products 100-10, 100-20, 100-30, 200 and 300 (these are the product SKUs associated with the alias names for the children of Colas, Root Beer, and Cream Soda).

	A	B	C	D	E	F	G
1			Sales				
2			East				
3			Actual				
4	Cola	Qtr1	5371				
5	Diet Cola	Qtr1	620				
6	Caffeine Free Cola	Qtr1	301				
7	Colas	Qtr1	6292				
8	Root Beer	Qtr1	5726				
9	Cream Soda	Qtr1	4868				
10	Fruit Soda	Qtr1	3735				
11	Diet Drinks	Qtr1	1884				
12	Product	Qtr1	20621				

Children of Colas + Root Beer + Cream Soda

Copyright © 2019, 2026, Oracle and/or its affiliates.

	A	B	C	D	E
1	DIMENSION_PRODUCT	DIMENSION_MARKET	YEAR_PARENT	DIMENSION_SCENARIO	SALES
2	300	Connecticut	Qtr1	Actual	1070
3	300	New Hampshire	Qtr1	Actual	225
4	300	New York	Qtr1	Actual	2033
5	300	Massachusetts	Qtr1	Actual	391
6	300	Florida	Qtr1	Actual	1149
7	200	Connecticut	Qtr1	Actual	869
8	200	New Hampshire	Qtr1	Actual	509
9	200	New York	Qtr1	Actual	1778
10	200	Massachusetts	Qtr1	Actual	1385
11	200	Florida	Qtr1	Actual	1185
12	100-30	New Hampshire	Qtr1	Actual	301
13	100-20	Florida	Qtr1	Actual	620
14	100-10	Connecticut	Qtr1	Actual	944
15	100-10	New Hampshire	Qtr1	Actual	353
16	100-10	New York	Qtr1	Actual	1998
17	100-10	Massachusetts	Qtr1	Actual	1456
18	100-10	Florida	Qtr1	Actual	620

Debug Drill Through using Essbase Server Platform Log

When Smart View users run drill through reports, the SQL query that Essbase executes is written to the Essbase Server platform log. You can use this log to examine the queries in case you are not seeing the expected drill through results.

The Essbase Server `platform.log` is located in `<Domain Root>/<Domain Name>/servers/essbase_server1/logs/essbase`.

It is recommended to use this log to test your drill through report definitions, during the design phase. To find the relevant and most recent log entries written right after you have executed a drill through, use the following command (for Linux bash shell):

```
tail -f platform.log
```

For each drill through operation, the user name and timestamp are logged, and Essbase records the generated query as shown in the following example:

```
Query executed on the database: SELECT "PRODUCT", "MONTH", "CITY"[[
FROM (select * from SAMPLEBASIC) DatasourceName
WHERE ("MONTH" = 'Feb' OR "MONTH" = 'Jan' OR "MONTH" = 'Mar')
AND ("PRODUCT" = '100-10-30' OR "PRODUCT" = '100-10-40' OR "PRODUCT" =
'100-30' OR "PRODUCT" = '100-20' OR "PRODUCT" = '100-10-10' OR "PRODUCT" =
'100-10-20') AND "CITY" = 'New York']]
```


Copyright © 2019, 2026, Oracle and/or its affiliates.

11
Calculate Cubes

An Essbase cube contains two types of values: values that you enter, called input data, and values that are calculated from input data.

A cube can be calculated using outline formulas, calculation scripts, or both.

Outline calculation, which is the simplest calculation method, bases the calculation of a cube on the relationships between members in the cube outline and on any formulas that are associated with members in the outline.

Essbase block storage (BSO) calculation functions may be applied in BSO outline formulas, with their results impacting queries from Smart View, MDX, and other grid clients. The same functions, along with the calculation commands, can be used to write procedural calculation scripts.

Calculation script calculation lets you procedurally calculate a cube; for example, you can calculate one part of a cube before another, or copy data values between members.

The topics in this section are about BSO calculation script calculation:

- Access to Calculations
- Create Calculation Scripts
- Execute Calculations
- Use Substitution Variables
- Set Two-Pass Calculation Properties
- Tracing Calculations
- Calculate Selected Tuples

Access to Calculations

If you have the Database Update application permission, you have access to run the default calculation on the cube (from Smart View), and to run specific calculation scripts provisioned to you. If you have the Application Manager or Database Manager application permission, you have Calc privileges and rights to execute all calculations, and to provision access to execute specific calculation scripts.

To provision users access to execute calculation scripts in the Essbase web interface, first provision the user to that application with Database Update permission, then add the user on the **Roles** tab in the calc script editor.

- Redwood
- Classic

Copyright © 2019, 2026, Oracle and/or its affiliates.

Redwood

1. On the Home page, open the application.
2. Click **Permissions**.
3. Click **Add** and a list of users and groups is displayed.

> **Note**
>
> Clicking **Add** in this dialog does not allow you to add new users. Instead, you can add users that have already been provisioned using an identity provider. This topic assumes that you have users provisioned. There are several ways to provision Essbase users. See Manage Essbase User Roles and Application Permissions for independent deployments, or Manage Users and Roles for stack deployments.

4. Click **Add user** + next to a user.
5. Click **Close** to close the user list.
6. Select **Database Update** for the added user.
7. Grant calculation script access,
   a. Select the **General** page of the open application, and open the database (cube).
   b. Click **Scripts**, and click **Calculation Scripts**.
   c. Click on the script name.
   d. Click **Roles**, and click **Add members**.
   e. Click **Add user** + next to the user name.
   f. Click **Close**.
   g. The user is displayed as a member of the script.

Classic

1. On the Applications page, click the **Actions** menu to the right of the application name.
2. Select **Inspect**, and then select **Permissions**.
3. Click **Add** on the right-hand side of the dialog box.
   A list of users and groups is displayed.
4. Click **Add** next to a user.
5. Select **Database Update**.
6. Grant calculation script access.
   a. On the Applications page, expand the application and click the **Actions** menu to the right of the cube name.
   b. Select **Inspect**, and then select **Scripts**.
   c. Select **Calculation Scripts**.
   d. Click on the script name.
   e. Select **Roles**.

Copyright © 2019, 2026, Oracle and/or its affiliates.

f. Click **Add** .

g. Click **Add** next to the user name.

h. Click **Close**.
The user is displayed as a member of the script.

Create Calculation Scripts

Essbase calculation scripts specify how block storage cubes are calculated and, therefore, override outline-defined cube calculations. For example, you can calculate cube subsets or copy data values between members.

You create calculation scripts using a script editor in the Essbase web interface.

Calculation scripts do not apply to aggregate storage applications.

- Redwood
- Classic

Redwood

1. On the Home page, open the application and then open the database (cube).
2. Click **Scripts**, and then click **Calculation Scripts**.
3. Click **Create** to create a new calculation script.
4. Enter a name for the new script.
5. If member names are required in your calculation script, drill into the **Member Tree** to find the members you want to add.
6. Double-click dimension or member names to insert them into the script.
7. If function names are required in your calculation script, use the **Function Name** menu to find calculation functions and add them to the script.
See the **Function description** under the menu to read descriptions of each function.
8. Click **Validate** before saving your script.
Validating a script verifies the script syntax. For example, incorrectly spelled function names and omitted end-of-line semicolons are identified. Validation also verifies dimension names and member names.
9. Correct any validation errors.
10. Click **Save**.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Classic

1. On the Application page, expand the application.
2. From the Actions menu, to the right of the cube name, launch the inspector.
3. Select the **Scripts** tab, and then select the **Calculation Scripts** tab.
4. Click Add to create a new calculation script.
5. Enter a name in the **Script Name** field.
6. If member names are required in your calculation script, drill into the **Member Tree** to find the members you want to add.
   Right-click dimension or member names to insert them into the script.
7. If function names are required in your calculation script, use the **Function Name** menu to find calculation functions and add them to the script.
   See the **Function description** under the menu to read descriptions of each function.
8. Click **Validate** before saving your script.
   Validating a script verifies the script syntax. For example, incorrectly spelled function names and omitted end-of-line semicolons are identified. Validation also verifies dimension names and member names.
9. Correct any validation errors.
10. Click **Save**.

To learn about calculation script logic, see Developing Calculation Scripts for Block Storage Databases.

To learn about calculation functions and commands, see Calculation Functions and Calculation Commands.

Execute Calculations

After creating and saving Essbase calculation scripts, you can execute them in the script editor of the Essbase web interface to perform the calculations on data loaded in your cube.

1. Create your calculation script, or upload an existing calculation script.
2. Navigate to a script,
   - In the Redwood Interface
     a. On the Home page, click **Files**.
     b. Double click the application name and then double click the cube name.
     c. Click the script name, and the script is opened in the calc script editor.
   - In the Classic Web Interface
     a. On the Applications page, expand an application, and select a cube.
     b. From the **Actions** menu, to the right of the cube name, launch the inspector.
     c. Select **Scripts**, and select the script you want to execute.
3. In the script editor, click **Execute** and then select either **Run in Foreground** or **Run in Background**.

Copyright © 2019, 2026, Oracle and/or its affiliates.

- If you choose **Run in Forground**, **Script execution in progress** is displayed and you can't close the script editor until the calculation is completed.
- If you choose **Run in Background**, you can close the script editor and later check the Jobs page (from the Applications page, select Jobs) for the status of the calculation.

You can also execute calculation scripts from the Jobs page or from Smart View (whether or not they contain point-of-view based substitution variables).

Calculation scripts can contain runtime substitution variables designed to derive the calculation scope from the point of view (POV) in a Smart View grid. These types of calculation scripts can only be executed from Smart View, because the point of view can only be known from a Smart View grid.

Assign access to execute specific calculation scripts:

1. Ensure that you are logged into the Essbase web interface as a service administrator or power user.
2. Navigate to the **Roles** tab for your calculation script.
   In the Redwood Interface
   a. On the Home page, open the application and then open the database (cube).
   b. Click **Scripts** and click **Calculation Scripts**.
   c. Select the script for which you want to assign access, and click the **Roles** tab.

   In the Classic Web Interface
   a. On the Applications page, expand an application, and select a cube.
   b. From the **Actions** menu, to the right of the cube name, launch the inspector.
   c. Select the **Scripts** tab, and then select the **Calculation Scripts** tab.
   d. Select a script and select the **Roles** tab.
3. Add the users or groups to assign them access and save your changes. The users or groups are given permission to execute the specific calculation script.

See also: Create Calculation Scripts

Work with Files and Artifacts

Use Substitution Variables

Use **substitution variables** in Essbase calculation scripts to store values that might change. Use **runtime substitution variables** when you need different users to specify different values for the same script.

For example, if a variety of your calculation scripts, formulas, filters, report scripts, and MDX scripts all need to refer to the current month, you would not want to search and replace the month approximately every 30 days throughout your library of cube artifacts. Instead, you can define a substitution variable named CurrMonth, and change its assigned value each month to the appropriate month. All of the cube artifacts that reference the variable will then reference the appropriate month.

Here is an example of a simple substitution variable to represent the current month:

Variable name: CurrMonth

Value: Jan

Copyright © 2019, 2026, Oracle and/or its affiliates.

Substitution variable values apply to all users who run a calculation script containing the variable. For example, if CurrMonth has the value Jan, then all scripts containing &CurrMonth will execute for Jan. The scope of a substitution variable can be:

- global (for all applications and cubes on the server)
- application (for all cubes in the application)
- cube (for a single cube)

To define or update a substitution variable for a specific cube,

---

- Redwood
- Classic

Redwood

1. On the Home page, open the application and then open the database (cube).
2. Click **Variables**.
3. To create a new variable, click **Create**, enter the variable name and value, and click **Save** .
4. If you are editing the value of an existing variable, double click the value (or click **Edit**), type the updated value, and press Enter (or click **Save**).

Classic

1. In the Essbase web interface, on the Applications page, expand the application to show the cube you want to modify.
2. From the **Actions** menu to the right of the cube, launch the inspector.
3. Select the **Variables** tab.
4. To create a new variable, click **Add**, enter the variable name and value, and click **Save**.
5. If you are editing the value of an existing variable, double click the **Value** field, type the updated value, and press Enter.
6. Click **Close**.

To define or update a substitution variable for a specific application,

---

- Redwood
- Classic

Redwood

1. On the Home page, open the application.

Copyright © 2019, 2026, Oracle and/or its affiliates.

2. Click **Variables**.
3. To create a new variable, click **Create**, enter the variable name and value, and click **Save** .
4. If you are editing the value of an existing variable, double click the value (or click **Edit**), type the updated value, and press Enter (or click **Save**).

Classic

1. On the Applications page, from the Actions menu to the right of the application, launch the inspector.
2. Select the **Variables** tab.
3. To create a new variable, click **Add**, enter the variable name and value, and click **Save**.
4. If you are editing the value of an existing variable, double click the **Value** field, type the updated value, and press Enter.
5. Click **Close**.

To define or update a substitution variable globally,

- Redwood
- Classic

Redwood

1. On the Home page, click **Console**.
2. Click the **Variables** tile.
3. To create a new variable, click **Create**, enter the variable name and value, and click **Save** .
4. If you are editing the value of an existing variable, double click the value (or click **Edit**), type the updated value, and press Enter (or click **Save**).

Classic

1. In the Essbase web interface, click **Console**.
2. Click the **Variables** tab.
3. To create a new variable, click **Add**, enter the variable name and value, and click **Save**.
4. If you are editing the value of an existing variable, double click the **Value** field, type the updated value, and press Enter.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Once your substitution variable is defined, you can use it in calculation scripts, formulas, filters, MDX scripts, load rules, and reports. To reference the variable, prefix it with the & symbol.

Here is an example of a calculation script that references a substitution variable:

```
FIX(&CurrMonth)
   CALC DIM (Measures, Product);
ENDFIX
```

Here is an example of a formula that references a substitution variable:

```
@ISMBR(&CurrMonth)
```

Runtime substitution variables enable you to declare variables and their values in the context of a runtime action, such as a calculation script, MaxL script, or MDX query. Runtime substitution variables can be assigned to have numeric values or refer to member names. A default value can be assigned in case a user does not change an input value. Also, for calculation scripts, the variable value can be populated at runtime from the members of a dimension presented on a Smart View grid. For calculation scripts with variable values that populate at runtime, you must launch the calculation script from Smart View, as the variable has no definition outside the context of the grid.

Runtime substitution variables may be defined in the calculation script using key-value pairs:

```
SET RUNTIMESUBVARS
{
   myMarket = "New York";
   salesNum = 100;
   pointD = "Actual"->"Final";
}
```

Or, to define runtime substitution variables with values that change dynamically depending on the POV, assign the definition to POV, and use XML syntax to enable Smart View contextual prompts.

For more information, see

- Implement Variables for Changing Information
- Runtime Substitution Variables in Calculation Scripts Run in Essbase and Runtime Substitution Variables in Calculation Scripts Run in Smart View
- The SET RUNTIMESUBVARS calculation command
- The gallery template Sample_Basic_RTSV, which you can find in **Files** > `Gallery` > `Technical` > `Calc`.

Set Two-Pass Calculation Properties

From the outline editor in the Essbase web interface, you can apply the Two-Pass Calculation property to members in non hybrid mode, block storage cubes, if you have members that need to be calculated twice to produce the required value.

To obtain the correct values for two-pass members, the outline is calculated, and then members that are dependent on the calculated values of other members are recalculated.

Copyright © 2019, 2026, Oracle and/or its affiliates.

> **Note**
>
> Do not use two-pass calculation with hybrid mode cubes. Only use solve order.

Even though two-pass calculation is a property that you can give to any non-attribute dimension member, it works only on members of the Accounts dimension and Dynamic Calc members. If two-pass calculation is assigned to any other member, it is ignored.

Two-pass calculations are supported only on block storage cubes. Aggregate storage cubes use member solve order, instead of two-pass calculation, to control when members are calculated.

1. On the Home page, navigate to the application, and select the Actions menu ⋯.
2. Choose **Databases** > *Database Name* > **Launch Outline**.
3. If the outline is locked, and you are an administrator, click **Unlock**.
   Before you forcefully unlock a locked outline, make sure that no one else is working with it.
4. Click **Edit**.
5. In the outline editor, find and select the member you want to modify.
6. Right-click the member and select **Inspect**.
7. On the **General** tab, in the **Two-pass calculation** field, select **True**.

See Setting Two-Pass Calculations.

Trace Calculations

Use Essbase calculation tracing to gain insight into member formula processing, helping you debug and refine your block storage calculation scripts. Enable CALCTRACE for context sensitive Smart View calc tracing, or use SET TRACE command to select data intersections to trace.

Calculation tracing enables you to access logged information about a calculation, after the calculation script successfully executes against a cube.

Tracing a calculation does not change anything about calculation behavior. If a calculation is launched in Smart View, and the connected server has calculation tracing enabled by an administrator, Smart View displays a pop-up dialog box containing details, after the calculation runs. The calculation tracing information can be pasted from the pop-up dialog into a text editor. Or, you can find the same information in `calc_trace.txt`, located in the database files directory in Essbase.

The calculation tracing information can help you debug calculation script execution, in case the results of the calculation are not what you expected.

Calculation tracing is not supported on applications with scenario management enabled.

To enable calculation tracing, the administrator must first turn on the CALCTRACE application configuration parameter. After calculation tracing is enabled for your application, there are two ways to take advantage of it:

- In Smart View, you can use context-sensitive tracing for a single cell value.
  1. In Smart View, connect a query sheet to the application for which you enabled calculation tracing.

Copyright © 2019, 2026, Oracle and/or its affiliates.

2. Highlight a data cell whose calculated value you would like to trace.
3. In the Data panel of the Essbase tab, click the **Calculate** button and select a calculation script to execute. You will see the point-of-view from your highlighted data cell in the trace member runtime prompts.
4. Click **Launch** to execute the calculation script.
   The full scope of the calculation as contained in the script will be calculated, but only the highlighted data cell context will be traced during the calculation.
5. At the end of the calculation script, examine the **Calculation Result** dialog box, which shows the pre- and post-calculation results for your highlighted data cell.
   If the highlighted data cell was not modified during the calculation, you will see a message indicating that the cell was not modified.

- In calculation scripts, you can use the SET TRACE calculation command to select data intersections to trace. SET TRACE enables you to trace multiple data cells. Additionally, you can trace sections of calculation scripts by using a combination of `SET TRACE` *`mbrList`* (to turn calculation tracing on over a member list) and `SET TRACE OFF` (to disable calculation tracing until a new SET TRACE is encountered in the script. To use SET TRACE command, you must execute the calculation script outside of Smart View, using Cube Designer, the CLI calc command, a Run Calculation job in the Essbase web interface, or MaxL (execute calculation statement).

The following calculation script is run on Sample Basic. The script includes a SET TRACE command, requesting detailed information to be logged for the data intersection (cell) that represents budgeted January sales, in the California market, for the product SKU number 100-10.

```
SET TRACEID "id042"
SET TRACE ("100-10", "California", "Jan", "Sales", "Budget");
FIX("California", "Budget")
   "Sales" (
      "100-10" = @MEMBER(@CONCATENATE(@NAME(@PARENT("Product")), "-20")) / 10;
   );
ENDFIX;
```

> **Note**
>
> The SET TRACEID command is also recommended, to prevent your calc tracing file from being overwritten.

Sample Basic has two sparse dimensions: Product and Market. The member formula is on Sales, a member of Measures, which is a dense dimension. The FIX statement's member list only contains one sparse member, California, which belongs to the Market dimension.

The number of existing blocks in the FIX scope determines the number of times the traced cell is calculated. In this example, the calculation cycles through all existing sparse member combinations of California. Each of these combinations represents a block.

After the calculation completes, the following tracing information is logged and displayed in `calc_trace_id042.txt`:

```
Tracing cell: [100-10][California][Jan][Sales][Budget]  (Cell update count: 1)
Previous value: 840.00
Dependent values:
    [100-20][California][Jan][Sales][Budget] = 140.00
```


Copyright © 2019, 2026, Oracle and/or its affiliates.

```
New value: [100-10][California][Jan][Sales][Budget] = 14.00

Computed in lines: [91 - 93] using:
"Sales"(
"100-10"=@MEMBER(@CONCATENATE(@NAME(@PARENT("Product")),"-20"))/10;
)

Tracing cell: [100-10][California][Jan][Sales][Budget]  (Cell update count: 2)
Block from FIX scope: [100-30][California]
Actual block used in calculation: [100-10][California]
Previous value: 14.00
Dependent values:
    [100-20][California][Jan][Sales][Budget] = 140.00
New value: [100-10][California][Jan][Sales][Budget] = 14.00
Computed in lines: [91 - 93] using:
"Sales"(
"100-10"=@MEMBER(@CONCATENATE(@NAME(@PARENT("Product")),"-20"))/10;
)

Tracing cell: [100-10][California][Jan][Sales][Budget]  (Cell update count: 3)
Block from FIX scope: [200-10][California]
Actual block used in calculation: [100-10][California]
Previous value: 14.00
Dependent values:
    [200-20][California][Jan][Sales][Budget] = 520.00
New value: [100-10][California][Jan][Sales][Budget] = 52.00
Computed in lines: [91 - 93] using:
"Sales"(
"100-10"=@MEMBER(@CONCATENATE(@NAME(@PARENT("Product")),"-20"))/10;
)

[...calc iterations 4-7 are omitted from example...]

Tracing cell: [100-10][California][Jan][Sales][Budget]  (Cell update count: 8)
Block from FIX scope: [400-30][California]
Actual block used in calculation: [100-10][California]
Previous value: 9.00
Dependent values:
    [400-20][California][Jan][Sales][Budget] = 90.00
New value: [100-10][California][Jan][Sales][Budget] = 9.00
Computed in lines: [91 - 93] using:
"Sales"(
"100-10"=@MEMBER(@CONCATENATE(@NAME(@PARENT("Product")),"-20"))/10;
)
```

The calculation tracing log provides the following insights about how the calculation worked, on the cell that was traced:

- The traced cell was calculated several times, and the cell value was overwritten each time with the new value (the reported cell update count stops at 8).
- The value of the cell, before calculation, was 840.00.
- For each calculation occurrence, dependent values and new values are shown. Dependent values come from the member formula in the FIX statement.

Copyright © 2019, 2026, Oracle and/or its affiliates.

- The final value of the traced cell, after all calculation completes, is 9, but it represents the value of product `"400-20"->California` divided by 10.
- Lines 91-93 of the calculation script, containing a member formula on Sales, are responsible for the updated values.

For each of the blocks cycled through, Sales is calculated using the formula:

```
"100-10"=@MEMBER(@CONCATENATE(@NAME(@PARENT("Product")),"-20"))/10
```

The formula contains a sparse member on the left hand side, which could cause the actual calculation block to be different than the initial FIX block. For example, when the calculation cycles through "California"->"100-20", the calculations are actually done in "California"->"100-10".

The trace log entries entitled `Block from FIX scope` and `Actual block used in calculation` are only printed if there is a discrepancy between the blocks in the FIX statement and the block that is represented in the member formula. These log entries can provide indications as to why there are duplicate calculations, helping you to debug your calculation scripts.

Calculate Selected Tuples

By selecting tuples, you can focus your Essbase calculations in the active Smart View grid, limiting their scope to specific slices of data in your block storage cube.

The following sections describe tuple calculation:

- Use Case for Tuple Calculation
- Understand Tuple-Based Calculation
- Select Tuples for Point of View Calculation
- Examples of Tuple Selection to Reduce Calculation Scope

For the syntax for employing @GRIDTUPLES in a calculation script, see FIX…ENDFIX.

Use Case for Tuple Calculation

By selecting tuples, you can focus your Essbase calculations in the active Smart View grid, limiting their scope to specific slices of data in your block storage cube.

Tuple selection helps you optimize asymmetric grid calculations across dimensions, avoiding over-calculation.

Essbase calculation tuples differ from tuples used in MDX queries. Calculation performance and cube size are mainly driven by the number of blocks in the cube (given a specific block size). For this reason, calculation tuples are specified only for sparse member combinations. In addition, for ease of calculation scripting, multiple members from a single sparse dimension can be included in a calculation tuple specification. For example, if you specify ("New York", "California", "Actual", "Cola") as a calculation tuple, then you calculate the following cell intersections:

```
"New York"->"Actual"->"Cola"
"California"->"Actual"->"Cola"
```

Consider the following symmetric grid. It is symmetrical because each product has the same markets and scenario (Actual) represented in the grid.

Copyright © 2019, 2026, Oracle and/or its affiliates.

		Profit	Inventory	Ratios
		Actual	Actual	Actual
		Jan	Jan	Jan
Cola	New York			
	Massachus			
	Florida			
	Connecticu			
	New Hamp			
Diet Cola	New York			
	Massachus			
	Florida			
	Connecticu			
	New Hamp			

The following grid is asymmetric, because the Diet Cola product has fewer markets in the grid than the Cola product has.

		Profit	Inventory	Ratios
		Actual	Actual	Actual
		Jan	Jan	Jan
Cola	New York			
	Massachus			
	Florida			
	Connecticu			
	New Hamp			
Diet Cola	New York			
	Florida			

The default calculation scope, when more than one dimension is in a FIX statement or a Smart View grid point of view (POV), is to calculate the cross product (all possible combinations) of the members in the FIX or grid. In other words, a POV-driven calculation in which product and market combinations are taken from the grid calculates all of these row-member combinations:

```
Cola->"New York"
Cola->"Massachusetts"
Cola->"Florida"
Cola->"Connecticut"
Cola->"New Hampshire"
"Diet Cola"->"New York"
"Diet Cola"->"Massachusetts"
"Diet Cola"->"Florida"
"Diet Cola"->"Connecticut"
"Diet Cola"->"New Hampshire"
```

This may be more calculation activity than you need. If you want to calculate *only* the combinations shown on the grid, you can specify which tuples to calculate, and limit the calculation to a smaller slice. Calculating tuples can also lower calculation time and cube size.

```
Cola->"New York"
Cola->"Massachusetts"
Cola->"Florida"
Cola->"Connecticut"
Cola->"New Hampshire"
```


Copyright © 2019, 2026, Oracle and/or its affiliates.

```
"Diet Cola"->"New York"
"Diet Cola"->"Florida"
```

Understand Tuple-Based Calculation

A calculation **tuple** is a way to represent a data slice of members, from two or more sparse dimensions, to be used in an Essbase block storage calculation.

Examples of valid calculation tuples:

- `("Diet Cola", "New York")`
- `("Diet Cola", "Cola", Florida)`
- `(Cola, "New Hampshire")`

If you write MDX expressions, you might be aware of these tuple restrictions that apply to MDX:

- Only a single member from each dimension can be included in an MDX tuple
- All tuples in an MDX set must have the same dimensions represented, in the same order

However, when you select tuples in calculation scripts, these requirements are relaxed for convenience. You may freely write tuple expressions, and the tuples may describe member lists, as the following tuple does: `(@Children(East), Cola)`.

Select Tuples for Point of View Calculation

An easy way to select tuples is to insert them explicitly into a calculation script, as a list inside the FIX statement.

Recall that the format of a FIX statement is as follows:

```
FIX (fixMbrs)
COMMANDS ;
ENDFIX
```

In the FIX statement below, two tuples are specified before the command block begins. The tuples are enclosed within the curly braces { } that delimit a **set**, which is a collection of tuples.

```
FIX({
  (@Children(East), Cola),
  ("New York", Florida, "Diet Cola")
  })
Sales (Sales = Sales + 10;);
ENDFIX
```

Another way to select tuples is contextually, based on whichever members are present in a Smart View grid POV at calculation run time. You do this by providing the @GRIDTUPLES function as an argument to FIX, in your calculation script.

```
FIX ({@GRIDTUPLES(Product, Market)})
   Sales (Sales = Sales + 10;);
ENDFIX
```


Copyright © 2019, 2026, Oracle and/or its affiliates.

If you execute this calculation script from Smart View against the grid below, then only the displayed combinations of products and markets are calculated. For example, "Diet Cola"->Massachusetts is not calculated, as it is not shown explicitly on the grid. Note that all scenarios (the third sparse dimension in this sample cube) are calculated, even though only Actual is shown on the grid. This is because the Scenario dimension is not part of the GRIDTUPLES statement in the calculation script.

		Profit	Inventory	Ratios
		Actual	Actual	Actual
		Jan	Jan	Jan
Cola	New York			
	Massachus			
	Florida			
	Connecticu			
	New Hamp			
Diet Cola	New York			
	Florida			

Tuple selection, whether done using explicit lists of tuples or by using the @GRIDTUPLES function, is applicable only in the context of the FIX…ENDFIX calculation command. The syntax of the FIX statement is expanded to enable tuple selection:

```
FIX ([{ tupleList | @GRIDTUPLES(dimensionList) },] fixMbrs)
COMMANDS ;
ENDFIX
```

- *tupleList* - comma-separated set of tuples.
- *dimensionList* - at least two sparse dimensions whose members from the active Smart View grid are used to define the calculation regions. (In calculation scripts, you can use only sparse dimensions to define tuples.)
- *fixMbrs* - a member or list of members.

Examples of Tuple Selection to Reduce Calculation Scope

Using a Smart View grid and an Essbase calculation script FIX statement, you can calculate selected member tuples based on the grid point of view (POV). Alternatively, you can explicitly type the tuple combinations in your FIX statement, removing the dependency on a particular Smart View grid to define the calculation scope.

Calculating selected tuples helps you efficiently work with asymmetric regions in both calculation scripts and Smart View grids.

Consider the following examples:

- No Tuple Selection - Calculates in the default manner, based on current Smart View grid point-of-view (POV). The calculation is not limited to any specific tuples.
- Selection of Named Sparse Dimensions - Calculates tuples from two or more sparse dimensions named in a calculation script. The calculation is limited to members from the tuple dimensions that are present in the Smart View grid.
- Selection of Contextual Sparse Dimensions - Calculates tuples from sparse dimensions selected at run-time. The calculation is limited to members from the tuple dimensions present in the Smart View grid.

Copyright © 2019, 2026, Oracle and/or its affiliates.

To try the examples, download the `CalcTuple_Tuple.xlsx` workbook template from the `Technical > Calc` section of the **gallery** folder in the **Files** area of the Essbase web interface. Refer to the README worksheet in the workbook for instructions.

No Tuple Selection

Demonstrating the default Essbase block storage calculation behavior that occurs when you do not select tuples, the following calculation script calculates the entire cross-product of Product and Market dimension members from a Smart View grid.

With the help of two runtime substitution variables (RTSV) defined in the SET RUNTIMESUBVARS block, calculation is limited to whichever Product and Market points of view are present in the grid when the calculation is run from Smart View.

```
SET RUNTIMESUBVARS
{
ProductGridMembers = POV
<RTSV_HINT><svLaunch>
<description>All Product's members on the grid</description>
<type>member</type>
<dimension>Product</dimension><choice>multiple</choice>
</svLaunch></RTSV_HINT>;
MarketGridMembers = POV
<RTSV_HINT><svLaunch>
<description>All Market's members on the grid</description>
<type>member</type> <dimension>Market</dimension><choice>multiple</choice>
</svLaunch></RTSV_HINT>;
};
FIX (
&ProductGridMembers, &MarketGridMembers
)
Marketing(
   Marketing = Marketing +1;
);
ENDFIX
```

Selection of Named Sparse Dimensions

Using the @GRIDTUPLES function to select the tuple of Product and Market dimensions, this Essbase block storage calculation script calculates tuples for only those two dimensions, limiting its scope to those members present in a Smart View grid at the time the calculation is executed from Smart View.

```
FIX (
{@GRIDTUPLES(Product, Market)}
)
Marketing(
   Marketing = Marketing + 1;
);
ENDFIX
```

By fixing on only the sparse dimensions named in the tuple, the calculation encompasses a much smaller number of blocks than a default calculation would. However, all members from dimensions not mentioned in the fix (Year, Scenario) are calculated by this calculation script.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Selection of Contextual Sparse Dimensions

Using the @GRIDTUPLES function and a runtime substitution variable, this Essbase block storage calculation script calculates only selected tuples from the grid, based on the sparse dimension selections in the RTSV prompt.

The runtime substitution variable *&DimSelections*, which is defined in the SET RUNTIMESUBVARS block, limits the calculation scope to only the sparse dimensions of the cube, excluding Scenario. The @GRIDTUPLES function used in the FIX statement calls this variable, limiting how many intersections are calculated.

```
SET RUNTIMESUBVARS
            {
            DimSelections = "Version", "Site", "Entity", "Product", "Market"
            <RTSV_HINT><svLaunch>
            <description>List two or more sparse dimensions used for forming
calculation tuples:</description>
            <type>string</type>
            </svLaunch></RTSV_HINT>;
            };
            FIX (
            {@GRIDTUPLES(&DimSelections)}
            )
            Marketing(
            Marketing = Marketing + 1;
            );
            ENDFIX
```

The calculation encompasses an even smaller number of blocks than the previous example, because in this case, the tuple definition extends to more sparse dimensions beyond Product->Market.

To try the examples, download the `CalcTuple_Tuple.xlsx` workbook template from the Technical > Calc section of the **gallery** folder in the **Files** area of the Essbase web interface. Refer to the README worksheet in the workbook for instructions.

Copyright © 2019, 2026, Oracle and/or its affiliates.

12
Model Data in Private Scenarios

Using scenario management, scenario participants can perform what-if analysis to model Essbase data in their own private work areas.

These scenarios can optionally be subject to an approval workflow which includes a scenario owner and one or more approvers. In the workflow, scenario owners merge scenario data with the final cube data only after it is approved.

- Understand Scenarios
- Scenario Workflow
- Enable Scenario Modeling
- Work with Scenarios

Understand Scenarios

Scenarios are private work areas in which users can model different assumptions within the Essbase data and see the effect on aggregated results, without affecting the existing data.

Each scenario is a virtual slice of a cube in which one or more users can model data and then commit or discard the changes.

Scenario-enabled cubes have a special dimension called Sandbox. The sandbox dimension is flat, with one member called Base and up to 1000 other members, commonly referred to as sandbox members. All members in the sandbox dimension are level-0. Sandbox members are named sb0, sb1, and so on. Each sandbox is a separate work area, whereas the Base holds the data currently contained in the cube. A specific scenario is associated with exactly one sandbox member.

```
Sandbox
    Base
    sb0
    sb1
    sb2
```

Base data is the starting point before you use the sandbox to model possible changes. Sandbox data (also known as scenario data) is not committed unless the scenario owner applies it, at which point it overwrites the Base data.

When first created, sandbox member intersections are all virtual and have no physical storage. The physical data from the cube is stored in the Base member slice. Querying new sandbox members dynamically reflects the values stored in the Base.

Only after you update any of the values in a sandbox are your changes stored physically in the sandbox. After you update some values in a sandbox member, queries against the sandbox reflect a mixture of stored sandbox values and values inherited dynamically from the Base.

Changes made in a sandbox are not committed to the Base until you do so explicitly, generally after an approval workflow. See Understand Scenario User Roles and Workflow.

Copyright © 2019, 2026, Oracle and/or its affiliates.

After you're finished with the sandbox, you can put the sandbox through the approval workflow, or you can skip the workflow and commit the updated values to the Base, or reject and discard the sandbox changes.

You must enable hybrid mode for scenario management to work. For queries, it is enabled by default. Do not disable it. For calculations, you also need to enable the HYBRIDBSOINCALCSCRIPT application configuration. See HYBRIDBSOINCALCSCRIPT (or use the SET HYBRIDBSOINCALCSCRIPT calculation command to control it on a per-calculation basis).

Security and filters apply to the Sandbox dimension.

Scenario enabled cubes have a CellProperties dimension that you should ignore, as it is for internal processes. You do not need to modify it nor account for it in calculations, queries, or load rules, and it shouldn't be included in any calculations or other operations.

View and Work with Scenario Data

There are two entry points for viewing and working with scenario data in Smart View.

You can use the Essbase web interface to launch a scenario in Smart View, or you can use a Smart View private connection and work with the scenario data that way.

To analyze data in a scenario, you must have all of the following permissions:

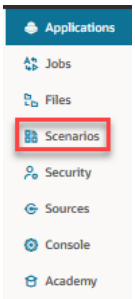
- Be a user provisioned to the application.
- Have a minimum of database access permission for the application (and have a write filter if you want to change data in the sandbox).
- Be a participant in the scenario (created by a user with higher privilege).

View and Work With Scenario Data From the Essbase Web Interface

You can launch Smart View from a scenario in the web interface.

When you do this, because you enter from the scenario, you can only work in Smart View in the sandbox member associated with the scenario from which you entered. The sandbox member is implicit. You will not see it in the Smart View grid.

1. In the Essbase web interface, on the Home page, click **Scenarios**.



2. Click the Excel icon next to the scenario you want to view.
3. Select to open the file.
4. This launches Excel with a Smart View connection to the scenario.

Copyright © 2019, 2026, Oracle and/or its affiliates.

When you do this, the slice of data for that specific scenario is in the worksheet. You can query data only in that scenario. If you have minimum database update permission on the application, you can submit data to the scenario. (When you submit data to a scenario, you are submitting data to one sandbox member).

You can launch a scenario in Smart View from the web interface only on Windows using Firefox, Internet Explorer, or Chrome browsers.

View and Work With Scenario Data From a Smart View Private Connection

You can open Excel and make a private connection to your cube, without starting from the web interface.

When you do this, the sandbox dimension will be in the worksheet, so you can submit data to any sandbox member to which you have access. This is helpful when you are a participant in more than one scenario, but you must explicitly know which sandbox you want to work in.

To see which sandbox member is associated with a scenario, go to the web interface, click on **Scenarios**, click on the scenario name, and view the **General Information** tab.

1. Open Excel.
2. Make a private connection to your scenario-enabled cube.
3. Do an ad hoc analysis.
4. Drill into the Sandbox dimension to view the sandbox members.

Examples

This is a Smart View grid including the Base member and a sandbox member. Sandbox values have not been updated, so they reflect the Base values. Those values are stored only in the Base, not in the sandbox members:

					Base	sb10
Cola	New York	Actual	Jan	Sales	678	678
Cola	New York	Actual	Jan	COGS	271	271

The changed sandbox value below, 500, is stored in a sandbox member. The remaining sandbox value, 271 that was not updated is stored only in the Base:

					Base	sb10
Cola	New York	Actual	Jan	Sales	678	500
Cola	New York	Actual	Jan	COGS	271	271

Below is a grid with multiple sandbox members. If you have the Database Access user role and the appropriate write filter, you can submit data within multiple scenarios simultaneously:

					Base	sb0	sb1
Actual	Jan	Sales	New York	Cola	678	500	600
Actual	Jan	COGS	New York	Cola	271	271	271

Copyright © 2019, 2026, Oracle and/or its affiliates.

About Scenario Calculations

When you want to calculate a sandbox dimension in a scenario-enabled cube, you must include the sandbox members you want calculated in a FIX statement. Otherwise, Essbase calculates only the base member.

By default, in non scenario-enabled BSO cubes, Essbase calculates all members from a dimension unless a FIX statement is used to limit the scope of the calculation to a specific member or group of members from the dimension.

The sandbox dimension is an exception to this behavior. If members from the sandbox dimension are not included in the FIX for a calculation, only the base member from the sandbox dimension is calculated by default. To calculate non-base members from the sandbox dimension, include them in the FIX statement, optionally along with the base member.

When you specify non-base sandbox members in a FIX statement, base is excluded from the calculation unless explicitly added into the FIX.

This behavior is different from calculations on non-sandbox dimensions excluded from the FIX; if you exclude a dimension from your FIX statement, Essbase calculates all members from the implied dimension. Sandbox dimensions are calculated differently, as the intent is usually to calculate either Base or specific sandboxes at a given time. Essbase calculates the Base member values, rather than the working sandbox values, except:

- When the calculation fixes on particular sandbox members.
- When the calculation is executed from a sheet launched from a scenario in the web interface (this is called a scenario-launched sheet). See View and Work With Scenario Data From the Essbase Web Interface.
- When a sandbox cell value is selected in a private connection Smart View sheet and a calculation script is launched.

If you execute a calculation script from a scenario-launched sheet, the calculation runs in the sandbox associated with the scenario as long as no sandbox is explicitly mentioned in the script.

If you're in a sheet opened using a Smart View private connection and you're displaying sandbox and base values, if you highlight any data cell from the sandbox and launch a calculation script without explicit sandbox FIX, the sandbox will implicitly be calculated and Smart View will indicate that the sandbox was calculated. If you highlight a cell from the base member (or highlight no cell), then the base will be calculated when you launch your calc script and Smart View will indicate that the base was calculated.

You can calculate sandbox members using your pre-existing MaxL scripts by using the reserved runtime substitution variable name: ess_sandbox_mbr.

The following statement can be implemented (for your sandbox) in any MaxL script without creating any substitution variable on the server or application.

About Data Loads to Scenario-enabled Cubes

You can load scenario-enabled cubes using data exports taken before enabling the cube for scenarios. The data will load to the base sandbox member.

If you didn’t use column export, then you can’t have outline member changes that would invalidate your data load. If you used column export but your outline has changed, you may need a .rul file to load the data.

Copyright © 2019, 2026, Oracle and/or its affiliates.

About Data Exports from Scenario-enabled Cubes

Scenario-enabled Essbase cubes have a CellProperties dimension that is included in data exports, and must be considered when loading exported data. Also, it is important to understand the behavior of the sandbox dimension when working with exported data.

The following are considerations when exporting data from scenario-enabled cubes:

- If you use the web interface **Jobs** page to export data from a scenario-enabled cube, the resulting data file contains all three members from the CellProperties dimension (EssValue, EssStatus, and EssTID). Do not eliminate any of these columns.
- The data file from the export includes data physically stored in the cube, based on the selection you make: level zero data, all data, or input data.
- If values have been changed in sandboxes, then sandbox values will be in your export.
- In order to load exported data into sandboxes, values for all three CellProperties members (EssValue, EssStatus, and EssTID) must be in the data file.

About Transparent and Replicated Partitions in Scenario-enabled Cubes

Transparent and Replicated partitions connect slices from two Essbase cubes together. This is the case when neither, one, or both cubes are scenario-enabled.

Sandboxes come into use when scenarios are created. However, there is no guarantee that scenarios on partitioned cubes will map to the same sandbox number. The same user may not be a participant in sandboxes in multiple cubes. Introducing scenarios imposes the following limitations:

- If source of a transparent partition is scenario-enabled, target queries will always pull data from the source base sandbox member.
- Write-back between scenario-enabled source and target cubes is only allowed between base members in the cubes, target cube base to source cube base.
  Example: Write-back to source, which is normally enabled from transparent partition target cubes, is disabled for non-base sandbox members of scenario-enabled target cubes. It is a violation of permissions to allow a remote sandbox user to write directly into the base of the source cube.
- For replicated partitions, replication is only possible between source cube base and target cube base.

See Understand Transparent and Replicated Partitions.

About @XREF/@XWRITE in Scenario-enabled Cubes

In a scenario-enabled Essbase cube, you can use the calculation functions @XREF and @XWRITE to reference or write to data in another cube.

@XREF queries a remote cube from a local cube (the cube containing the @XREF statement). If the remote cube is scenario-enabled, @XREF only pulls base data from the remote cube.

@XWRITE updates a remote cube from a local cube (the cube containing the @XWRITE statement). Because @XWRITE writes data into the remote cube, the scope of the @XWRITE statement matters.

For different combinations of scenario-enabled and non-scenario-enabled cubes, @XWRITE behaves in the following ways:

Copyright © 2019, 2026, Oracle and/or its affiliates.

When a scenario-enabled local cube references a non-scenario-enabled remote cube,

- A FIX on the base member in the local cube with an @XWRITE to the remote cube writes the local cube base into the remote cube.
- No FIX on any sandbox member in the local cube with @XWRITE to the remote cube writes the local cube base into the remote cube. If you don't include a sandbox member in the Fix, base is included automatically.
- A FIX on sandbox in the local cube with an @XWRITE to the remote cube returns an error. Writing from a non-base sandbox member into a remote cube is not supported.

When a scenario-enabled local cube references a scenario-enabled remote cube,

- A FIX on the base member in the local cube with an @XWRITE to the remote cube writes the local cube base into the remote cube base.
- No FIX on any sandbox member in the local cube with an @XWRITE to the remote cube writes the local cube base into the remote cube base. If you don't include a sandbox member, base is included automatically.
- A FIX on sandbox in the local cube with an @XWRITE to the remote cube returns an error. Writing from a non-base sandbox member into a remote cube is not supported.

When a non-scenario-enabled local cube references a scenario-enabled remote cube, @XWRITE always updates the remote cube base member.

See Understand @XREF/@XWRITE.

About Audit Trail in Scenario-enabled Cubes

Data audit trail tracks updates made to data in a cube. To work with audit trail in scenario-enabled cubes you, should understand what defines "old" and "new" data values, and the two different entry points for working with sandbox data in Smart View.

This topic assumes you are familiar with the different entry points for viewing scenario data. See:

- View and Work With Scenario Data From a Smart View Private Connection
- View and Work With Scenario Data From the Essbase Web Interface

If you consider the latest data update committed to a cell to be “new” data, and all prior data values for that cell to be “old,” it can help you understand how audit trail works in scenario-enabled cubes.

A new or unused sandbox in a scenario-enabled cube contains no stored values. The values shown to users, such as the values displayed in a spreadsheet, reflect the values stored in the base.

If you use data audit trail on a new scenario-enabled cube, the base values that display in the spreadsheet for the sandbox are considered the “old” values.

When you update values in a sandbox, those values are stored in the sandbox (not in the base). For the purposes of data audit trail, these values are the “new” values.

If you later update these “new” values, audit trail will track the latest changes. It will treat the previous values as “old” and the updated values as “new.”

In summary,

- Old values are the base values reflected in a new sandbox.
- Initially, new values are the updated, stored values in the sandbox.

Copyright © 2019, 2026, Oracle and/or its affiliates.

- Subsequently, updated values are new, and the values they replace are old.

There are two possible entry points for working with data in Smart View:

- Where you open Excel and make a private connection to your cube, without starting from the Essbase web interface.
- Where you launch Smart View from a scenario in the web interface.

When you start by opening Excel and making a private connection to your cube, audit trail works as you would expect with any other data set.

When you launch Smart View from a scenario in the Essbase web interface, audit trail works differently.

- When you export logs to a sheet, the sheet does not show the implicit sandbox member.
- When you launch a new sheet using the **Ad hoc** button below the **Audit Trail** pane, the new sheet does not show the implicit sandbox member, and any changes in that sheet affect the data values for that sandbox member.

About Scenario Limitations

These limitations apply to scenarios and sandbox dimensions.

- Scenarios are not supported on aggregate storage cubes.
- The DATAEXPORT calculation command isn't supported on sandbox members. It is only supported on the Base member.
- When you connect to a scenario from a scenario-launched sheet, MDX queries, MDX inserts, and MDX exports will work with the base instead of working with the sandbox for that scenario.
- Runtime substitution variables with the svLaunch parameter are not supported when you launch the scenario in Smart View from the Essbase web interface. See View and Work With Scenario Data From the Essbase Web Interface.

  Runtime substitution variables with the svLaunch parameter work correctly when you connect to the scenario directly from a private connection. This is because the sandbox member is included in the sheet.

There are a limited number of functions that are not supported in hybrid mode, which is used with scenario-enabled cubes. See Functions Supported in Hybrid Mode.

Enable Scenario Modeling

You enable an Essbase cube for scenario modeling by selecting Enable Scenarios when you create the cube in the web interface, or, by entering a positive number in the Scenario Sandboxes field in an application workbook.

You can create or enable a cube for scenario modeling using one of the following methods:

- Create a Scenario-Enabled Cube
- Create a Scenario-Enabled Sample Cube
- Enable an Existing Cube for Scenario Management
- Create Additional Sandbox Members

Data Audit Trail is not supported on scenario-enabled cubes.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Create a Scenario-Enabled Cube

You can create a scenario-enabled cube in the Essbase web interface by selecting the **Enable Scenarios** check box during the cube creation process.

Scenario-enabled cubes have specialized dimensions required to use scenario management. These include the Sandbox dimension and the CellProperties dimension. CellProperties is considered a hidden dimension in that you do not need to interact with it in any way when performing Essbase tasks such as building cubes, loading data, or calculating cubes.

1. On the Home page, click **Create Application.**
2. On the Create Application dialog box, enter an **Application Name** and a **Database Name** (cube name).
3. Ensure that in **Database Type**, **Block Storage (BSO)** is selected.
4. Select **Enable Scenarios.**
5. Click **OK.**

Create a Scenario-Enabled Sample Cube

You can create a scenario-enabled Essbase cube by importing a scenario-enabled sample application workbook.

1. In the Essbase web interface, click **Import**.
2. Click **Catalog**.
3. Navigate to `gallery > Applications > Demo Samples > Block Storage`.
4. Select **Sample_Basic_Scenario.xlsx** and click **Select**.
5. Provide a unique name and click **OK**.

Enable an Existing Cube for Scenario Management

You can enable an existing cube for scenario management by clicking the **Scenarios** button in the Essbase web interface, and adjusting the number of scenario members you want to create.

If you have at least Application Manager permission, you can enable an existing cube to use scenario modeling. It is best to do so on a copy of the original cube. Existing scripts, rules, and queries will work as before, on the base member. If you need to run them on a sandbox member, you can run them from a scenario-launched sheet.

A scenario launched sheet is an Excel sheet launched from a scenario in the web interface. See View and Work With Scenario Data From the Essbase Web Interface.

1. Enable scenarios.

   In the Redwood Interface,

   a. On the Home page, open the application and then open the database (cube).
   b. On the **General** page, for **Scenarios**, click **Not Enabled.**

   In the Classic Web Interface,

   a. On the Applications page, expand the application.
   b. Click the **Actions** menu to the right of the cube name and select **Inspect.**

Copyright © 2019, 2026, Oracle and/or its affiliates.

   c. On the **General** tab, for **Scenarios**, click **Not Enabled.**

2. Adjust the number of scenarios members (non-base sandbox members) you want to create and click **Ok.**

Create Additional Sandbox Members

By default, a new scenario-enabled cube has 100 sandbox members. You can create additional sandbox members (up to 1000).

1. Navigate to the **General** page.
   - In the Redwood Interface, on the Home page, open the application and then open the database (cube).
   - In the Classic Web Interface,
     a. On the Applications page, expand the application.
     b. Click the **Actions** menu to the right of the cube name, and select **Inspect**.
2. On the **General** page, click the plus sign next to **Scenarios**.
3. Enter the number of sandbox members you want to create.
4. Click **OK**.

Scenario Workflow

To use scenario management for an Essbase block storage cube, first create the dimensions, then enable the cube for scenario management. Afterwards, load the data. Optionally, you may set up an approval process.

General Workflow (Required)

To use scenario management, you must follow this workflow:

1. Create all dimensions (dense and sparse).
2. Enable the cube for scenario management.
3. Load data into the application.

Approval Process (Optional)

You can review a scenario using an optional approval workflow, or you can change data values in the scenario and commit or reject them without going through an approval process.

Scenario status changes and workflow are affected by the number of participants and approvers. When there are no designated approvers, participants do not have the option to submit the scenario for approval. When there are no participants nor approvers, the scenario owner simply makes the changes and applies them.

- Scenario with participants but no approvers:
  1. Scenario owner creates the scenario (Status = In Progress)
  2. Scenario owner and participants make changes in Smart View or the web interface
  3. Scenario owner applies changes to base (Status = Applied)
- Scenario with no approvers and no participants
  1. Scenario owner creates the scenario (Status = In Progress)

Copyright © 2019, 2026, Oracle and/or its affiliates.

2. Scenario owner makes changes in Smart View or the web interface
3. Scenario owner applies changes to base (Status = Applied)

- Scenario with participants and approvers
  1. Scenario is created by owner (Status = In Progress)
  2. Scenario owner, participants and approvers can make changes in Smart View or the web interface
  3. Scenario owner submits the scenario for approval (Status = Submitted)
  4. Scenario is either approved by all approvers or rejected by one or more approvers (Status = Approved or Status=Rejected)
     Rejected status is the same as In Progress status, in that all participants can make changes to reach approved status.
  5. After the scenario reaches approved status (all approvers have approved the scenario), then the scenario owner applies the changes to the base (Status=Applied).
- Enable Email Notifications for Scenario Status Changes
- Create a Scenario
- Model Data
- Submit a Scenario for Approval
- Approve or Reject Scenario Changes
- Apply Data Changes
- Copy a Scenario
- Delete the Scenario
- Understand Scenario User Roles and Workflow

Enable Email Notifications for Scenario Status Changes

If the system administrator has enabled outgoing emails from Essbase, then the appropriate scenario participants receive email notifications for scenario changes.

To set up SMTP email notifications:

1. Log in to Essbase as a system administrator.
2. Click **Console**.
3. Click **Email Configuration**.
4. Click the SMTP Configuration tab.
   SMTP controls outgoing email.
5. Enter your company's SMTP host and port.
6. Enter your company email address and password, of the sender of the notification email.
7. Click **Save**.

When SMTP mail is set up, scenario participants begin receiving emails when their scenarios change status, ownership, priority, or due date.

When users are added to the system, email is an optional field. If it has not been filled out, then that user cannot receive emails even if they participate in scenarios.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Scenario State	Email to	Email Cc	Email Subject
Create scenario	Participant, approver	Owner	You are invited to participate in scenario *<scenario name>*
Submit	Approver	Owner, participant	Scenario *<scenario name>* is submitted for approval
Approve	Owner	Participant, approver	Scenario *<scenario name>* is approved
Reject	Owner	Participant, approver	Scenario *<scenario name>* is rejected by *<user>*
Apply	Participant	Owner, approver	Scenario *<scenario name>* is updated
Delete	Participant, approver, owner	Deleting user	Scenario *<scenario name>* is deleted
Update action Can be a change in ownership, priority, or due date.	Participant, approver	Owner	Scenario *<scenario name>* is updated

An existing scenario can be updated (see *Update action* in the table) to change the owner, the priority, or the due date. If, for example, the scenario's due date is changed, then the participants will receive an email indicating the new due date. The old due date will appear in strike through text, so that it is clear what information about the scenario was updated.

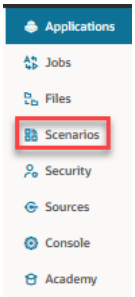
Create a Scenario

To create a scenario, specify general information about your scenario, including creating a scenario name, selecting a due date, selecting an Essbase application and cube, and choosing whether to use calculated values. Then, add users and define whether each user is a participant or an approver.

To create a scenario you must:

- Be a user provisioned to the application with at least Database Update permission, or be the application's owner.
- Be working with a scenario-enabled cube. See Enable Scenario Modeling.

1. In the Essbase web interface, login as a user with Database Update (or higher) permission to at least one application.
2. On the Home page, click **Scenarios**.



Copyright © 2019, 2026, Oracle and/or its affiliates.

3. Click **Create Scenario**.
4. On the **General Information** tab, enter a scenario name and select a **Priority** (optional), **Due Date**, **Application**, and **Database** (cube). You will only see applications for which you have minimum database update permission.
5. Turn on **Use Calculated Values** if you want to merge calculated values to base values when running calculation scripts on scenarios.
6. (Optional) Enter a description.
7. On the **Users** tab, click **Add** for a list of users.
8. Add the users that you want.
9. Close the **Select Users/Groups** dialog box.
10. For each user, keep the default (**Participant**), or select **Approver**.

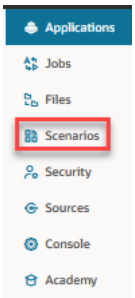
    Scenario user roles determine the workflow for the scenario.
11. Save your changes.

See also: Understand Scenario User Roles and Workflow.

Model Data

As a scenario user, you can model data slices in your own scenario.

1. In the Essbase web interface, on the Home page, click **Scenarios**.



2. On the Scenarios page, locate the scenario in which you want to model data.
   - You can search for the scenario by name in the **Search** field.
   - You can select your application from the **All Applications** drop-down list and search within that application.
   - After selecting the application, you can further narrow your search by selecting the database (cube) from the **All Databases** drop-down list and searching within that specific cube.
3. Launch Smart View by clicking the Excel icon before the scenario name.
4. Make data changes and perform your what-if analysis in Smart View.

   If you change and submit values and decide you want to go back to the base values, you can revert to the base by typing #Revert in the changed cells and choosing **Submit Data** on the Smart View Essbase ribbon.

   If a cell in the base has a value, and you want the corresponding cell in the scenario to be #Missing, you can send #Missing to the scenario or you can delete the value in Smart View and select **Submit Data** on the Smart View Essbase ribbon.

Copyright © 2019, 2026, Oracle and/or its affiliates.

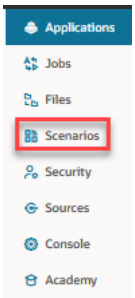
5. Continue this process until you're ready to submit data for approval.

If a calculation has been run on a sandbox and the changes are not acceptable, request from your application designer a calc script to revert the changes, or request a new sandbox.

Submit a Scenario for Approval

After you submit a scenario for approval, no one will be able to write to that scenario.

1. In the Essbase web interface, log in as the application owner or the scenario owner.
2. On the Home page, click **Scenarios**.



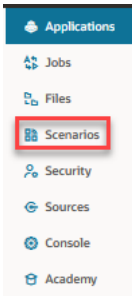
3. Under **Actions**, click **Submit** .
4. (Optional) Enter a comment.
5. Click **OK.**

After a scenario is submitted for approval, the scenario approver can approve or reject the data changes.

Approve or Reject Scenario Changes

After the owner of the scenario submits for approval, the approver has the option to approve or reject scenario changes, and the scenario owner is notified of the action. You must be logged in as an approver to have the options to approve or reject a scenario.

1. In the Essbase web interface, on the Home page, click **Scenarios**.



2. Next to the submitted scenario, under **Actions**, click **Approve** or **Reject** .
3. Enter a comment on the Approve or Reject dialog box.

After a scenario is approved, the scenario owner can apply the changes to the cube.

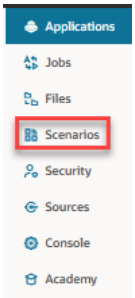
Copyright © 2019, 2026, Oracle and/or its affiliates.

Apply Data Changes

When you apply data changes in a scenario-enabled cube, the changes stored within the scenario overwrite the base data. You can apply data changes from the Scenario page in the Essbase web interface, or by using the DATAMERGE calculation command.

To apply data changes as part of your scenario workflow,

1. In the Essbase web interface, on the Home page, click **Scenarios**.



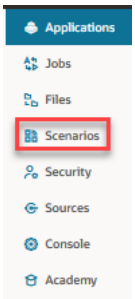
2. Next to the approved scenario, under **Actions**, click **Apply**.
3. When prompted, leave an optional comment and confirm your selection.

- You can also apply data changes using the DATAMERGE calculation command.
- After a scenario is applied, you can delete the scenario to reuse the sandbox for that scenario.
- Database managers and higher can execute a calculation script to perform a DATAMERGE. They do not need to be designated as scenario approvers in order to do so.
- After a scenario is applied, it can be re-applied, but it cannot be changed.

Copy a Scenario

If you have the service administrator role, or if you are a scenario user (participant, approver, or owner), you can copy a scenario. You can copy scenarios at any point in the scenario workflow, prior to Delete Scenario. The approval state of the copied scenario is reset to In Progress.

1. In the Essbase web interface, on the Home page, click **Scenarios**.



2. Click the **Actions** menu for the scenario you want to copy, and click **Copy**.

Copyright © 2019, 2026, Oracle and/or its affiliates.

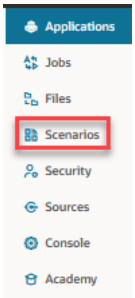
3. Enter the scenario name and select what scenario components to copy from **Approvers**, **Participants**, **Comments**, and **Data**.
4. Click **OK**.

Delete the Scenario

As there are a fixed number of available sandboxes in a scenario-enabled Essbase cube, you may need to clean up inactive scenarios. After a scenario is deleted, its sandbox is returned to the pool of available sandboxes.

To delete a scenario using the Essbase web interface,

1. In the Essbase web interface, on the Home page, click **Scenarios**.



2. Click the **Actions** menu for the scenario you want to delete, and select **Delete.**

Understand Scenario User Roles and Workflow

You can review a scenario using an optional approval workflow.

Scenario user role assignments determine the workflow for scenarios. You must have at least one approver to enable the scenario workflow. Without an approver, participants do not have the option to submit the scenario for approval, for example, and there is no option to approve or reject the scenario.

The only action for scenarios without at least one approver is Apply. Without an approver, the scenario owner can still change data values in the scenario and apply data changes to the cube (or reject them), without going through an approval process.

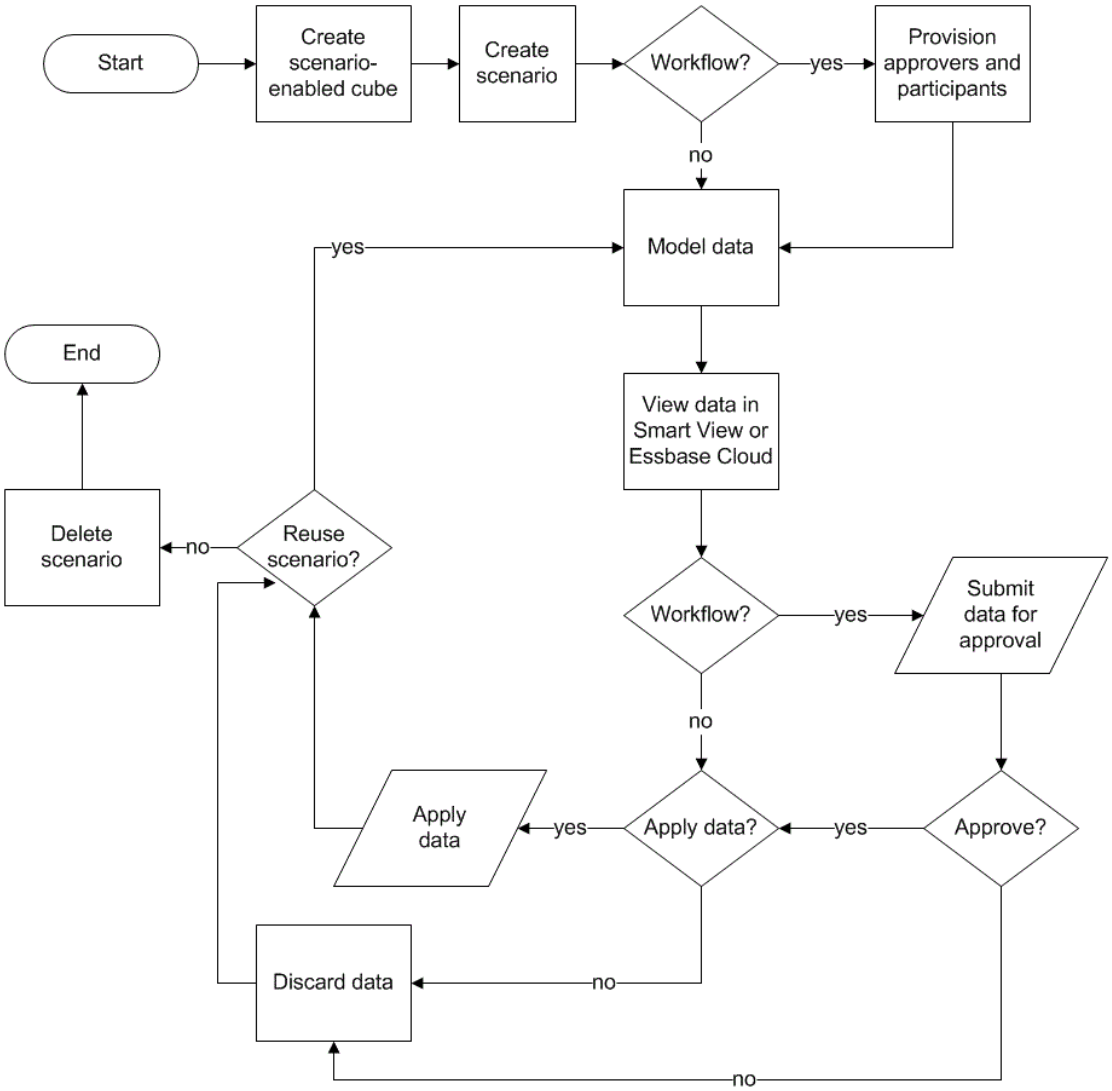
Participants can participate in a what-if analysis. They must have Database Update or Database Access user role. Adding participants is not mandatory.

Approvers monitor the process, and approve or reject scenarios. They must have Database Access or higher role. Scenarios can have multiple approvers, in which case each one must approve the scenario before it can be submitted.

Participants and approvers with the Database Access user role cannot write to a scenario until they are granted write access through a filter.

Participants and approvers are not mandatory. The scenario owner can change data values in the scenario and commit data changes to the cube (or reject them) without designating participants or approvers.

Copyright © 2019, 2026, Oracle and/or its affiliates.



Work with Scenarios

After you enable scenario modeling for an Essbase cube, you can work with scenario data. You can view base member data, compare scenario values to base values, set scenario cells to #Missing, revert scenario values back to base values, and aggregate scenario data.

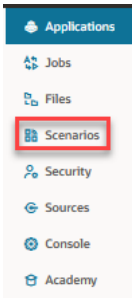
- View Base Member Data
- Compare Scenario Values to Base Values
- Set Scenario Cells to #Missing
- Revert Scenario Values Back to Base Values
- Understand When to Aggregate Sandbox Dimensions

View Base Member Data

From the Essbase web interface, you can launch an Excel sheet showing base data for a scenario.

Copyright © 2019, 2026, Oracle and/or its affiliates.

1. In the Essbase web interface, on the Home page, click **Scenarios**.



2. Click the **Actions** menu for the scenario you want to view, and click **Show Base Data.**
3. Click the downloaded link to launch Smart View.

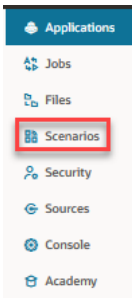
The Excel sheet that is launched shows base data for the cube. It does not show sandbox data.

Compare Scenario Values to Base Values

If you are the owner, approver or participant for a given scenario, you can view scenario and base values in a spreadsheet or in the Essbase web interface to compare models.

Compare Values in Excel

1. In the Essbase web interface, on the Home page, click **Scenarios**.



2. From the **Actions** menu, select **Show Changes in Excel.**
3. Click on the downloaded link to open the Smart View link.
4. You can view values for both the scenario and base members in the spreadsheet.

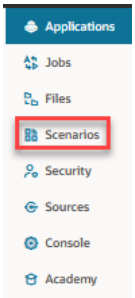
	A	B	C	D	E	F	G
1						Base	sb10
2	Cola	New Yo	Actual	Jan	Sales	678	700
3	Cola	Massacl	Actual	Jan	Sales	494	500
4	Cola	Florida	Actual	Jan	Sales	210	250
5	Cola	Connec	Actual	Jan	Sales	310	350
6	Cola	New Ha	Actual	Jan	Sales	120	150
7	Cola	East	Actual	Jan	Sales	1812	1950

Copyright © 2019, 2026, Oracle and/or its affiliates.

- In column G, sb10 is the scenario (or sandbox) member.
- In column F, Base shows the base values.
- In the scenario, values for sb10 on rows 2 through 6 have been changed, and you can see the aggregated result in row 7.

Compare Values in the Web User Interface

1. In the Essbase web interface, on the Home page, click **Scenarios**.



2. From the **Actions** menu, select **Show Changes.**

   The **Data Changes** dialog box is empty if no data changes have been made.

Compare the scenario to the base in order to determine your next steps. For example, you might choose to change the status of the scenario to approved based on this information.

Set Scenario Cells to #Missing

You can set scenario cells to #Missing even if the corresponding base cells have values.

To set a scenario cell to #Missing:

1. Type #Missing in the cell or delete the cell contents.
2. Select **Submit Data** on the Smart View ribbon.

Example

1. Initially, the value in sb1 is an exact mirror of the value in the base.

				Base	sb1
				Jan	Jan
100-10	New York	Sales	Actual	678	678

2. Enter #Missing in sb1 (or delete the cell contents) and submit data.

				Base	sb1
				Jan	Jan
100-10	New York	Sales	Actual	678	#Missing

3. Refresh the sheet. See that sb1 is #Missing.

Copyright © 2019, 2026, Oracle and/or its affiliates.

				Base	sb1
				Jan	Jan
100-10	New York	Sales	Actual	678	#Missing

Revert Scenario Values Back to Base Values

In a worksheet connected to a scenario-enabled Essbase cube, you can revert the scenario values back to the base by typing #Revert in the changed cells and clicking **Submit Data** on the Smart View ribbon.

Initially, scenario values are not stored and they are an exact mirror of the base values. After you change the scenario values in Excel and submit the changes to the cube, the scenario values are stored, and they are different from the base, but you can still revert them back to the base values.

To revert scenario values back to the base:

1. In Excel, type #Revert in the scenario cells you want to revert to the base.
2. Click **Submit Data** on the Smart View ribbon.

The selected scenario values are updated to the base values.

Example

1. Initially, the value in sb1 is an exact mirror of the value in base.

				Base	sb1
				Jan	Jan
100-10	New York	Sales	Actual	678	678

2. Submit a new value, 100, to sb1.

				Base	sb1
				Jan	Jan
100-10	New York	Sales	Actual	678	100

3. Submit #Revert to sb1.

				Base	sb1
				Jan	Jan
100-10	New York	Sales	Actual	678	#Revert

4. Refresh the sheet. See that sb1 again reflects the base value of 678.

				Base	sb1
				Jan	Jan
100-10	New York	Sales	Actual	678	678

Copyright © 2019, 2026, Oracle and/or its affiliates.

Understand When to Aggregate Sandbox Dimensions

As you model in scenarios, you will need to determine whether or not to calculate within each sandbox.

Submit data changes to the sandbox and calculate as little other data as possible, just enough data to allow users to validate their work. This preserves the storage efficiency of the sandbox design.

For example, when all upper level members in a cube are dynamic calc, aggregations in the form of calculation script are not needed.

If you have stored upper level members, limit the scope of any sandbox calculation to the minimum needed for users to do their work.

Example: Calculate Scenarios with Dynamic Upper Level Members

Dynamic hierarchies (both dense and sparse) aggregate automatically, and users making changes in sandboxes see their changes immediately.

Let's look at an example from the Sample_Scenario.Basic block storage demo application.

Assume that Product and Market are dynamic hierarchies with data stored only at level zero, and that a scenario is created using sandbox dimension member sb0.

When the sandbox is newly created, values for sb0 are the same as the values for Base. This is because sandbox members are virtual, reflecting base values until users submit changes to them.

	A	B	C	D
1			Budget	Budget
2			Sales	Sales
3			Jan	Jan
4			Base	sb0
5	California	Cola	840	840
6	Oregon	Cola	200	200
7	Washington	Cola	160	160
8	Utah	Cola	160	160
9	Nevada	Cola	90	90
10	West	Cola	1450	1450

After modifying Sales->Budget->Jan->Cola data in member sb0, we immediately see that the dynamic sandbox member, West (in D10) aggregates to the correct total by using a combination of stored members from Base and sb0.

Values for Oregon, Utah and Nevada are stored in the Base sandbox member. Values for California and Washington have been submitted by scenario participants and are stored in the sb0 sandbox member. The total for West->Cola->sb0 aggregates dynamically using these stored values.

Copyright © 2019, 2026, Oracle and/or its affiliates.

	A	B	C	D
1			Budget	Budget
2			Sales	Sales
3			Jan	Jan
4			Base	sb0
5	California	Cola	840	900
6	Oregon	Cola	200	200
7	Washington	Cola	160	200
8	Utah	Cola	160	160
9	Nevada	Cola	90	90
10	West	Cola	1450	1550

You can also use calculation scripts in sandboxes. Assume that Oregon is meant to be budgeted as 80% of California. The following calculation script can do this:

```
FIX("Jan", "Budget", "Cola", "Sales")
"Oregon"="California"*.8;
ENDFIX
```

When a scenario participant launches an Excel worksheet from the web interface and runs this calculation, sb0 is the default sandbox member calculated and the value for member Oregon is updated:

	A	B	C	D
1			Budget	Budget
2			Sales	Sales
3			Jan	Jan
4			Base	sb0
5	California	Cola	840	900
6	Oregon	Cola	200	720
7	Washington	Cola	160	200
8	Utah	Cola	160	160
9	Nevada	Cola	90	90
10	West	Cola	1450	2070

This view is not from a scenario-launched sheet, but rather from a Smart View private view, where Base and sb0 can both be represented on the sheet.

Example: Calculate Scenarios with Stored Upper Level Members

In some cases, a sparse or dense hierarchy may have stored upper level members, and aggregations on level- or generation-based calculations could be required.

Continuing from the last grid of the previous example, assume now that upper level members in the Market dimension are stored, rather than dynamic.

If we change the value for Oregon to 250, the West member will need to be re-calculated before we can see correct results:

Copyright © 2019, 2026, Oracle and/or its affiliates.

	A	B	C	D
1			Budget	Budget
2			Sales	Sales
3			Base	sb0
4			Jan	Jan
5	California	Cola	840	900
6	Oregon	Cola	200	250
7	Washington	Cola	160	200
8	Utah	Cola	160	160
9	Nevada	Cola	90	90
10	West	Cola	1450	2070

The following calc script can be used to aggregate the Market dimension in the sandbox, when executed from a scenario-launched excel sheet:

```
AGG("Market");
```

	A	B	C	D
1			Budget	Budget
2			Sales	Sales
3			Jan	Jan
4			Base	sb0
5	California	Cola	840	900
6	Oregon	Cola	200	250
7	Washington	Cola	160	200
8	Utah	Cola	160	160
9	Nevada	Cola	90	90
10	West	Cola	1450	1600

Copyright © 2019, 2026, Oracle and/or its affiliates.

13

Audit Data, Security, Artifact Changes, and LCM Events

Essbase auditing tracks changes to cube data, server-level security, LCM events, artifact changes, and MaxL statements that are run on the server, including imports.

Use cube-level data auditing to track updates made to data values, including changes to Linked Reporting Objects (LROs), such as adding notes, attaching files, and referencing URLs. You can export the audit log to an Excel spreadsheet.

Use server-level auditing to track security, LCM events, artifact changes, and executed MaxL statements including imports of data or dimensions. The tracking information is saved to a security audit log file or streamed to an external database. You configure what events to track by defining an auditing policy file.

- Track Data Changes
- Audit Security, Artifact Changes, and LCM Events

Track Data Changes

You can create an audit trail to track updates made to Essbase data values, including changes to Linked Reporting Objects (LROs), such as adding notes, attaching files, and referencing URLs. You can export the log to an Excel spreadsheet.

To view data audit trail records, you must be at least a power user with Database Update permission on the application. You can only view those records where your user name matches the user name registered in the audit records. To delete data audit trail records, you must be at least a power user with Application Manager permission on the application. See Understand Your Access Permissions in Essbase.

- Turn on Data Audit Trail and View the Data Audit Trail
- Link a Reporting Object to a Cell
- Export Logs to a Sheet
- Refresh the Audit Log
- View and Manage Audit Trail Data in the Essbase Web Interface

Turn on Data Audit Trail and View the Data Audit Trail

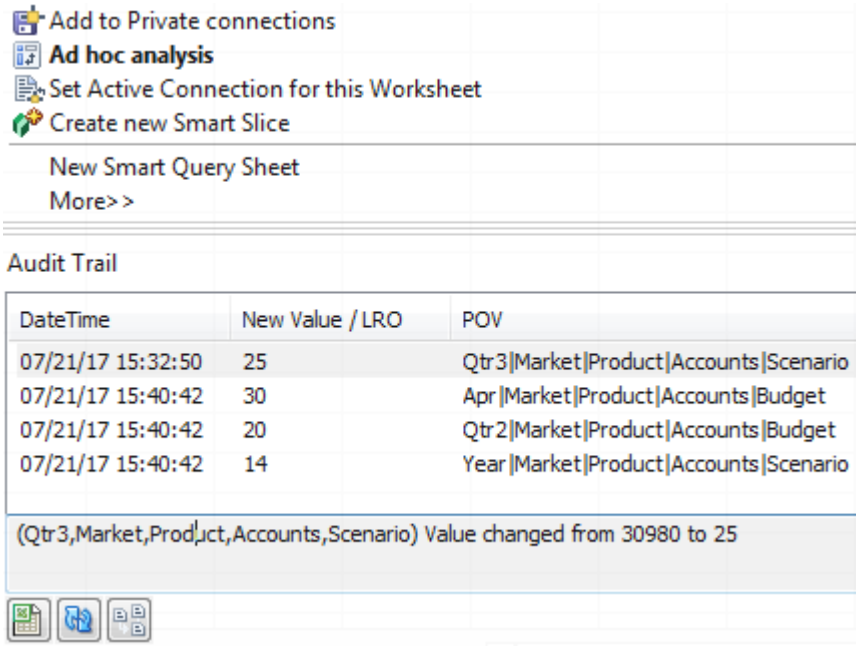
Enable data audit trail for Essbase by adding AUDITTRAIL DATA as an application level configuration setting.

1. To turn on Data Audit Trail, add the following to the application configuration parameters: AUDITTRAIL DATA.
2. Perform ad hoc analysis through Smart View, make data changes through Smart View, and click on **Submit** - this results in an audit record being stored in the Essbase repository schema, in the table ESSBASE_DATA_AUDIT_TRAIL.

Copyright © 2019, 2026, Oracle and/or its affiliates.

When doing ad hoc analysis, there are many ways of getting a particular Point of View (POV) onto the grid. One of them is by using the POV toolbar, which allows you to zoom in on certain members in one or more dimensions. Refer to Selecting Members from the POV Toolbar in the Smart View documentation.

3. With Data Audit Trail enabled, you can view the audit trail in the connection Panel in Smart View. Under the connection information, click on the menu of operations under **More** to locate a menu option titled **Audit Trail**. Click on **Audit Trail** to view the data audit trail records for a cube.



4. The audit trail record shows the date and time of the change in the first column, the new value or the linked reporting object in the second column, and the POV in the third column. The time corresponds to your time zone. Click on an item in the audit trail for a description of the change.

5. You can display a sheet with the new POV and refreshed data value by clicking **Ad hoc** below the **Audit Trail** pane. When you click on subsequent audit records and click this icon, a different sheet is displayed with the POV for that audit record and refreshed data for that POV. This way, you can do further analysis on targeted data.

More Information

For information about application configuration, refer to Set Application-Level Configuration Properties.

For the limit on number of records displayed, refer to Other Size or Quantity Limits.

To use the REST API to fetch audit records, refer to Get Audit Data.

Link a Reporting Object to a Cell

You can add a linked reporting object (LRO) to a cell in the Essbase database. You can add a note to a cell, attach a file, or reference a URL. When you make these changes, the cells are highlighted in your cube.

LRO additions and updates are tracked in the data audit trail.

Copyright © 2019, 2026, Oracle and/or its affiliates.

See these topics in the *Working with Oracle Smart View for Office* on how to link reporting objects to cells:

- Linked Reporting Objects
- Attaching a Linked Reporting Object to a Data Cell
- Launching a Linked Reporting Object from a Data Cell

Export Logs to a Sheet

You can easily export your logs to a new Excel sheet just by clicking an icon.

Export your log onto a new sheet using **Export** . Click this icon to export the logs with all the details for each entry onto a new sheet that looks like this:

	A	B	C	D	E	F	G	H	I	J
1	User	DateTime	Cell Note	New Value	Old Value	Operation	POV			
2	weblogic	07/21/17 15:32:50		25	30980	INPUT	Qtr3|Market|Product|Accounts|Scenario			
3	weblogic	07/21/17 15:40:42		30	9777.5	INPUT	Apr|Market|Product|Accounts|Budget			
4	weblogic	07/21/17 15:40:42		20	29903.1	INPUT	Qtr2|Market|Product|Accounts|Budget			
5	weblogic	07/21/17 15:40:42		14	133980	INPUT	Year|Market|Product|Accounts|Scenario			

Once exported, you can re-sort columns or remove them to show the information you want to analyze.

Refresh the Audit Log

You can refresh the Essbase Server audit log to see the latest tracked changes.

To refresh the security audit log to view the latest changes made on the Essbase Server, click **Refresh** in the audit log spreadsheet.

	A	B	C	D	E	F	G	H	I	J
1	User	DateTime	Cell No	New Value	Old Value	Operation	POV			
2	weblogic	07/21/17 15:32:50		25	30980	INPUT	Qtr3|Market|Product|Accounts|Scenario			
3	weblogic	07/21/17 15:40:42		30	9777.5	INPUT	Apr|Market|Product|Accounts|Budget			
4	weblogic	07/21/17 15:40:42		20	29903.1	INPUT	Qtr2|Market|Product|Accounts|Budget			
5	weblogic	07/21/17 15:40:42		14	133980	INPUT	Year|Market|Product|Accounts|Scenario			
6	weblogic	07/23/17 16:20:13		45	-403	INPUT	Jul|East|Visual|Accounts|Variance			
7	weblogic	07/23/17 16:20:13		55	-271	INPUT	Sep|South|Visual|Accounts|Variance			
8	weblogic	07/23/17 16:20:13		65	-1840	INPUT	Qtr4|South|Visual|Accounts|Variance			

View and Manage Audit Trail Data in the Essbase Web Interface

You can view audit trail data in the Essbase web interface. You can also export the data to an Excel sheet (in .csv format), purge the data before a specific date, or purge all of the audit trail data.

1. To view and manage audit trail data:

   In the Redwood Interface,

   a. On the Applications page, open the application, then open the cube.

   b. Select the **Audit Data** tab.

   In the Classic Web Interface,

Copyright © 2019, 2026, Oracle and/or its affiliates.

   a. On the Applications page, expand the application.
   b. Click the **Actions** menu to the right of the cube name and select **Inspect**.
   c. Select the **Audit Trail** tab.
2. You can:
   - View audit trail data.
   - Export the data to a CSV file.
   - Purge the audit trail data until a specific date.
   - Purge all of the audit trail data.

To purge data audit trail records, you must be a power user with Application Manager permission on the application.

Audit Security, Artifact Changes, and LCM Events

Service administrators can enable security auditing to track changes made to the Essbase server.

Based on parameters you specify in an auditing policy file, Essbase gathers information about changes to system-level security, artifacts, LCM events, and executed MaxL statements (including imports). Essbase consolidates the tracked information into an audit log file, or streams it to an external database. Tracked information about each event includes time, client, user, artifacts affected, duration, ID, application and database name, status, and a description.

You enable server-level auditing of these events using the AUDITTRAIL SECURITY Essbase configuration setting.

Video

- Workflow to Enable Security Auditing for Essbase Server
- About the Auditing Policy File
- Security Auditing Events

Workflow to Enable Security Auditing for Essbase Server

This workflow explains how to enable security auditing on the Essbase Server using AUDITTRAIL SECURITY. After you enable auditing, you define EssbaseSecurityAuditLogPolicy. You can have Essbase write audit records to a CSV file or stream them to an external database.

In this workflow, Oracle Database is the external database, but you can also use SQL Server, MySQL, or DB2.

To complete the workflow, you must be a system administrator, and you will need to access the <*Essbase Config Path*> on the Essbase Server machine.

This path contains files you will need to edit:

- The essbase.cfg configuration file
- A default security auditing policy file

1. Enable auditing of server events by adding the following configuration to essbase.cfg on the Essbase server machine:

   AUDITTRAIL SECURITY

Copyright © 2019, 2026, Oracle and/or its affiliates.

After you update the configuration, restart Essbase.

Refer to Set Server-Level Configuration Properties and Start, Stop, and Check Servers.

2. A default policy file (XML) is created on the Essbase server. This file, `EssbaseSecurityAuditLogPolicy.xml`, is in the path you specified during the configuration phase of deployment to store Essbase configuration (the <*Essbase Config Path*>, which is also where essbase.cfg resides).

   The default policy file created has these contents:

```
<?xml version="1.0" encoding="UTF-8"?>
<security-audit-policy>

<audit_events_to_capture>LOGIN,LOGINAS,LOGIN_FAIL,LOGOUT,SERVICE_ROLE_ASSIG
N,SERVICE_ROLE_REVOKE,APPLICATION_ROLE_ASSIGN,APPLICATION_ROLE_REVOKE,ARTIF
ACT_UPLOADED,ARTIFACT_MODIFIED,ARTIFACT_DELETED,ARTIFACT_CREATE,ARTIFACT_RE
NAMED,APPLICATION_DELETED,APPLICATION_CREATE,APPLICATION_RENAMED,DATABASE_D
ELETED,DATABASE_CREATE,DATABASE_RENAMED,LCM_EXPORT_START,LCM_EXPORT_END,LCM
_IMPORT_START,LCM_IMPORT_END,LCM_IMPORT_FAIL,DATA_LOAD_MAXL,LOAD_DATA_JOB_S
TART,LOAD_DATA_JOB_END,LOAD_DATA_JOB_FAILED,DELETE_SESSION,EXECUTE_MAXL,APP
LICATION_SET_ACTIVE,APPLICATION_START,APPLICATION_STOP,DATABASE_START,DATAB
ASE_STOP</audit_events_to_capture>
   <audit_sinks>
      <audit_sink>
         <audit_sink_type>CSV</audit_sink_type>
         <max-file-size>50000000</max-file-size>
         <roll-nos>100</roll-nos>
      </audit_sink>
   </audit_sinks>
</security-audit-policy>
```

   CSV is the default audit sink type. If you are using the default (CSV) audit sink type, and you want to test that the audit details are written to the security audit log CSV file,

   a. Perform an action that is an auditable event, such as creating an application. You can select any action listed in the <audit_events_to_capture> section of your policy.

   b. SSH to the Essbase server.

   c. Navigate to `<DOMAIN_HOME>/servers/serverName/logs/essbase/`. If you do not know where <*DOMAIN_HOME*> is, refer to Environment Locations in the Essbase Platform.

   d. Open and review the file, `SecurityAuditLog_n.csv`.
      Example of a security audit log CSV file:

Time	Client	User Nam	Session ID	Event Typ	Artifact Type	Artifact Name	Additional Info	Description	ID	Duration	Application	Database	Status
30:02.2	phoenix	admin		LOGIN				User [admir	6348b503-	0			
30:08.0	phoenix	admin		LOGIN				User [admir	11408d1f-	0			
30:19.5	phoenix	admin		LOGIN				User [admir	34aa8859-	0			
30:19.6	phoenix	admin		LCM_IMP	Application	17	/users/admin/t	LCM import	2c22aaa3-	0	new1		
30:21.9	phoenix	admin		APPLICATI	APPLICATION	new1	Application :ne	Application	a4dc47bf-	0	new1		
30:21.9	phoenix	admin		APPLICATI	APPLICATION	new1	created Applica	Application	7bc4351a-	0	new1		
30:22.1	phoenix	admin		LOGIN				User [admir	8af964cd-	0			
30:22.1	phoenix	admin		APPLICATI	User	user3		User/Group	ae5fb53c-	0	new1		
30:22.1	phoenix	admin		APPLICATI	User	user3		User/Group	0a6afd66-	0	new1		
30:22.3	phoenix	admin		APPLICATI	APPLICATION	new1	Application :ne	Application	08fb2da3-	0	new1		
30:23.8	phoenix	admin		APPLICATI	APPLICATION	new1	Application :ne	Application	978b422e-	0	new1		
30:24.0	phoenix	admin		DATABASE	DATABASE	new1	Application :ne	Database [B	e8b3998a-	0	new1	Basic	
30:24.0	phoenix	admin		DATABASE	DATABASE	Basic	created databa	Database [B	e39ebf84-	0	new1	Basic	
30:24.3	phoenix	admin		APPLICATI	APPLICATION	new1	Application :ne	Set active ca	ed4c3aeb-	0	new1	Basic	

3. If you want the security audit trail streamed to an external database,

Copyright © 2019, 2026, Oracle and/or its affiliates.

a. Create a connection to the external source. Refer to Create a Global Connection and Datasource or Create an Application-Level Connection and Datasource.

b. Edit the policy file to change the audit sink to DATABASE.

c. Add a <db_connection_name> parameter within the <audit_sink> parameter. The value of the <db_connection_name> parameter should be the exact name of the connection created in substep a above.

Example of edited audit policy for streaming security audit trail to Oracle Database:

```
<?xml version="1.0" encoding="UTF-8"?>
<security-audit-policy>

<audit_events_to_capture>LOGIN,LOGINAS,LOGIN_FAIL,LOGOUT,SERVICE_ROLE_ASSIG
N,SERVICE_ROLE_REVOKE,APPLICATION_ROLE_ASSIGN,APPLICATION_ROLE_REVOKE,ARTIF
ACT_UPLOADED,ARTIFACT_MODIFIED,ARTIFACT_DELETED,ARTIFACT_CREATE,ARTIFACT_RE
NAMED,APPLICATION_DELETED,APPLICATION_CREATE,APPLICATION_RENAMED,DATABASE_D
ELETED,DATABASE_CREATE,DATABASE_RENAMED,LCM_EXPORT_START,LCM_EXPORT_END,LCM
_IMPORT_START,LCM_IMPORT_END,LCM_IMPORT_FAIL,DATA_LOAD_MAXL,LOAD_DATA_JOB_S
TART,LOAD_DATA_JOB_END,LOAD_DATA_JOB_FAILED,DELETE_SESSION,EXECUTE_MAXL,APP
LICATION_SET_ACTIVE,APPLICATION_START,APPLICATION_STOP,DATABASE_START,DATAB
ASE_STOP</audit_events_to_capture>
   <audit_sinks>
      <audit_sink>
         <audit_sink_type>DATABASE</audit_sink_type>
           <db_connection_name>OraclePDB</db_connection_name>
      </audit_sink>
   </audit_sinks>
</security-audit-policy>
```

4. Test that the audit details streamed to the Database.

   a. Perform an action that is an auditable event, such as creating an application. You can select any action listed in the <audit_events_to_capture> section of your policy. Essbase should create an audit table named ESSBASE_SECURITY_AUDIT_EVENT_LOG in the external database schema.

   b. Log in to the external RDBMS and run a query to check for the presence of the table. For example, log in to SQL Developer and run

   ```
   select * from ESSBASE_SECURITY_AUDIT_EVENT_LOG
   ```

5. Use a data visualization tool to view and analyze the security audit records. You can use Smart View, Oracle Data Desktop (available with an Oracle Technology Network license), open-source visualization tools from open source or your non-Oracle database vendor.

About the Auditing Policy File

The auditing policy is defined in an XML file that you can edit to suit your needs. In this file, you can specify which Essbase Server events to track and whether to write the data to a security audit log or stream it to an external database. If you are writing data to an audit log, you can indicate the maximum file size and the number of security audit log files to keep.

Essbase creates `EssbaseSecurityAuditLogPolicy.xml` when you restart Essbase after enabling security auditing. You can then edit the file as needed to refine the auditing policy. The file is located in the path you specified during the configuration phase of deployment to store Essbase configuration (the <*Essbase Config Path*>, which is also where essbase.cfg

Copyright © 2019, 2026, Oracle and/or its affiliates.

resides). If you don't know where this is in your environment, refer to Environment Locations in the Essbase Platform for an explanation.

To edit the auditing policy file,

1. Navigate to `EssbaseSecurityAuditLogPolicy.xml`. The file is located in the application directory specified during the configuration phase of Essbase deployment.
2. Open it in a text editor.
3. Edit the audit sink, logging details, and events to track.
   a. Optionally add `<audit_sink_type>`DATABASE`</audit_sink_type>` if you want to stream data to an external database.
   b. If you indicated an audit sink type of DATABASE in step a, on the following line, add <db_connection_name>*ConnectionName*</db_connection_name> with the name of the database connection you defined in the Workflow to Enable Security Auditing for Essbase Server.
   c. If writing data to an audit log file, Optionally change the maximum file size using `<max-file-size>`*n*`</max-file-size>`, where n = the number of bytes . The default is 50000000 bytes.
   d. If writing data to an audit log file, indicate how many security audit log CSV files to save using `<roll-nos>`*n*`</roll-nos>`, where n = the number of files.
   e. Indicate which audit events you want to capture, using `<audit_events_to_capture>`*`events_list`*`</audit_events_to_capture>`.

The events that you indicate in the auditing policy file are tracked in a security audit log file or streamed to an external database.

You can indicate the following events to capture in the auditing policy file:

Event	Description
LOGIN	User [x] logged in successfully
LOGIN_AS	User [x] logged in as [y]
LOGOUT	User [x] logged out
LOGIN_FAIL	User [x] login failed
SERVICE_ROLE_ASSIGN	Assigned Essbase service role [x] to [y]
SERVICE_ROLE_REVOKE	Revoked Essbase service role [x] from [y]
APPLICATION_ROLE_ASSIGN	User/Group [x] has been provisioned the role [y] on the application [z]
APPLICATION_ROLE_REVOKE	User/Group [x] has been revoked from the role [y] on the application [z]
ARTIFACT_CREATE	Artifact [x] of type [y] created
ARTIFACT_UPLOADED	Artifact upload request called for application [a] database [b] object name [c] and object type [d]
ARTIFACT_MODIFIED	Artifact [x] of type [y] modified
ARTIFACT_DELETED	Artifact [x] of type [y] deleted
ARTIFACT_RENAMED	Artifact [x] of type [y] renamed to [z]
APPLICATION_DELETED	Application [x] deleted
APPLICATION_CREATE	Application [x] created
APPLICATION_RENAMED	Application [x] renamed to [y]
DATABASE_DELETED	Database [x] deleted in application [y]
DATABASE_CREATE	Database [x] created in application [y]

Copyright © 2019, 2026, Oracle and/or its affiliates.

Event	Description
DATABASE_RENAMED	Database [x] renamed to [y] in application [z]
LCM_EXPORT_START	LCM export job started with file name [x]
LCM_EXPORT_END	LCM export job completed with file name [x] and job status [y]
LCM_IMPORT_START	LCM import started for application [x] with file name [y]
LCM_IMPORT_END	LCM import completed for application [x] with file name [y]
LCM_IMPORT_FAIL	LCM import failed for application [x] with file name [y]
DATA_LOAD_MAXL	The MaxL import data statement executed for application [x] and database [y] by user [z]
EXECUTE_MAXL	MaxL statement [x] executed from user [y]
LOAD_DATA_JOB_START	Data load job started using data file [x] and rule file [y]
LOAD_DATA_JOB_END	Data load job for data file [x] and rule file [y] completed with status [z]
LOAD_DATA_JOB_FAILED	Data load job failed due to [x]
DELETE_SESSION	Session [x] deleted

Security Auditing Events

Essbase security auditing events are tracked either in a security audit log file, or streamed to an external database, depending on which audit sink you indicate in the auditing policy file.

For instructions on opening the security audit log or the audit table in the in the external database schema, see Workflow to Enable Security Auditing for Essbase Server.

The security audit log and the audit table include the following information (when it is applicable) about each event:

- Time - when the event occurred
- Client - client IP address or hostname
- Username - the user initiating the action
- Session ID – the Essbase session ID
- Event Type - the event type
- Artifact Type - the type of artifact involved in the event
  Artifact type examples:
  - Artifact Type partition_file for Event Type ARTIFACT_UPLOADED
  - Artifact Type Application for Event Type LCM_EXPORT_START
  - Artifact Type User for Event Type APPLICATION_ROLE_ASSIGN
- Artifact Name – the name of the artifact involved in the event. For example, a file name, username, or application name
- Additional Info - additional information associated with the event
- Description - description of the event
  The Description field content is localized.

Copyright © 2019, 2026, Oracle and/or its affiliates.

- ID - A 128 bit universally unique identifier describing the event.
  Example: 123e4567-e89b-12d3-a456-426614174000
- Duration - duration of the event in miliseconds
- Application Name - name of the application
- Database Name - name of the database
- Status - success or failure

Copyright © 2019, 2026, Oracle and/or its affiliates.

14

Link Cubes Using Partitions or @XREF/@XWRITE

If you have more than one Essbase cube involved in data analysis, you can share data across the cubes. To do so, you can connect them by implementing partitions, @XREF/@XWRITE, or both.

Two cubes connected by a partition can be thought of as a source and target pair. When using @XREF/@XWRITE, it is easiest to think of the local cube and the remote cube.

When partitioning between cubes on the same Essbase instance, no reference to the host instance or login credentials are required. However, if the cubes you wish to connect are on separate Essbase instances, you will first need to create a reusable connection to link the two instances.

To use partitions, users must be provisioned on the remote cube as well as the local cube.

The source cube and target cube of a partition must be on the same Essbase version.

If you set up a NAT gateway, when using public and private subnets, the NAT gateway needs to be added to ingress rules in load balancer security rules for partitions to work.

- Define a Reusable Connection for Partitions or Location Aliases
- Understand Transparent and Replicated Partitions
- Create a Transparent Partition
- Create a Replicated Partition
- Refresh a Replicated Partition
- Understand @XREF/@XWRITE
- Create a Location Alias

Define a Reusable Connection for Partitions or Location Aliases

In the Essbase web interface, you can create a reusable connection between two Essbase instances, and use it to create partitions or location aliases, expanding user access to more than one Essbase database.

Create connections globally for use with all applications on the system, or at the application level for use within the context of an application. Global connections require System Administrator role, whereas application connections require, at minimum, User role with provisioned with Application Manager permission.

1. In the Essbase web interface, click **Sources**, and select **Create Connection > Essbase** to create a global Essbase connection. Alternatively, to create the connection at the application level,
   - In the Redwood Interface, go to the application, click **Sources**, and then select **Create Connection**.

Copyright © 2019, 2026, Oracle and/or its affiliates.

- In the Classic Web Interface, use the **Actions** menu on the target or local application and select **Inspect**, followed by **Sources**, **Create Connection**, and **Essbase**.

2. In the **Name** field, enter a name for the saved connection; for example `myhost01_conn`.
3. Select the **Use URL** checkbox, and enter the discovery URL of the remote Essbase instance. The discovery URL is available from your system administrator, and ends in `/agent`.
4. Enter a user name, password, and a description. The user defined in the connection must be provisioned for the source application you intend to access on the remote instance. If you have used a global connection, the user will need to be a system administrator or be provisioned for all applications you intend to access using the connection.
5. Click **Test** to verify that the connection is valid.
6. If it is valid, click **Create** to save the connection.

Now you have a remote Essbase connection defined. You can use this connection to define partitions and/or a location alias between the two instances.

Understand Transparent and Replicated Partitions

A partition is a region of a cube that is shared with another cube. You can create a transparent or replicated partition between a target and a source cube, to share congruent cube regions between them. In the Essbase web interface, you create partition definitions in the target cube.

A **transparent** partition target region is virtual; it pulls data on-demand from a source cube region containing stored data. The source cube can be in the same or another application, or on another Essbase instance.

A **replicated** partition target region is a physical copy of stored data from the source cube region. Data stored in a replicated partition target must be synchronized when data changes in the source cube. Using the replicated partition, some users access the data in the target, while others access it in the source.

Changes made to the data in a replicated partition flow from the source to the target. If users are permitted to change the data in the target partition region, it is overwritten when the replicated partition is refreshed.

The user creating the partition must be provisioned on the target application and also the source application. Business users querying the target cube must also be provisioned on both cubes, typically with Read access.

Create a Transparent Partition

Create a transparent partition if you need to provide access to data from a source Essbase cube as if it were in the current (target) cube. The source cube can be on the same Essbase instance or a remote instance.

1. On the Home page, open the target application and then open the target database (cube).
2. Click **Partitions**.
3. Click **Create** >**Transparent**.
4. On the **Connection** tab, in the **Server** field, if the source cube is on a different Essbase instance, enter the URL for that Essbase instance.

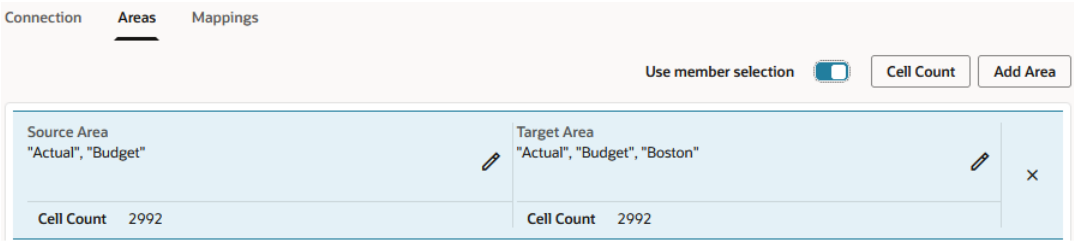
Copyright © 2019, 2026, Oracle and/or its affiliates.

If the URL doesn't work, you can use the Essbase host and port. For example,

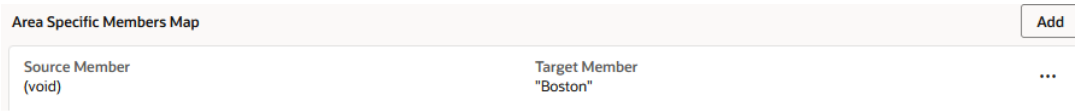
```
myserver.example.com:1423
```

> **Note**
>
> Do not enter the remote instance details in the target server name.

5. Enter your **User name** and **Password**, then click ✓ to **Validate User**.
6. Click ↻ to **Refresh Applications**, then select the source application and database (cube), and enter an optional description.
7. In the **Target Information**, enter your **User name** and **Password**, then click ✓ to **Validate User**.
8. You need to define at least one area. Go to the **Areas** tab.
9. (Optional) Select **Use member selection** using the toggle switch to select members from the outline. If you choose not to use member selection, type directly in the work area.
10. (Optional) Click **Open Source Area** to open a larger work area.
11. If you have enabled member selection in step 9, click **Edit source area** and provide at least one source area definition. If you have not enabled member selection, type directly in the source work area. For example, add a source area of some *`valid upper-level member specifications`*, and add the same matching target area. If the same member doesn't exist in both cubes, create an area mapping as described below.



12. Click **Cell Count** to identify how many cells are in the defined partition area and to ensure that the counts are matching.
13. Optionally, you can map member names between the target and source cubes within a specific area, using the **Areas** tab, or for multiple areas, using the **Mappings** tab. See Mapping Members in Partitions.



14. Click **Validate**.
15. If the validation succeeded, click **Create**.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Create a Replicated Partition

Create a replicated partition if you need to provide access to data from a source Essbase cube by duplicating some of it into the current (target) cube. The source cube can be on the same Essbase instance, or a remote instance.

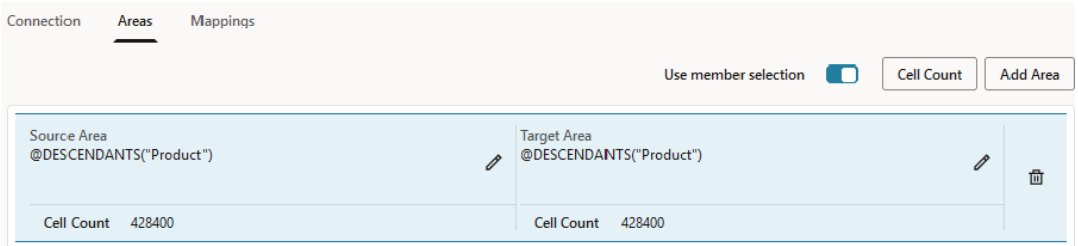
1. On the Home page, open the target application and then open the target database (cube).
2. Click **Partitions**.
3. Click **Create** >**Replicated**.
4. On the **Connection** tab, in the **Server** field, if the source cube is on a different Essbase instance, enter the URL for that Essbase instance.

   If the URL doesn't work, you can use the Essbase host and port. For example,

   ```
   myserver.example.com:1423
   ```

   > **Note**
   >
   > Do not enter the remote instance details in the target server name.

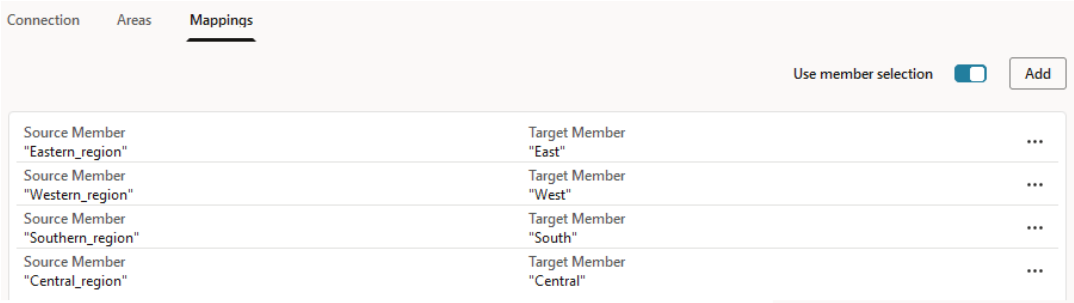
5. Enter your **User name** and **Password**, then click ✓ to **Validate User**.
6. Click ↻ to **Refresh Applications**, then select source application and database (cube), and enter an optional description.
7. In the **Target Information**, enter your **User name** and **Password**, then click ✓ to **Validate User**.
8. You need to define at least one area. Go to the **Areas** tab.
9. (Optional) Select **Use member selection** using the toggle switch to select members from the outline. If you choose not to use member selection, type directly in the work area.
10. (Optional) Click **Open Source Area** to open a larger work area.
11. If you have enabled member selection in step 9, click **Edit source area** and provide at least one source area definition. For example, add a source area of some `@DESCENDANTS(valid upper-level member specification)`, and add the same matching target area. If the same member doesn’t exist in both cubes, create an area mapping as described below.



12. Click **Cell count** to identify how many cells are in the defined partition area and to ensure that the counts are matching.

Copyright © 2019, 2026, Oracle and/or its affiliates.

13. Optionally, you can map member names between the target and source cubes within a specific area, using the **Areas** tab, or for multiple areas, using the **Mappings** tab. See Mapping Members in Partitions.



14. Click **Validate**.
15. If the validation succeeded, click **Create**.

Refresh a Replicated Partition

If you have at least Database Manager permission on an Essbase replicated partition target application, you can replicate the data from the source.

To refresh a replicated partition using the Essbase web interface

1. Navigate to the **Partitions** page:

   In the Redwood Interface,

   a. On the Home page, open the target application and then open the target database (cube).
   b. Click **Partitions**.

   In the Classic Web Interface,

   a. On the **Applications** page, expand the target application.
   b. In the row for the target cube, click the **Actions** menu, and click **Inspect**.
   c. Select the **Partitions** tab.
2. From the **Actions** menu on the replicated partition, select **Replicate Data from Source**.
3. Select **Update change cells only** to update the target only with source data that has been updated since the last update, or select **Update all cells** to update the target with all source data.

Export and Import Transparent and Replicated Partitions

You can export transparent and replicated partitions in JSON format and import them to re-create the partitions.

To export a partition,

1. On the Home page, open the target application and open the database (cube).
2. Click **Partitions**.
3. Click the Actions menu ••• to the right of the partition and select **Export**.

Copyright © 2019, 2026, Oracle and/or its affiliates.

4. Save the export file to a local folder.

To import a partition,

1. On the Home page, open the target application and open the database (cube).
2. Click **Partitions**.
3. Click **Import** and browse to the location where you saved the export file.
4. Open the file.
5. Enter the source password, then click ✓ to **Validate User**.
6. Click ↻ to **Refresh Applications**, then select the source application and cube.
7. Enter the target password, then click ✓ to **Validate User**.
8. Validate the partition and click **Create**.

Understand @XREF/@XWRITE

@XREF is an Essbase calculation function for referencing data in another cube. @XWRITE is a calculation function for writing back data to another cube. The cube containing the @XREF or @XWRITE formula is called the local cube. The second cube is called the remote cube.

To implement @XREF, you define a formula in the local cube that pulls values from a remote cube. The member containing the @XREF formula can either be stored or dynamically calculated.

To implement @XWRITE, you define a formula in the local cube that pushes (writes) values into a remote cube. The remote cube data intersection must be stored, since @XWRITE writes values into the remote cube.

If the local and remote cubes are on different Essbase instances, a location alias containing connection information must be defined.

To implement @XREF or @XWRITE for cubes on the same instance, two options are available:

1. Location alias
2. Application name and database name combination

Function **syntax 1** calls for a location alias:

```
@XREF (locationAlias [, mbrList])
@XWRITE (expression, locationAlias [, mbrList])
```

Function **syntax 2** calls for using an application name and database name combination:

```
@XREF(appName, dbName [, mbrList])
@XWRITE (expression, appName, dbName [, mbrList])
```

When using application name and database name combination, users of the local cube must also be provisioned on the remote cube.

Additional References:

- @XREF

Copyright © 2019, 2026, Oracle and/or its affiliates.

- @XWRITE
- Create a Location Alias

Create a Location Alias

Create a location alias if you need addressability to another Essbase cube for your block storage (BSO) calculations or formulas. Location alises help you call @XREF or @XWRITE functions to reference data from another cube, whether that cube is on the same Essbase instance or a different one.

You do not need to provide a user name and password when you create a location alias using the Essbase web interface. However, if the remote cube is not on the same Essbase instance, a saved connection is required (refer to Define a Reusable Connection for Partitions or Location Aliases if you need to create one).

1. Navigate to the **Location Aliases** page.
   - In the Redwood Interface, on the Home page, open the application and then open the database (cube).
   - In the Classic Web Interface, on the Applications page, expand the target application. In the row for the local cube, click the **Actions** menu, and click **Inspect**.
2. Click **Location Aliases**.
3. Click **Add Location Alias**.
4. In the **Location alias name** field, enter a name.
5. In the **Select an Essbase connection** field, choose a saved connection to the Essbase instance that hosts the remote cube, or choose **Within same instance** if the cube is in the same Essbase instance.

> **Note**
>
> The username appears when you create a location alias using an Essbase connection other than **Within same instance**.

To use the location alias for read operations from a remote cube to the target, use the @XREF function in a member formula or calculation script on the local cube. To use it to write from the local to the remote cube, use @XWRITE on the local cube.

Copyright © 2019, 2026, Oracle and/or its affiliates.

15

Federated Cubes: Integrate Essbase with Autonomous AI Database

Federated cubes enable Essbase to directly query your data stored in Autonomous AI Lakehouse, combining Essbase's analytical power with the benefits of Autonomous AI Database.

Benefits of Federated Cubes

With a federated cube, you can bypass the process of loading data into an Essbase cube before performing aggregations and queries. Data storage and processing occurs within Autonomous AI Lakehouse, to take advantage of the benefits of Autonomous AI Database and also of Essbase's analytical features.

Bypassing regular data loads from relational data sources to Essbase can save you operational costs surrounding the extract, transform, load (ETL) pipeline (using rule files or other data load processes), and eliminates the need for outline restructuring.

With Autonomous AI Database, the database configuration, tuning, object storage, backups, and updates are all Oracle managed, so you can use Essbase in a federated cloud environment without spending time on infrastructure management.

Writeback is supported through Essbase to stored intersections. For example, the data values you submit using Smart View (or MDX Insert) are updated in the fact table on Autonomous AI Lakehouse.

You can also perform Essbase calculations and data loads, and Essbase will write SQL to update the fact table in Autonomous AI Lakehouse.

Data Storage Management in Federated Cubes

The data for your federated cube is stored in Autonomous AI Lakehouse. Either you can manage the data, or you can let Essbase manage it for you.

If you already use Essbase and are interested in trying a federated cube, consider making your existing Essbase database into a federated cube. In this situation, create an Essbase-managed federated partition.

If you're new to Essbase, and/or you already have an established workflow for updating your data in Autonomous AI Lakehouse, then a good solution is to manage your own federated cube data.

1. Decide which data storage option is best for your organization:
   - Essbase managed – Essbase creates a fact table while creating the federated partition, and manages it afterwards.
   - User managed – You set up your own fact table in Autonomous AI Lakehouse, using SQL or an Essbase data export.
2. If you want to let Essbase manage the Autonomous AI Lakehouse storage, enable the Essbase managed option by setting USERMANAGEDFEDFACT FALSE.

Copyright © 2019, 2026, Oracle and/or its affiliates.

> **Note**
>
> Set USERMANAGEDFACT FALSE before creating your federated partition. The option cannot be changed afterwards.

With Essbase managed data storage, you do not have to set up a fact table, because Essbase creates it for you. Additionally, Essbase manages the fact table. You do not have to delete and recreate the partition when you make outline changes.

More Topics:

- Prerequisites for Federated Cubes
- Federated Cube Deployment Workflow
- Provision Autonomous AI Lakehouse for Federated Cubes
- Deploy Essbase from Marketplace for Federated Cubes
- Create a Schema for Federated Cubes
- Identify the Pivot Dimension and Set up the Fact Table
- Create a Connection for Federated Cubes
- Set up Credentials for Federated Cubes
- Create a Federated Partition
- Data Load Options for Federated Cubes
- Calculate and Query Federated Cubes
- Federated Cube Maintenance and Troubleshooting
- Remove a Federated Partition
- Restrictions for Federated Cubes

Prerequisites for Federated Cubes

Once you have decided you want to try a federated cube with Essbase, you need make a decision about storage management. Then, you need to provision an Autonomous AI Lakehouse, deploy Essbase, and complete required setup tasks.

> **Note**
>
> After you create a federated partition, your Essbase cube becomes a federated cube.

Before you can create a federated partition,

1. Based on your use case, decide on a data storage management option. See Data Load Options for Federated Cubes.
2. Provision an Oracle Autonomous AI Database Serverless instance with the Autonomous AI Lakehouse workload type.
3. Deploy Essbase to the same Oracle Cloud Infrastructure tenancy using Marketplace.
4. Perform the setup tasks listed below.

Copyright © 2019, 2026, Oracle and/or its affiliates.

The following setup tasks must be completed before you can create a federated partition in Essbase.

Review the following checklists, and then proceed to Federated Cube Deployment Workflow to learn the order of tasks for implementation.

Table 15-1 Cloud Deployment Prerequisites

Requirement	Reason	What to Do / More Information
Essbase and Autonomous AI Lakehouse are deployed together in a shared Oracle Cloud Infrastructure tenancy, using the Marketplace listing.	Oracle Cloud Infrastructure enables Essbase to take advantage of flexible and scalable cloud computing architectures. Autonomous AI Lakehouse Serverless stores the data for your Essbase cube.	Marketplace Deploy Essbase from Marketplace for Federated Cubes
Essbase uses the Autonomous AI Lakehouse as its schema repository.	The following schemas in Autonomous AI Lakehouse have different purposes for Essbase: The **Repository Creation Utility (RCU)** schemas are created automatically during Essbase deployment, and hold information about platform artifacts and components. The **Database user schema** is home to the fact table that holds Essbase data. **Caution** These are separate schemas by design. Do not use any of the RCU schemas for the fact table.	Deploy Essbase from Marketplace for Federated Cubes
The Essbase deployment is configured to use OCI object storage.	The Essbase file catalog storage must be integrated with the OCI **Object Storage Bucket**.	Deploy Essbase from Marketplace for Federated Cubes

Copyright © 2019, 2026, Oracle and/or its affiliates.

Table 15-2 Database Prerequisites

Requirement	Reason	What to Do / More Information
Your organization deploys an Autonomous AI Lakehouse Serverless.	Configuration, tuning, storage, backups, and updates are all Oracle managed, so you can use Essbase in a cloud environment without spending time on infrastructure. Autonomous AI Lakehouse also handles the data storage for Essbase. Whether you require fastest query performance, highly concurrent workloads, or a mixture of both, Autonomous AI Lakehouse provides the right service you need to meet those data access requirements.	Provision Autonomous AI Lakehouse for Federated Cubes
The Database Administrator for Autonomous AI Lakehouse creates a new schema.	You need a schema in Autonomous AI Lakehouse that will contain the data for your federated cube. **Note** Oracle recommends using a dedicated schema. A new Autonomous AI Lakehouse user is equivalent to a new schema. In the remainder of this documentation, we will refer to the owner of the schema as **DB User**.	Create Users on Autonomous Database (if you want to use the OCI Console) or CREATE USER (to create the Autonomous AI Lakehouse user/schema using any SQL client tool)
The Database Administrator for Autonomous AI Lakehouse grants resource privileges to the **DB User**.	The Database user in Autonomous AI Lakehouse needs to be able to: • create a connection to Autonomous AI Lakehouse • create a fact table to store Essbase data	Manage User Roles and Privileges on Autonomous AI Database Provision Autonomous AI Lakehouse for Federated Cubes

Copyright © 2019, 2026, Oracle and/or its affiliates.

Table 15-2 (Cont.) Database Prerequisites

Requirement	Reason	What to Do / More Information
Optional: The **DB User** creates a fact table in the schema. **Note** This prerequisite does not apply if you plan to let Essbase manage the data storage. See Data Load Options for Federated Cubes.	A fact table in Autonomous AI Lakehouse is needed to store the Essbase cube data. **Note** You can load data into Autonomous AI Database using Data Studio tools. See The Data Load Page.	Identify the Pivot Dimension and Set up the Fact Table

Copyright © 2019, 2026, Oracle and/or its affiliates.

Table 15-2 (Cont.) Database Prerequisites

Requirement	Reason	What to Do / More Information
SQL Session Governor is enabled.	You can cancel SQL sessions that are running on Autonomous AI Lakehouse. SQL cancellation may be necessary if an Essbase calculation is taking too long, or if the federated cube needs backup or recovery. If Essbase calculations, data loads, or aggregations running on federated cubes generate long running SQL statements, it is possible to cancel the sessions, even though they are running on Autonomous AI Lakehouse.	Cancel Long Running SQL on Federated Cubes **Note** Steps 1 and 2 are required. Step 2 is a validation of step 1. If it fails, repeat step 1, correcting any errors.

Table 15-3 Essbase Platform Prerequisites

Requirement	Reason	What to Do / More Information
An Essbase application and cube are created. The cube does not need to have any data in it. The cube must be within its own uniquely-named application. Federated cubes should not share an application with other cubes. Do not use the same Autonomous AI Lakehouse schema for multiple instances of Essbase.	An Essbase outline is required, to map the cube to the fact table in Autonomous AI Lakehouse.	Create a Cube from an Application Workbook
The Essbase service administrator or application manager defines a connection.	Essbase must have connectivity with Autonomous AI Lakehouse.	Create a Connection for Federated Cubes

Copyright © 2019, 2026, Oracle and/or its affiliates.

Table 15-3 (Cont.) Essbase Platform Prerequisites

Requirement	Reason	What to Do / More Information
This is a prerequisite for creating a federated partition. One or more individuals configures DBMS_CLOUD credentials.	Before creating a federated partition, you'll need to enable cloud credentials, so that Essbase can store data and metadata in Autonomous AI Lakehouse.	Set up Credentials for Federated Cubes

Federated Cube Deployment Workflow

In order to implement a federated cube, Essbase and Autonomous AI Database Serverless need to be deployed together in a shared Oracle Cloud Infrastructure tenancy, with Autonomous AI Lakehouse serving as the repository database that holds RCU schemas for the Essbase stack deployed on OCI from Marketplace.

The workflow to set up a federated cube is:

1. Review Plan a Federated Partition Environment.
2. Review the Prerequisites for Federated Cubes.
3. Log in to your organization's Oracle Cloud Infrastructure tenancy.
4. Optional: Provision an instance of Autonomous AI Lakehouse. (You may instead choose to provision one during the next step).

   > **Note**
   >
   > Here and elsewhere in this federated partitions documentation, all references to Autonomous AI Lakehouse should be considered to mean Autonomous AI Database Serverless with Autonomous AI Lakehouse workload type.

   See Provision Autonomous AI Lakehouse for Federated Cubes.
5. From the Marketplace listing in Oracle Cloud Infrastructure, deploy an Essbase stack to the same tenancy. See Deploy Essbase from Marketplace for Federated Cubes.
6. Create a schema in Autonomous AI Lakehouse to use for the fact table. See Create a Schema for Federated Cubes.
7. Create an Essbase application and cube.

   Select a BSO or ASO application and cube as the starting point. Depending on your circumstances, the starter cube may already exist, or you may create a new one and then create the federated partition over it.

   > **Note**
   >
   > After you create the federated partition, your cube becomes a federated cube.

   If you are not sure which type of cube to start with, review the comparison table in Federated Cubes: Integrate Essbase with Autonomous AI Database.

Copyright © 2019, 2026, Oracle and/or its affiliates.

8. Create a fact table in the Autonomous AI Lakehouse schema.

> **Note**
>
> This step is not required if you plan to let Essbase manage the data storage. See Data Load Options for Federated Cubes.

For guidelines on fact tables (and pivot dimensions), see Learn About Fact Tables and Pivot Dimensions.

9. Define a connection that enables Essbase to access the schema on Autonomous AI Lakehouse, as shown in Create a Connection for Federated Cubes.

   To create a global connection, you need to have the service administrator role. To create an application level connection, you need to have user role, plus application manager permission on the application.

10. Set up cloud credentials to enable Essbase to push data and metadata to Autonomous AI Lakehouse. See Set up Credentials for Federated Cubes.
11. Log in to the Essbase web interface and create the federated partition, as described in Create a Federated Partition.
12. Complete a workflow to enable SQL cancellation of long running requests. See Cancel Long Running SQL on Federated Cubes.
13. Learn about maintaining and troubleshooting the federated cube. See Federated Cube Maintenance and Troubleshooting.

Provision Autonomous AI Lakehouse for Federated Cubes

To use a federated cube with Essbase, you must provision an instance of Autonomous AI Database Serverless, and create a dedicated schema. You can provision the database either before you deploy the Essbase stack on Oracle Cloud Infrastructure using the Marketplace listing, or during the deployment.

The following workflow describes how to create the Autonomous AI Database required for a federated cube.

> **Tip**
>
> Skip these steps if you want to create and provision Autonomous AI Database during Essbase 21c deployment. Refer instead to Deploy Essbase from Marketplace for Federated Cubes.

1. Log in to your organization's Oracle Cloud Infrastructure tenancy.
2. In Oracle Cloud Infrastructure Console, click **Oracle AI Database**.
3. Click **Autonomous AI Database**.
4. Under **Applied filters**, ensure the correct target compartment is applied.
5. Click **Create Autonomous AI Database**.
6. Optionally change the **Display name** and the **Database name** to something more user-friendly than the default names.
7. Double check that the correct target compartment is selected.

Copyright © 2019, 2026, Oracle and/or its affiliates.

8. For the workload type, keep the default selection of **Lakehouse**.
9. In the configuration area,
   a. Select a database version.
   b. Select an ECPU count (2 recommended).
   c. Select an amount of storage to allocate (1 TB recommended).
   d. Select auto scaling requirements.
10. In the **Administrator credentials creation** area, define the password for the Autonomous AI Database administrator.
11. In the **Network access** area,
    a. Select one of the access types:
       - Secure access from everywhere
       - Secure access from allowed IPs and VCNs only
       - Private endpoint access only
    b. Leave the option checked to require mutual TLS authentication.
12. Click **Create**.
13. Allow a few minutes for OCI to provision the Autonomous AI Lakehouse.
14. Use a vault in the **Identity & Security** section of OCI Console to encrypt and save the Autonomous AI Database administrator password. Refer to Create Vault, Secrets, and Encrypt Values.

Deploy Essbase from Marketplace for Federated Cubes

To implement a federated cube, you must deploy Essbase according to specific requirements.

Prerequisites

These instructions assume you have already done the following, as a domain administrator on Oracle Cloud Infrastructure:

- Created a compartment, dynamic group, and policy on the Oracle Cloud Infrastructure tenancy, as described in Before You Begin with Oracle Essbase.
- Created a vault for encryption keys and secrets on the Oracle Cloud Infrastructure tenancy, as described in Create Vault, Secrets, and Encrypt Values.
- Created a confidential application and an initial Essbase system administrator on the Oracle Cloud Infrastructure tenancy, as described in Set Up Essbase Access in Identity Cloud Service. Be sure to save the application's client secret, as well as the Essbase administrator password, in the vault.
- Provisioned an Autonomous AI Lakehouse Serverless as described in Provision Autonomous AI Lakehouse for Federated Cubes. Be sure to save the Database administrator's password in the vault.

> **Note**
>
> You can provision Autonomous AI Lakehouse during the Essbase deployment.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Instructions

From the Marketplace on Oracle Cloud Infrastructure, deploy an Essbase stack to the same tenancy where you have provisioned (or will provision) Autonomous AI Lakehouse.

Follow the instructions in Create Stack, but while completing those steps,

1. If you already provisioned Autonomous AI Lakehouse Serverless previously, ensure that it is up and running first.
2. In the Essbase Instance screen, change the Catalog Storage Type to **Object Storage Bucket**. This step is mandatory for federated cubes.

> **Note**
>
> - You cannot change Catalog Storage Type back to Local File system after deployment.
> - Any Essbase jobs that require read/write access to files in the Essbase catalog will search for them in (or export them to) the OCI object storage bucket associated with the Essbase stack on OCI.
> - For more information, see Specify Files in a Catalog Path.

3. In the Database Configuration screen, select how you want to configure Autonomous AI Lakehouse Serverless with Essbase. Essbase will use this database as the repository for its RCU schemas.
   a. If you want to provision Autonomous AI Lakehouse Serverless during this Essbase deployment, click **Show Advanced Database Options**, and from **Choose a database workload type**, select Autonomous AI Lakehouse.
   b. Or, you can select **Use Existing Database** to use a previously provisioned Autonomous AI Lakehouse Serverless.
4. Use a vault in the **Identity & Security** section of OCI Console to encrypt and save the Autonomous AI Database administrator password. Refer to Create Vault, Secrets, and Encrypt Values.
5. Complete the rest of the instructions in Deploy Essbase.

After Essbase Deployment - What's Next?

In the same Autonomous AI Lakehouse that is the repository database for Essbase, you will need to do the following:

- Create a Schema for Federated Cubes. No other database instance or type can be used for this schema.
- Identify the Pivot Dimension and Set up the Fact Table. No other database instance or type can be used for the fact table.
- An administrator with SSH access to the Essbase instance on OCI must set up credentials by running the `./configure-dbms-writeback.sh` script. You only need to run the script once per database schema in Autonomous AI Lakehouse. Refer to Set up Credentials for Federated Cubes.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Create a Schema for Federated Cubes

After you have provisioned an instance of Autonomous AI Lakehouse Serverless, create a schema in the database for the fact table you need to work with Essbase federated cubes.

> **Note**
>
> Oracle recommends using a dedicated schema for federated cubes.

The schema you need to create for a federated cube, including its fact table, is independent from the Essbase RCU schemas. However, it does need to be in the same Autonomous AI Lakehouse Serverless database, which is also known as the repository database.

> **Note**
>
> If you are letting Essbase manage the fact table, Essbase creates the fact table for you.

1. Log in to Autonomous AI Lakehouse as the Autonomous AI Database administrator.
2. Create a schema / Database user (for example, `ADB_USER`) with sufficient privileges to work with federated cubes.

   ```
   CREATE USER ADB_USER identified by schemapass DEFAULT TABLESPACE DATA
   TEMPORARY TABLESPACE TEMP ACCOUNT UNLOCK;
   grant CREATE ANALYTIC VIEW, CREATE HIERARCHY, CREATE TABLE, CREATE
   ATTRIBUTE DIMENSION, CREATE SESSION, CREATE VIEW, RESOURCE, CONNECT to
   ADB_USER;
   grant execute on dbms_cloud to ADB_USER;
   grant execute on dbms_cloud_oci_obs_object_storage to ADB_USER;
   ALTER USER ADB_USER DEFAULT ROLE RESOURCE;
   ALTER USER ADB_USER QUOTA UNLIMITED ON DATA;
   commit;
   ```

3. Optional: If you plan to create more than one federated cube, you must make a choice: You can use a single schema for all federated cubes, or create multiple schemas (typically one schema per application containing a federated cube).

   > **Note**
   >
   > Regardless of the number of schemas you create, you must maintain the following:
   >
   > - Each Essbase cube has only one federated partition
   > - Each federated cube uses only one fact table.
   > - Unlike other partition types used in Essbase, the data is not in two locations. Your federated cube's fact table must contain *all* the cube's data.

Copyright © 2019, 2026, Oracle and/or its affiliates.

For a comprehensive list, refer to Restrictions for Federated Cubes.

4. Now that you have created a schema for the fact table, you can move on to Identify the Pivot Dimension and Set up the Fact Table.

> **Note**
>
> You will also need to set up cloud credentials to enable Essbase to push data and metadata to Autonomous AI Lakehouse. Refer to Set up Credentials for Federated Cubes.

Identify the Pivot Dimension and Set up the Fact Table

A fact table in Autonomous AI Lakehouse stores the data of the federated cube. If you don't have a fact table that meets the requirements for federated partitions, and you are not letting Essbase manage the fact table, you must create one.

> **Note**
>
> If you are letting Essbase manage the fact table, Essbase creates the fact table for you.

Before starting this section, create an Essbase application and cube, if you don't already have one.

- Identify the Pivot Dimension
- Create the Fact Table

Identify the Pivot Dimension

As part of designing a federated cube, you need to select the *pivot dimension*. A pivot dimension is a dimension you designate from the Essbase cube outline to represent numeric data values.

- The pivot dimension does not have to be measures/accounts, but it may be.
- All stored members of the pivot dimension must map to the fact table columns that represent your numeric data values in Autonomous AI Lakehouse.
- If you need to run Essbase block storage (BSO) calculation scripts, select a dense dimension as the pivot dimension. Calculation scripts are not supported for federated cubes if the pivot dimension is sparse.
- The pivot dimension should have fairly static member names, and not a very large number of members. Reason: Changing the pivot dimension in the Essbase cube outline (for example, by adding or renaming stored members) necessitates corresponding, manual updates to the fact table in Autonomous AI Lakehouse, and also requires recreation of the federated partition.
- Essbase dimensions that include members requiring complex, dynamic formulas (such as "Opening Inventory" and "Ending Inventory," using Sample Basic as an example) should not be selected as the pivot dimension.

Copyright © 2019, 2026, Oracle and/or its affiliates.

- You provide your selected pivot dimension at the time of creating a federated partition.
- Oracle Database has a limit of 1,000 columns, and the pivot dimension inherits this limit. Determine the number of eligible column members in the pivot dimension to ensure that you do not encounter the limit. The number of potential stored member combinations in the pivot dimension plus the number of dimensions in the cube should be less than or equal to 1,000.
- For federated cubes based on ASO, Oracle recommends that dimensions containing multi-level stored member hierarchies not be selected as the pivot dimension. Select a pivot dimension with dynamic hierarchies, or a stored hierarchy that is a flat, single-level hierarchy (where all members are level 0 stored members).

Create the Fact Table

A fact table in Autonomous AI Lakehouse stores all the data values of a federated cube. If you don't already have the required fact table, and your federated cube is not Essbase- managed, you must create one.

> **Note**
>
> This topic is not required if you are letting Essbase manage the fact table, because Essbase creates the fact table for you. See Data Load Options for Federated Cubes.

Before you start, ensure you have a schema for the fact table. See Create a Schema for Federated Cubes.

The fact table must be in Essbase-ready format, meaning that it meets the following requirements for its content and shape:

- Each of the cube's (non-attribute) dimensions must represented as a single column header, with the exception that one of the cube's dimensions (typically the one containing measures/accounts) must be pivoted into two or more columns. This exception is called the *pivot dimension*.
- The fact table must be comprised of unique records (no duplicates), with one row per sequence of Essbase cell intersections.

If you are familiar with Essbase data exports, you will notice that the shape of the fact table is exactly like an Essbase column export.

Similarly to a column export, the fact table must include:

- one column for each (non-attribute) dimension of the outline (except for the pivot dimension)
- one column for each stored member of the pivot dimension

The following is an example of a fact table in which the measures dimension has been pivoted, which means it is the pivot dimension. The pivot dimension affects the shape of the fact table, as that dimension's stored members become column headers: SALES, COGS, MARKETING, PAYROLL, MISC, INTITIAL_INVENTORY, and ADDITIONS.

Copyright © 2019, 2026, Oracle and/or its affiliates.

	DIMENSION_PRODUCT	DIMENSION_MARKET	DIMENSION_YEAR	DIMENSION_SCENARIO	SALES	COGS	MARKETING	PAYROLL	MISC	INITIAL_INVENTORY	ADDITIONS
1	100-10	Oklahoma	Jul	Budget	110	50	10	10	(null)	(null)	100
2	100-10	Missouri	Jun	Actual	169	76	28	33	1	(null)	202
3	100-10	Missouri	Jun	Budget	170	80	20	30	(null)	(null)	190
4	100-10	Missouri	Jul	Actual	169	76	28	33	1	(null)	162
5	100-10	Missouri	Jul	Budget	170	80	20	30	(null)	(null)	150
6	100-10	Missouri	Aug	Actual	160	72	27	33	1	(null)	153
7	100-10	Missouri	Aug	Budget	160	70	20	30	(null)	(null)	140
8	100-10	Missouri	Sep	Actual	150	67	25	33	0	(null)	144

You can either build the fact table using SQL, or you can create it from an Essbase data export. You can load data to the fact table using Autonomous AI Lakehouse tools, or using Essbase data load functionality.

Additional guidelines for building a fact table include:

- The fact table must have fewer than 1000 columns.
- Do not include columns that will map in Essbase to attribute dimensions.
- The fact table should not have lesser precision than IEEE binary64 (double).
- The fact table should have internationalized strings for dimension members, using NVARCHAR2 type, with 1024 bit character length.

Fact Table Creation Example

To create a fact table in Autonomous AI Lakehouse, you can use SQL.

1. Using SQL Developer or your tool of choice, log in to Autonomous AI Lakehouse as the schema owner (from step Create a Schema for Federated Cubes).
2. Use SQL to create the fact table, if you don't already have one.

   For example, the following SQL creates a fact table from data export from the Essbase cube, Sample Basic.

```
CREATE TABLE "SAMP_FACT"
( "PRODUCT" NVARCHAR2(1024),
"MARKET" NVARCHAR2(1024),
"YEAR" NVARCHAR2(1024),
"SCENARIO" NVARCHAR2(1024),
"SALES" NUMBER(38,0),
"COGS" NUMBER(38,0),
"MARKETING" NUMBER(38,0),
"PAYROLL" NUMBER(38,0),
"MISC" NUMBER(38,0),
"INITIAL_INVENTORY" NUMBER(38,0),
"ADDITIONS" NUMBER(38,0)
) NOCOMPRESS LOGGING PARALLEL 4;
```

Notes

- In the example above, the fact table name is SAMP_FACT, and is based on Sample Basic.
- For best performance, all non-numeric columns in the fact table should be of type NVARCHAR2(1024), and all numeric columns should be of type NUMBER.
- Oracle recommends enabling parallel creation of the index in Autonomous AI Lakehouse, by adding PARALLEL 4.
- Metadata columns should not allow inclusion of NULL values.

Copyright © 2019, 2026, Oracle and/or its affiliates.

- Oracle recommends NOCOMPRESS when the usage of the cube includes data generative processes such as incremental data load or batch script updates. If the cube is to be used mostly for read operations, use COMPRESS to optimize the fact table for reporting.
- If you get the following validation error when creating the fact table, delete null rows.

```
ORA-18265: fact table key column ("<DIM_NAME>") with value ('') not in
      dimension("<Name_of_Column") star table key column
```

- For best performance, refrain from adding any specific constraints on the table without a definitive need.
- In the example above, the fact table name is based on Sample Basic, which is available in the gallery in the Essbase file catalog. You can export the data from this sample cube or any other Essbase cube, and load it to build a fact table. Before you can do this, you will need to set up credentials for loading data to a federated cube. Refer to Set up Credentials for Federated Cubes to set up the credentials and learn about how to export data to DBMS format using the DATAEXPORT command.

Create a Connection for Federated Cubes

Define a connection between Essbase and Autonomous AI Lakehouse Serverless, to work with federated cubes.

Federated partitions are supported only for Essbase deployment on Oracle Cloud Infrastructure.

Before you start defining the required connection, review the Federated Cube Deployment Workflow to ensure you have completed all required preliminary tasks.

Considerations for Implementing Federated Cube Connections

Review how you created the necessary schema(s) for federated cubes. If you have one Autonomous AI Lakehouse schema designated for multiple federated cubes, it makes sense to create one global connection that all the applications can share. If you have one or more schemas but only one Essbase application per schema, an application-level connection to each schema is a good approach.

- To create a global connection, you need to have the service administrator role.
- To create an application level connection, you need to have user role, and at least application manager permission on the application.

To create the required connection for federated cubes,

- Redwood
- Classic

Redwood

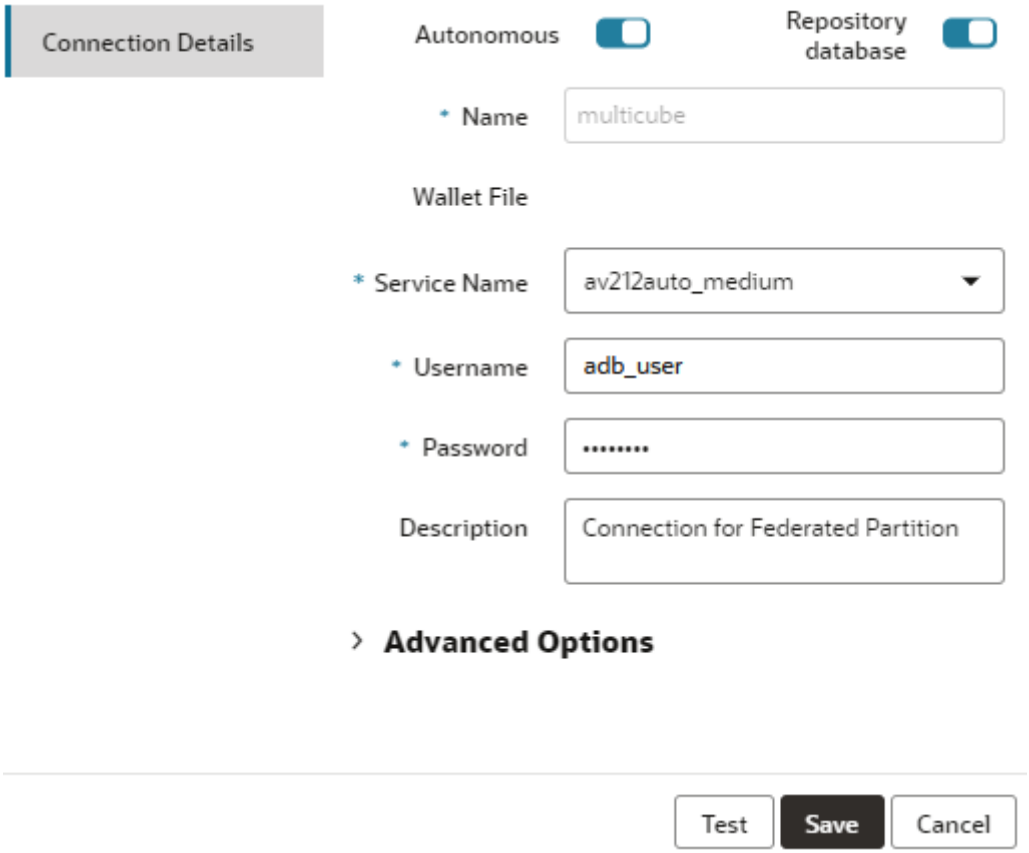
1. In the Essbase web interface, click **Sources**, then **Connections**.

   To define the connection and Datasource at application level instead of globally, start on the Applications page instead of the Sources page. Click an application name, and then click **Sources**.

2. Click **Create Connection**, and for the connection type, select **Oracle Database**.

Copyright © 2019, 2026, Oracle and/or its affiliates.

3. Enable the **Autonomous** switch.



4. Enter a connection name.

   If you are recreating an Autonomous AI Lakehouse connection after migrating the application using the Lcmimport CLI command (or an Import LCM job), it is recommended to use a new connection name, to avoid encountering errors.

5. Select a service name.

   Use connection type Medium when you need your application to run many Essbase calculations. Use connection type Low if your application is used mostly for queries. Consider using Medium connection type if your queries involve large calculations and you don't have a large number of concurrent users.

6. Enable the **Repository Database** switch. This is required for federated cube connections.

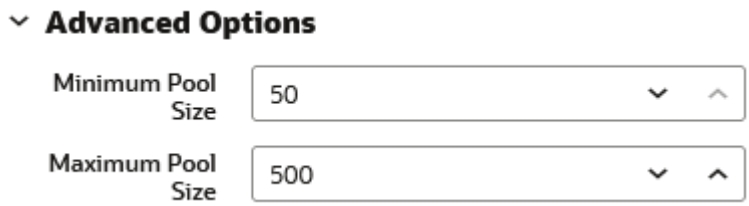
   You do not need to upload a wallet, because Essbase will use the wallet associated with the repository database.

7. Enter your Autonomous AI Lakehouse username, password, and optionally, a description.

8. Click **Test** to validate the connection, and if successful, click **Create**.

   If you get connection errors, you may need to expand **Advanced Options** to adjust the minimum and maximum connection pool sizes.

Copyright © 2019, 2026, Oracle and/or its affiliates.



Refer to About Controlling the Pool Size in UCP in *Universal Connection Pool Developer's Guide*.

9. Verify that the connection was created successfully and appears in the list of connections.

Classic

1. In the Essbase web interface, on the Sources page, click **Connections**.

   To define the connection and Datasource at application level, instead of globally, start on the Applications page instead of the Sources page. From the Actions menu to the right of an application name, launch the inspector and click **Sources**.

2. Click **Create Connection** and select **Oracle Database**.
3. Select **Autonomous** using the toggle switch.

4. Enter a connection name.

   If you are recreating an Autonomous AI Lakehouse connection after migrating the application using the lcmimport CLI command (or an Import LCM job), it is recommended to use a new connection name to avoid encountering errors.

5. Select a service name.
6. Select the **Repository Database** option. This option is required to be selected for federated partition connections.

   You do not need to upload a wallet, because Essbase will use the wallet associated with the repository database.

7. Enter your Autonomous AI Lakehouse username, password, and optionally, a description.
8. Click **Test** to validate the connection, and if successful, click **Create**.

   If you get connection errors, you may need to expand **Advanced Options** to adjust the minimum and maximum connection pool sizes.

   Refer to About Controlling the Pool Size in UCP in *Universal Connection Pool Developer's Guide*.

Copyright © 2019, 2026, Oracle and/or its affiliates.

9. Verify that the connection was created successfully and appears in the list of connections.

> **Note**
>
> Before you create the federated partition, one or more individuals must configure DBMS_CLOUD credentials. Refer to Set up Credentials for Federated Cubes for more information.

Set up Credentials for Federated Cubes

Before you can use federated partitions, you need to set up cloud credentials. To do this, you must integrate Essbase with OCI object storage, and configure DBMS_CLOUD credentials using the `configure-dbms-writeback` script.

This script creates the required Oracle Cloud credentials for the Autonomous AI Database schema that Essbase uses for your federated cube. The stored cloud credential is used by DBMS_CLOUD procedures to access OCI for the purpose of loading data.

Integrate Essbase with OCI Object Storage

To push data and metadata from a federated cube to your fact table in Autonomous AI Lakehouse, Essbase uses the DBMS_CLOUD package.

In order to access the DBMS_CLOUD package, you must have selected to integrate Essbase with OCI Object Storage when you deployed the Oracle Essbase stack from Oracle Cloud Marketplace.

For full details, see Deploy Essbase from Marketplace for Federated Cubes.

User Types in Credentials Workflow

Users with following types of access are or may be involved in the required workflow for setting up a federated cube. In your organization, these roles may be separate (so that the setup is a collaborative workflow), or the roles may combined (one person has all the required access).

User Type	Role within Workflow
SSH User	Can use the operating system command line to access, as **opc** user, the Essbase instance deployed on Oracle Cloud Infrastructure. (May be the same person who deployed Essbase as a stack on OCI).
DB User	Knows the Autonomous AI Lakehouse schema name and password -- the same schema and password that is used to create the Essbase connection to Oracle Database (a required prerequisite before creating the federated partition).
OCI User	Can access the OCI Console, including the object storage bucket for the Essbase catalog.
DB Admin	Knows the Oracle Database administrator schema name and password.

Copyright © 2019, 2026, Oracle and/or its affiliates.

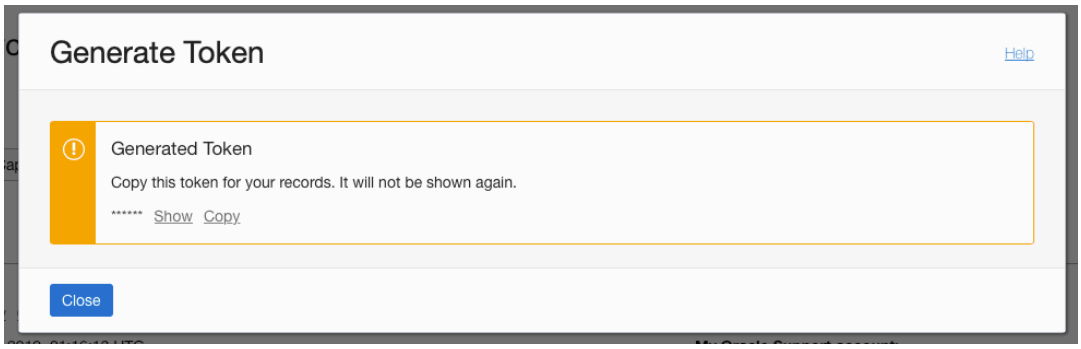
User Type	Role within Workflow
Essbase Admin	The Essbase system administrator. Can be the initial Essbase administrator created by the identity domain administrator (who may also be the **OCI User**), or it can be another Essbase system administrator created after completion of Essbase deployment.
Essbase Application Manager	The manager/owner of an Essbase application, created after completion of Essbase deployment.

Credentials Workflow

The following workflow of steps must be completed per each database schema that you are using for a federated cube.

1. **OCI User**: Follow instructions in Deploy Essbase from Marketplace for Federated Cubes to deploy Essbase to the OCI tenancy with the appropriate selections for federated cubes.
2. **DB User**, **Essbase Admin**, or **Essbase Application Manager**: Log in to Essbase web interface, and create a connection to Autonomous AI Lakehouse, as described in Create a Connection for Federated Cubes.
3. **OCI User**: From your user profile in OCI Console, generate and copy an authentication token. Provide this, and your user name, to the **SSH User**.

   See Getting an Auth Token.



4. **DB User**, **Essbase Admin**, or **Essbase Application Manager**: Create the federated partition, as described in Create a Federated Partition.
5. **SSH User**: Run the `./configure-dbms-writeback.sh` script, available on the Essbase instance on OCI. You only need to run the script once per database schema in Autonomous AI Lakehouse.

   Example:

   a. Switch to the **oracle** user.

   ```
   sudo su oracle
   ```

   b. Navigate to the script location.

   ```
   cd /u01/vmtools/config/adwwb_dbms
   ```


Copyright © 2019, 2026, Oracle and/or its affiliates.

c. Run the script.

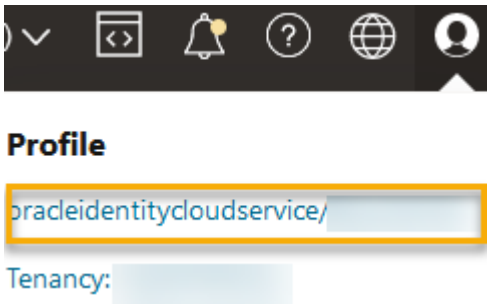
```
./configure-dbms-writeback.sh
```

> **Note**
>
> To see script options, run the script with the `-h` or `--help` argument. Syntax: `./configure-dbms-writeback.sh [--help | -h]`

> **Note**
>
> Optionally run the script with the vault option. This option sets the script to take the DB admin credentials stored in the vault, accessed using the OCID, instead of prompting you for the password. Syntax: `./configure-dbms-writeback.sh [--vault | -V]`

d. When prompted, enter the required information:

- **DB Admin** password, if you did not run the script with vault option. Because the password is protected information, you will not see the text as you type at the command prompt.
- **DB User** username and password. Because the password is protected information, you will not see the text as you type at the command prompt.
- **OCI User** username and auth token. Enter the full user identification string. To find this string, in the OCI Console, click the profile icon in the top-right corner to display an overview of the user profile. Copy the complete string displayed beneath **Profile** and above **Tenancy**.



The script creates the necessary cloud credential and stores it in the database schema. There is no need to rerun the script when you restart OCI, Essbase, or Essbase applications.

> **Note**
>
> To **SSH User**– If the OCI user's auth token used in the data load preparation script no longer has access to the object storage bucket for the Essbase catalog, you will need to find another OCI user meeting the requirements listed in *User Types in Data Load Preparation Workflow*, and repeat the steps in the workflow.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Create a Federated Partition

Create a federated partition between Essbase and Autonomous AI Lakehouse Serverless if you want to use Essbase to make the most of your data warehouse.

> **Note**
>
> An Essbase database with a federated partition is called a federated cube.

This topic assumes you have completed the prerequisites and reviewed the information detailed in the preceding topics.

To create federated partition you must be a Service Administrator or Application Manager.

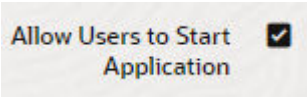
To build a federated cube using Cube Designer, refer to Create a Federated Partition using Cube Designer.

- Redwood
- Classic

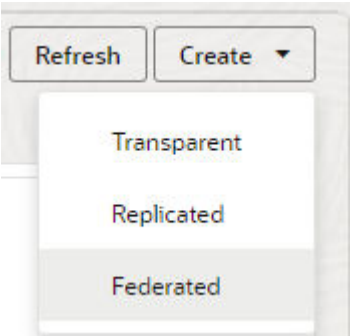
Redwood

1. Log in to the Essbase web interface.
2. On the Home page, click the name of the application you will use to create a federated cube.
3. Click **Settings**, and expand **Startup**.

   Ensure that **Allow Users to Start Application** is enabled.

4. Click **General**, click the **Database** name, and click **Partitions**.
5. Click **Create** >**Federated**.

Copyright © 2019, 2026, Oracle and/or its affiliates.

6. For **Connection name**, enter the connection to Autonomous AI Lakehouse that was previously created, as shown in Create a Connection for Federated Cubes.
7. For **Schema name**, ensure that it matches the name of the database schema (user name that you entered when you created the connection).
8. For **Fact table name**, select the name of the fact table in Autonomous AI Lakehouse that stores numeric values and keys.

> **Note**
>
> Not required for federated partitions with Essbase managed data storage.

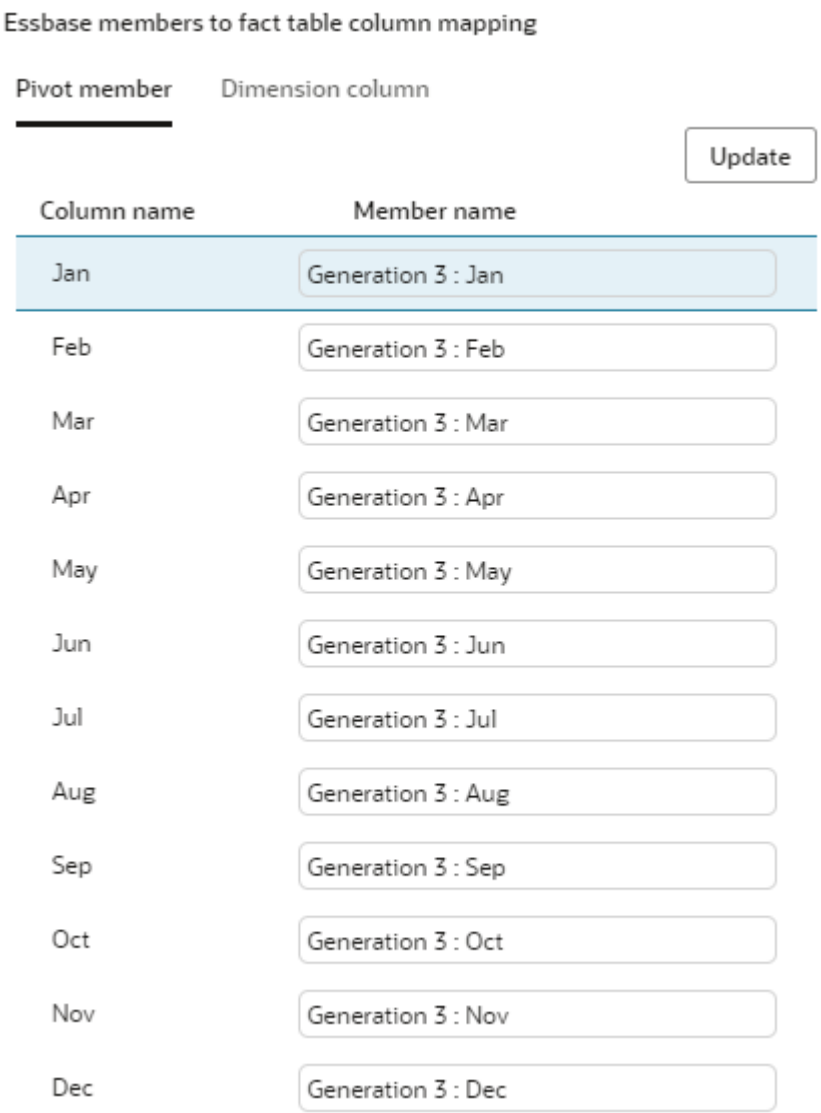
If Essbase recognizes dimension names from the fact table, the values in the **Dimension columns** field may autocomplete with Essbase dimension names. Similarly, the values in the **Pivot member columns** field may autocomplete with members of the presumed pivot dimension.

9. For **Pivot dimension**, select the name of the pivot dimension you decided to use from the Essbase outline, during the Identify the Pivot Dimension process.

   If the column names in the fact table are the same as the dimensions and pivot member names in the outline, then the mapping is automatically populated in in the Pivot member tab of the Essbase members to fact table column mapping. If any dimensions or members cannot be automatically mapped to a column in the fact table, you will need to map them manually.

   Use the **Update** button if you need to manually map stored pivot dimension members to fact table columns.

Copyright © 2019, 2026, Oracle and/or its affiliates.



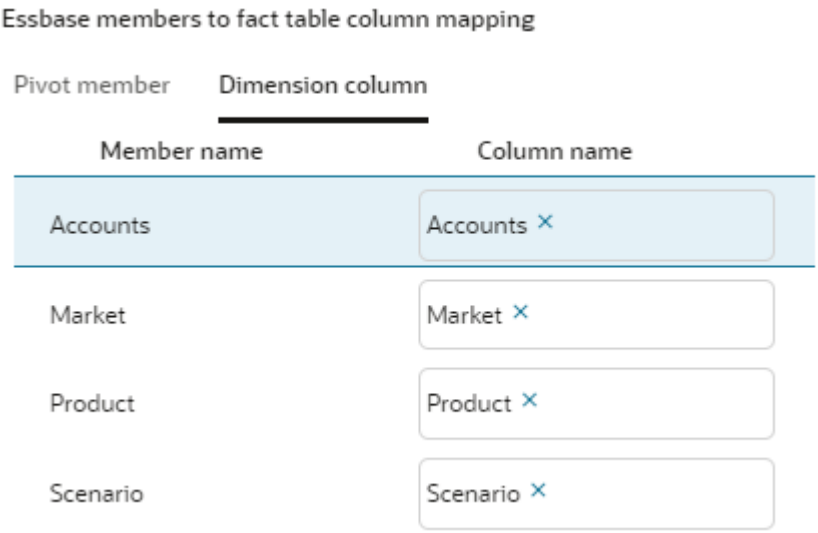
If a member of the pivot dimension (or a non-measures dimension name) includes a special character, such as &, Oracle recommends renaming it.

Essbase automatically maps member names with spaces to corresponding column names in the fact table with the space replaced by an underscore. For example, Essbase automatically makes the following mapping:

Pivot dimension member name	Fact table column name
"Initial Inventory"	INITIAL_INVENTORY

10. Click **Dimension column** to map the non-pivot dimensions to columns in the fact table. They may map automatically if the fact table column names match the dimension names in the outline. If needed, you can map them manually.

Copyright © 2019, 2026, Oracle and/or its affiliates.

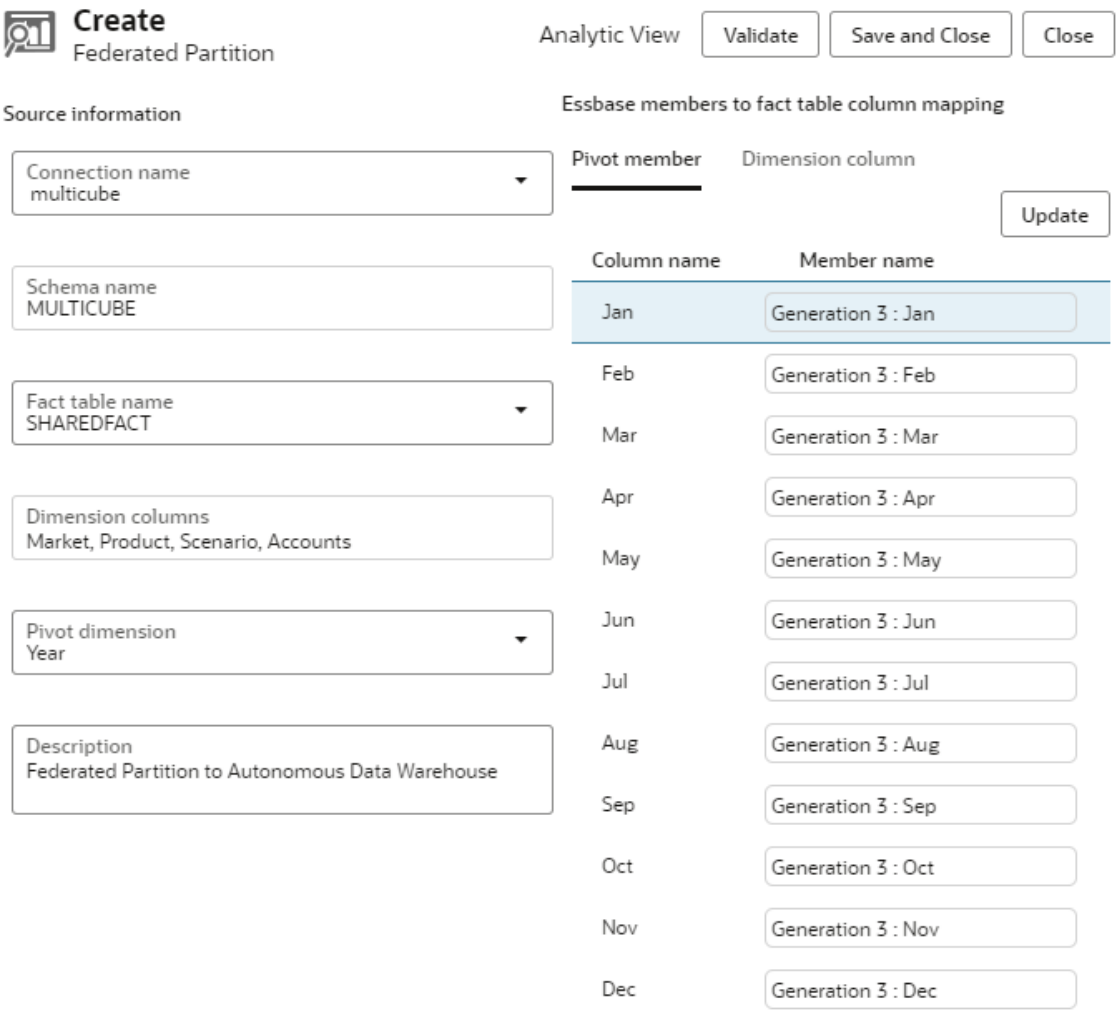


11. Validate and save the partition, then restart the application.

    Saving or validating your federated partition may take a while to complete. Check the job status.

    The federated partition is created, and your Essbase database is now a federated cube. This process also creates dimension helper tables (and other artifacts) in Autonomous AI Lakehouse, which are linked (by keys) to the fact table.

Copyright © 2019, 2026, Oracle and/or its affiliates.



Classic

1. In the Essbase web interface, open the application inspector: on the **Applications** page, find the target application name, click the **Actions** menu, and click **Inspect**.
2. On the **Settings** tab, click **Startup**.

   Ensure that **Allow Users to Start Application** is enabled.
3. In the Essbase web interface, open the cube inspector: on the **Applications** page, expand the target application. In the row for the target cube, click the **Actions** menu, and click **Inspect**.
4. Select the **Partitions** tab.
5. Click **Create** >**Federated**.
6. For **Connection name**, enter the connection to Autonomous AI Lakehouse that was previously created by an administrator or application manager, as shown in Create a Connection for Federated Cubes.
7. For **Schema name**, ensure that it matches the name of the database schema (user name that you entered when you created the connection).
8. For **Fact table name**, select the name of the fact table in Autonomous AI Lakehouse that stores numeric values and keys.

Copyright © 2019, 2026, Oracle and/or its affiliates.

9. For **Pivot dimension**, select the name of the pivot dimension you decided to use from the Essbase outline, during the Identify the Pivot Dimension process.

   If the column names in the fact table are the same as the dimensions and pivot member names in the outline, then the mapping is automatically populated in **Essbase to column map**. If any dimensions or members cannot be automatically mapped to a column in the fact table, you will need to map them manually.

   Use the **Update** button if you need to manually map stored pivot dimension members to fact table columns.

   If a member of the pivot dimension (or a non-measures dimension name) includes a special character, such as &, Oracle recommends renaming it.

   Essbase automatically maps member names with spaces to corresponding column names in the fact table with the space replaced by an underscore. For example, Essbase automatically makes the following mapping:

Pivot dimension member name	Fact table column name
"Initial Inventory"	INITIAL_INVENTORY

10. Click **Dimension column** to map the non-pivot dimensions to columns in the fact table. They may map automatically if the fact table column names match the dimension names in the outline. If needed, you can map them manually.
11. Click **Validate**.
12. If the validation succeeded, click **Save and Close**, and confirm that it is OK to restart the application.

    Saving or validating your federated partition may take a while to complete. Check the job status.

    The federated partition is created. This process also creates dimension helper tables (and other artifacts) in Autonomous AI Lakehouse, which are linked (by keys) to the fact table.

Data Load Options for Federated Cubes

Depending on how you selected to manage your federated data, you can load data using Essbase data load jobs, or, you can use Autonomous AI Database data load tools.

From an Essbase federated cube, you can load data to your fact table in Autonomous AI Lakehouse using Essbase data load methods, or methods available in Autonomous AI Database Data Studio (refer to Autonomous AI Database Data Studio Features).

Copyright © 2019, 2026, Oracle and/or its affiliates.

Loading Data to Federated Cubes through Essbase

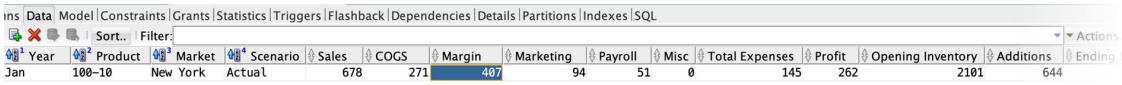
Before performing a data load through Essbase to the fact table, Oracle recommends you upload the data file to the Essbase Server. Client side data load is supported, but takes longer.

Loading Essbase-formatted data export files into federated cubes can be time consuming. To optimize data loads, use a DBMS formatted source file. You can make one using the DATAEXPORT calculation command with DataExportCSVFormat option. CSV formatted files can be loaded faster because they are in accordance with DBMS_CLOUD package format options for source files.

Pivot Dimension in Fact Table and Data Load Input Source

The pivot dimension used in data load input files must be the same as the pivot dimension of the fact table.

For example, in the following fact table, the pivot dimension is the Measures dimension (Sales, COGS, Margin, etc).



ins | Data | Model | Constraints | Grants | Statistics | Triggers | Flashback | Dependencies | Details | Partitions | Indexes | SQL

Sort.. Filter: Actions

Year	Product	Market	Scenario	Sales	COGS	Margin	Marketing	Payroll	Misc	Total Expenses	Profit	Opening Inventory	Additions	Ending
Jan	100-10	New York	Actual	678	271	407	94	51	0	145	262	2101	644	

An acceptable data load input file for this fact table has a similar shape, because it has the same pivot dimension. Example (truncated):

```
"Year","Product","Market","Scenario","Sales","COGS","Margin","Marketing","Payr
oll","Misc","Total Expenses","Profit","Opening Inventory","Additions","Ending
Inventory"
"Jan","100-10","New York","Actual",678,271,407,94,51,0,145,262,2101,644,2067
"Feb","100-10","New York","Actual",645,258,387,90,51,1,142,245,2067,619,2041
"Mar","100-10","New York","Actual",675,270,405,94,51,1,146,259,2041,742,2108
```

If the input file pivot dimension differs from the fact table pivot dimension, the load might take more time.

Multiple Import Files in MaxL not Supported

Importing data from multiple files in parallel using a MaxL **import** statement with wildcard characters is not supported for federated cubes.

Loading Data directly to Federated Cubes without using Essbase Data Load

> **Caution**
>
> This information applies only if you select to manage your own data in Autonomous AI Lakehouse. Do not use non-Essbase data load tools for Essbase-managed federated cubes.

If you do not need to load data through Essbase to Autonomous AI Lakehouse, you can use Data Tools in Autonomous AI Database to load data to the fact table and perform other management tasks. However, ensure that the cube outline and fact table do not get out of sync – see Metadata Precautions for Federated Partition Cubes.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Oracle does not recommend loading an Essbase-managed fact table with non-Essbase tools.

If you load data directly to the fact table without using Essbase tools, take the following precautions:

- Do not load duplicate rows to the fact table.
- Intersections must be for stored data.
- The data in your source file must use the exact member names (or qualified member names) that are used in Essbase. Do not load using aliases. For duplicate-member enabled cubes, the correct qualified member name format is required. The syntax is described in Duplicate Member Names in Outlines.
- To load data into Essbase-managed federated cubes, the rows in your source data must contain member IDs rather than member names.

Refer to Autonomous AI Database Data Studio Features for information about loading data into Autonomous AI Database using Data Studio.

Refer to the Oracle Autonomous AI Database 15 Minute Quick Start Livelab for an overview of how to deploy Autonomous AI Database on OCI.

Calculate and Query Federated Cubes

When you have an Essbase federated cube, calculations and queries, whenever possible, are converted by Essbase into SQL and pushed to Autonomous AI Lakehouse, so that processing occurs where the data is stored.

The Essbase cube outline contains the metadata (dimension and member names). Autonomous AI Lakehouse holds the data associated with the metadata. Data is stored in a fact table.

Because Essbase pushes calculation processing to where the data is stored, it helps solve data latency problems. This functionality is new beginning with Essbase 21.5.

If you are new to Essbase, start with reviewing Calculate Cubes to learn about calculation in general.

How calculation works with federated cubes depends on the type of Essbase cube you started with to create the federated cube: block storage (BSO), or aggregate storage (ASO).

Calculation Comparison Notes

For BSO-based federated cubes, when you calculate and query, Essbase analyzes dependencies and writes SQL to process the results using Analytic Views in Autonomous AI Lakehouse.

For ASO-based federated cubes, Essbase processes custom calculations and allocations, and then pushes the results to Autonomous AI Lakehouse.

Notes / Limitations on Federated Cube Calculations

Essbase block storage (BSO) calculation functions may be applied in BSO outline formulas, with their results impacting queries from Smart View, MDX, and other grid clients. The same functions, along with the calculation commands, can be used to write procedural calculation scripts.

Calculation Functions that Process in Autonomous AI Lakehouse

Copyright © 2019, 2026, Oracle and/or its affiliates.

In a federated cube, the following calculation functions are translated to SQL and processed in Autonomous AI Lakehouse. Other functions not listed here are processed in Essbase.

- @ABS
- @ALLANCESTORS
- @ANCEST
- @ANCESTORS
- @AVG (with SKIPMISSING option only)
- @AVGRANGE
- @CHILDREN
- @CURRMBR
- @DESCENDANTS
- @EXP
- @FACTORIAL
- @GENMBRS
- @IALLANCESTORS
- @IANCESTORS
- @ICHILDREN
- @IDESCENDANTS
- @INT
- @IRDESCENDANTS
- @ISANCEST
- @ISCHILD
- @ISDESC
- @ISGEN
- @ISIANCEST
- @ISIBLINGS
- @ISICHILD
- @ISIDESC
- @ISIPARENT
- @ISISIBLING
- @ISLEV
- @ISMBR (when argument is only one member name)
- @ISPARENT
- @ISSAMEGEN
- @ISSAMELEV
- @ISSIBLING
- @LEVMBRS
- @LN

Copyright © 2019, 2026, Oracle and/or its affiliates.

- @LOG
- @LOG10
- @LSIBLINGS
- @MAX
- @MAXRANGE (exception: no *XrangeList* argument)
- @MAXS
- @MAXSRANGE (exception: no *XrangeList* argument)
- @MBRPARENT
- @MEDIAN (exception: no *XrangeList* argument)
- @MEMBERAT
- @MIN
- @MINRANGE (exception: no *XrangeList* argument)
- @MINS
- @MINSRANGE (exception: no *XrangeList* argument)
- @MOD
- @PARENT
- @POWER
- @RDESCENDANTS
- @RELATIVE
- @REMAINDER
- @ROUND
- @RSIBLINGS
- @SIBLINGS
- @SUM
- @SUMRANGE (exception: no *XrangeList* argument)
- @TRUNCATE
- @XREF
- @XWRITE

Calculation Commands that Process in Autonomous AI Lakehouse

In a federated cube, the following Essbase calculation commands are translated to SQL and processed in Autonomous AI Lakehouse.

- AGG (except when aggregating Dynamic Calc members or members using non additive consolidation operator)
- CLEARDATA
- CLEARBLOCK (exception: no NONINPUT nor DYNAMIC keywords)
- DATAEXPORT (exception: only with the following data export options)

  ```
  DATAEXPORTLEVEL ALL
  DATAEXPORTCSVFORMAT
  ```


Copyright © 2019, 2026, Oracle and/or its affiliates.

```
DATAEXPORTOVERWRITEFILE
DATAEXPORTDECIMAL
```

- IF...ENDIF
- ELSE...ELSEIF (expressions with multiple, nested IF / ELSE statements may have slower performance)
- EXCLUDE...ENDEXCLUDE
- LOOP...ENDLOOP
- DATACOPY
- FIX statement assignments with expressions containing mathematical operations, IF / ELSE statements, cross references, and supported @ functions listed on this page.

Commands ARRAY and VAR, as well as dynamic formulas processed in CALC DIM or CALC ALL, are processed in Essbase and may have slower performance.

Some calculation commands are not supported for federated cubes, and return an error if used. See Restrictions for Federated Cubes.

If you need to run Essbase block storage (BSO) calculation scripts, select a dense dimension as the pivot dimension. Calculation scripts are not supported for federated cubes if the pivot dimension is sparse.

Block calculation mode (enabled when Essbase configuration setting CALCMODE is set to BLOCK) is not applicable for federated cubes. Calculation processing is pushed to Autonomous AI Lakehouse. If an exception exists and the calculation is processed on the Essbase Server instead, then solve order determines the dependency analysis.

When performing custom allocations on a federated cube based on ASO, you can only override existing values. You cannot add to, nor subtract from, existing values.

Other Limitations

See Restrictions for Federated Cubes.

Precision Digits in Query Results

When you calculate a federated cube, Autonomous AI Lakehouse partially processes the calculations and aggregations. Therefore, query results may have slightly different precision values if compared to the values obtained in a non-federated cube.

Calculation Order

The calculation priority of members in federated cubes follow a defined solve order that you set on the Essbase outline.

Ability to Run Essbase Calculations and Data Load Jobs

The Essbase configuration setting FEDERATEDAVCALC is implicitly set to TRUE by default for federated cubes based on block storage. This enables users to run Essbase BSO calculations and to perform data loads through Essbase to update records in the fact table.

Federated Cube Maintenance and Troubleshooting

Use the following guidelines to maintain or troubleshoot Essbase federated cubes.

This topic assumes you have created a federated partition and reviewed the information detailed in the preceding topics.

Copyright © 2019, 2026, Oracle and/or its affiliates.

- Model and Test Federated Cubes
- Metadata Precautions for Federated Cubes
- Cancel Long Running SQL on Federated Cubes
- What to Do if the Database Connection Details Changed
- Back up and Restore a Federated Cube

Model and Test Federated Cubes

When designing an Essbase federated cube, follow these design and testing guidelines, for a phased approach to troubleshoot or monitor performance.

- Begin the federated cube project on a test environment.
- Start with Essbase ASO or BSO cube models that have the following characteristics:
  - not many levels
  - not many shared members or attributes

1. When creating a federated partition, schedule offline operations when queries are not allowed against the instance.
2. Gradually disconnect active Essbase user sessions, using MaxL alter application `disable commands` and/or `disable connects` (to prevent any new user activity), followed by alter system `logout session` and/or `kill request` (if you need to terminate any active sessions that don't need to complete). Note that MaxL cannot terminate any requests that may be running in Autonomous AI Lakehouse. If you disable commands in the application, remember to re-enable commands after creating the federated partition.

   To terminate requests that are running as SQL sessions in Autonomous AI Lakehouse, additional steps are required. See Cancel Long Running SQL on Federated Cubes.
3. Perform timeout tuning:
   - HTTPS proxy on customer network - adjust customer network timeouts
   - Load balancer - increase LoadBalance timeout to 1260 seconds (21 minutes)
   - Increase HTTPD timeouts to 21 minutes

     ```
     /etc/httpd/conf.d/00_base.conf:ProxyTimeout 1260
     ```

     ```
     /etc/httpd/conf.d/00_base.conf:Timeout 1260
     ```

   - APS/JAPI timeout:
     - On the **Console** page in the Essbase web interface, select **Configuration**, and note the value of olap.server.netSocketTimeOut. A value of 200 ms means that every count of 5 for these properties gives 1 second of time-wait.
     - To set APS/JAPI timeout limit to 30 minutes, set `olap.server.netRetryCount` to 9000.
4. Create the federated partition.
5. Revert the timeout adjustments in step 3.
6. Enable users back onto the system using alter application enable commands and/or connects, if these were disabled previously.

Copyright © 2019, 2026, Oracle and/or its affiliates.

7. For reports on a federated cube, tune QRYGOVEXECTIME to be larger than the expected time to execute queries against federated partitions. Note that QRYGOVEXECTIME cannot terminate any requests that may be running in Autonomous AI Lakehouse.
8. After development environment testing and tuning are completed, then use the above steps 1 through 7 to implement a production environment.

> **Note**
>
> If you see a "Failed to save outline" error when creating the federated partition, wait for the sessions to complete, then refresh the browser. If the federated partition has been created, then validate it in SQL Developer. If it validates in SQL Developer then the federated cube is ready for use. Otherwise, the model needs to be fixed and timeout tuning is needed as described above in step 3.

Metadata Precautions for Federated Cubes

Take care when editing the outline for a non-Essbase managed, federated cube. If you add or rename members, ensure that the metadata changes are also represented in your fact table in Autonomous AI Lakehouse.

This topic does not apply for federated cubes with Essbase managed data storage.

If the Essbase outline becomes out of sync with the fact table in Autonomous AI Lakehouse, the federated cube will become invalid or not function correctly. To fix it, you will need to drop the federated partition, make changes to the outline and fact table, and then re-create the federated partition.

If a federated cube becomes invalid, you may encounter an error beginning with `Essbase Error(1040235): Remote warning from federated partition`.

The following types of Essbase outline changes will cause a (non-Essbase managed) federated cube to become invalid:

- Adding, renaming, or removing dimensions
- Adding, renaming, or removing stored members in the pivot dimension
- Changing any member from stored to dynamic

For other types of Essbase outline changes not indicated above (for example, adding or renaming a non-pivot-dimension member), you should make the corresponding change to the affected data row in the fact table. Otherwise, the federated cube may not function correctly.

If you know in advance that Essbase outline metadata will change, it is better to remove the federated partition first, make the outline changes, update the fact table, and then recreate the federated partition.

However, if the Essbase metadata changed and caused the federated partition to become invalid, take the following action:

1. Remove the federated partition, and the connection associated with it (if otherwise unused), as described in Remove a Federated Partition.

   From the user schema in Autonomous AI Lakehouse, manually delete any Essbase-generated tables and other objects that failed to be removed with the federated partition.
2. Ensure that the outline changes are completed in the Essbase cube.
3. Create the fact table again. See Create the Fact Table.

Copyright © 2019, 2026, Oracle and/or its affiliates.

4. Re-create the connection to Autonomous AI Lakehouse. This may be a global connection (under the main Sources icon in Essbase web interface), or it may be in the Sources defined just for the application. Follow the instructions in Create a Connection for Federated Cubes.
5. Re-create the federated partition, as described in Create a Federated Partition.

Cancel Long Running SQL on Federated Cubes

You can cancel SQL sessions that are running on Essbase federated cubes. SQL cancellation may be necessary if an calculation is taking too long, or if the federated cube needs backup or recovery.

If Essbase calculations, data loads, or aggregations running on federated cubes generate long running SQL statements, it is possible to cancel the sessions, even though they are running on Autonomous AI Lakehouse.

Before SQL session cancellation through Essbase can become available, a brief workflow of steps must be completed to enable SQL Session Governor in Essbase.

Users with following types of access are involved in the preparation steps.

User Type	Role within Workflow
DB Admin	Knows the administrator schema name and password for Autonomous AI Lakehouse. Can create stored PL/SQL procedures on Autonomous AI Lakehouse, and grant privileges to other users.
DB User Also referred to as **federated partition user**	Knows the Autonomous AI Lakehouse schema name and password -- the same schema and password that is used to create the Essbase connection to Oracle Database (a required prerequisite before creating the federated partition).
Essbase Application Manager	The manager/owner of an Essbase application used for a federated cube.

The following workflow of steps must be completed to enable SQL Session Governor:

1. **DB Admin**: Create the following PL/SQL stored procedures in Autonomous AI Lakehouse and grant EXECUTE privilege to the **DB User**:

   > **Note**
   >
   > The stored procedures are to be created on the Database user schema.

   a. ESS_SESSION_SET_MODULE (*jagent_id* in VARCHAR2,*app_name* in VARCHAR2, *db_name* in VARCHAR2, *req_id* in VARCHAR2)

      Use the following statement to create the procedure:

```
CREATE OR REPLACE NONEDITIONABLE PROCEDURE "ESS_SESSION_SET_MODULE"
(jagent_id in VARCHAR2, app_name in VARCHAR2, db_name in VARCHAR2,
req_id in VARCHAR2) AS
    ess_jagentid VARCHAR2(256);
    ess_appname VARCHAR2(256);
    ess_dbname VARCHAR2(256);
```


Copyright © 2019, 2026, Oracle and/or its affiliates.

```
    ess_reqid VARCHAR2(256);
    hash_str VARCHAR2(100);
BEGIN
    IF jagent_id is not NULL AND app_name is not NULL AND db_name is
not NULL AND req_id is not NULL THEN
        -- add prefixes to identify essbase sessions
        ess_jagentid := CONCAT('ess_jagentid_', jagent_id);
        ess_appname := CONCAT('ess_appname_', app_name);
        ess_dbname := CONCAT('ess_dbname_', db_name);
        ess_reqid := CONCAT('ess_reqid_', req_id);

        -- ess_jagentid is stored in client_info
        dbms_application_info.set_client_info(ess_jagentid);

        -- create hash out of CONCAT(ess_jagentid, ess_appname,
ess_dbname)
        -- the hash identifies the fed partition
        -- store the hash as the module
        -- store the ess_reqid as the action
        SELECT TO_CHAR(ORA_HASH(CONCAT(CONCAT(ess_jagentid,
ess_appname), ess_dbname))) INTO hash_str from dual;
        dbms_application_info.set_module(module_name => hash_str,
action_name => ess_reqid);

    ELSIF jagent_id is NULL AND app_name is NULL AND db_name is NULL
AND req_id is NULL THEN
        -- reset session
        dbms_application_info.set_module(null,null);
        dbms_application_info.set_client_info(null);
    END IF;
END ESS_SESSION_SET_MODULE;
```

b. ESS_SESSION_CANCEL (*jagent_id* in VARCHAR2, *app_name* in VARCHAR2, *db_name* in VARCHAR2, *req_id* in VARCHAR2)

Use the following statement to create the procedure:

```
CREATE OR REPLACE NONEDITIONABLE PROCEDURE "ESS_SESSION_CANCEL"
(jagent_id in VARCHAR2, app_name in VARCHAR2, db_name in VARCHAR2,
req_id in VARCHAR2) AS
    ess_jagentid VARCHAR2(256);
    ess_appname VARCHAR2(256);
    ess_dbname VARCHAR2(256);
    ess_reqid VARCHAR2(256);
    uname VARCHAR2(100);
    hash_str VARCHAR2(100);
    sessionid VARCHAR2(100);
    killcmd VARCHAR2(1000);
    XCP_MARKED_FOR_TERMINATION exception;
    pragma exception_init (XCP_MARKED_FOR_TERMINATION, -00031);
BEGIN
    -- add prefixes to identify Essbase sessions
    ess_jagentid := CONCAT('ess_jagentid_', jagent_id);
    ess_appname := CONCAT('ess_appname_', app_name);
    ess_dbname := CONCAT('ess_dbname_', db_name);
    ess_reqid := CONCAT('ess_reqid_', req_id);
```


Copyright © 2019, 2026, Oracle and/or its affiliates.

```
    -- create hash out of CONCAT(ess_jagentid, ess_appname, ess_dbname)
    -- the hash identifies the federated partition
    SELECT TO_CHAR(ORA_HASH(CONCAT(CONCAT(ess_jagentid, ess_appname),
ess_dbname))) INTO hash_str from dual;
    uname := SYS_CONTEXT('USERENV', 'SESSION_USER');

    IF jagent_id is not NULL AND app_name is not NULL AND db_name is
not NULL AND req_id is not NULL THEN
        -- valid jagent_id, app_name, db_name, req_id
        -- kill sessions associated with the given federated cube and
request id
        FOR r IN (select sid,serial# from v$session where
username=uname AND module=hash_str AND action=ess_reqid)
        LOOP
            BEGIN
                sessionid := DBMS_ASSERT.ENQUOTE_LITERAL(r.sid || ','
|| r.serial#);
                killcmd:='ALTER SYSTEM KILL SESSION ' || sessionid || '
IMMEDIATE';
                EXECUTE IMMEDIATE killcmd;
            EXCEPTION
                WHEN XCP_MARKED_FOR_TERMINATION THEN
                    -- ignore - the session could not be terminated
immediately, but is marked for termination.
                    NULL;
            END;
        END LOOP;
    ELSIF jagent_id is not NULL AND app_name is not NULL AND db_name is
not NULL AND req_id is NULL THEN
        -- null req_id
        -- kill all sessions for the given federated cube - invoked at
app startup
        FOR r IN (select sid,serial# from v$session where
username=uname AND module=hash_str)
        LOOP
            BEGIN
                sessionid := DBMS_ASSERT.ENQUOTE_LITERAL(r.sid || ','
|| r.serial#);
                killcmd:='ALTER SYSTEM KILL SESSION ' || sessionid || '
IMMEDIATE';
                EXECUTE IMMEDIATE killcmd;
            EXCEPTION
                WHEN XCP_MARKED_FOR_TERMINATION THEN
                    -- ignore - the session could not be terminated
immediately, but is marked for termination.
                    NULL;
            END;
        END LOOP;
    END IF;

    EXCEPTION
        WHEN OTHERS THEN
            RAISE;
END ESS_SESSION_CANCEL;
```


Copyright © 2019, 2026, Oracle and/or its affiliates.

c. Grant EXECUTE privileges for the stored procedures above to the **DB User**.

```
GRANT EXECUTE on ESS_SESSION_SET_MODULE TO "DB User";
GRANT EXECUTE on ESS_SESSION_CANCEL TO "DB User";
CREATE OR REPLACE SYNONYM "DB User".ESS_SESSION_SET_MODULE FOR
ESS_SESSION_SET_MODULE;
CREATE OR REPLACE SYNONYM "DB User".ESS_SESSION_CANCEL FOR
ESS_SESSION_CANCEL;
```

2. **DB User**: In your Autonomous AI Lakehouse schema, confirm that you have access to the following stored procedures under **Synonyms**:
   - ESS_SESSION_SET_MODULE
   - ESS_SESSION_CANCEL
3. With these prerequisites met, long running SQL requests on the federated cube can be cancelled the same way other Essbase requests can. Requests can be cancelled in the Essbase web interface from the **Sessions** area of the **Console**. If you are using MaxL, you can cancel requests using the statement alter system `kill request`. Review Model and Test Federated Partition Cubes for guidelines about session termination.

What to Do if the Database Connection Details Changed

If the Autonomous AI Lakehouse connection details that Essbase uses for a federated cube have changed, you will need to drop and re-create the federated partition, and clean up associated objects and metadata tables from the database schema.

This topic does not apply for federated cubes with Essbase managed data storage.

You will need to drop and re-create the federated partition if any of the following events occur after it was created:

- The Autonomous AI Lakehouse port changes
- The connection name changes
- The connection uses a wallet, and you switch from one service name to another (to make performance or concurrency changes)
- An outline update changes the member mapping to the fact table, causing the federated partition to become out of sync. See Metadata Precautions for Federated Cubes for details.

If you know in advance that the connection details will change, it is better to remove the federated partition before the change occurs, and create it again after. However, if the connection changed and caused the federated cube to become invalid, take the steps that follow.

Drop the Federated Partition

Remove the federated partition, and the connection associated with it (if otherwise unused), as described in Remove a Federated Partition.

Clean Up Federated Partition Related Objects and Metadata Tables

From the user schema in Autonomous AI Lakehouse, drop any Essbase-generated tables and other objects that failed to be removed with the federated partition.

Copyright © 2019, 2026, Oracle and/or its affiliates.

1. ssh to the Essbase Server host as the **opc** user.

   ```
   ssh -i MPOCI_KEY.pem opc@100.xxx.xxx.xxx
   ```

2. Change to **oracle** user (and go to their home directory).

   ```
   sudo su - oracle
   ```

3. Navigate to the applications directory.

   ```
   cd /u01/data/essbase/app
   ```

4. Using the Essbase application and cube name, identify the unique prefix associated with your federated partition objects and metadata.

   a. Get the application name (*AppName*). The name is case sensitive, so capture the exact case. In this example, *AppName* = Sample.

      ```
      ls
      Sample
      ```

   b. Navigate to the cube directory and get the cube name (*DbName*). The name is case sensitive, so capture the exact case. In this example, *DbName* = Basic.

      ```
      cd /Sample
      ls
      Basic
      ```

   c. Using SQL Developer or another tool, connect to Autonomous AI Database as the user of the schema to which the federated cube is (or was) connected.

   d. In SQL Developer, run a SELECT statement to get the value of *OTL_ID* and *OTL_STATUS*.

      The SELECT statement format is:

      ```
      SELECT OTL_ID, OTL_STATUS, TRC_HOST FROM ESSAV_OTL_MTD_VERSION where
      APPNAME ='<AppName>';
      ```

      Example

      Replace `Sample` with your *AppName*.

      ```
      SELECT OTL_ID, OTL_STATUS, TRC_HOST FROM ESSAV_OTL_MTD_VERSION WHERE
      APPNAME='Sample';
      ```

      The above query should return the value of *OTL_ID*; for example,

      ```
      987
      ```


Copyright © 2019, 2026, Oracle and/or its affiliates.

> **Note**
>
> If any rows are returned where OTL_STATUS is `ACTIVE`, you should attempt to drop the application from the TRC_HOST instance before you continue to the next step.

e. Build the *Prefix* as:

```
ESSAV_<OTL_ID>_
```

Example:

```
<Prefix> = ESSAV_987_
```

5. Run a SELECT statement to create a list of objects associated with your federated cube. These are the objects you will clean up in the next step.

   The SELECT statement format is:

```
SELECT * FROM user_OBJECTS WHERE OBJECT_NAME LIKE '<Prefix>%' ESCAPE '\';
```

   Example:

```
SELECT * FROM user_OBJECTS WHERE OBJECT_NAME LIKE 'ESSAV_987_%' ESCAPE '\';
```

6. Run a stored PL/SQL procedure that cleans up all the analytic views, packages, hierarchies, tables, and other objects associated with the *Prefix*.

   Example

   Replace `ESSAV_987_` with your *Prefix*.

```
SET SERVEROUTPUT ON;

declare
  prefix_str varchar2(70) := 'ESSAV_987_';

BEGIN

  FOR c IN ( SELECT ANALYTIC_VIEW_NAME FROM user_analytic_views WHERE
ANALYTIC_VIEW_NAME like prefix_str || '%' )
  LOOP
    EXECUTE IMMEDIATE 'DROP ANALYTIC VIEW "' || c.ANALYTIC_VIEW_NAME || '"
';
    DBMS_OUTPUT.PUT_LINE('ANALYTIC VIEW ' || c.ANALYTIC_VIEW_NAME || '
dropped successfully.');
  END LOOP;

  FOR c IN ( SELECT distinct OBJECT_NAME FROM USER_PROCEDURES WHERE
OBJECT_TYPE='PACKAGE' and OBJECT_NAME like prefix_str || '%' )
  LOOP
    EXECUTE IMMEDIATE 'DROP PACKAGE "' || c.OBJECT_NAME || '" ';
    DBMS_OUTPUT.PUT_LINE('PACKAGE ' || c.OBJECT_NAME || ' dropped
successfully.');
  END LOOP;
```


Copyright © 2019, 2026, Oracle and/or its affiliates.

```
  FOR c IN ( SELECT distinct HIER_NAME FROM USER_HIERARCHIES WHERE
HIER_NAME like prefix_str || '%' )
  LOOP
    EXECUTE IMMEDIATE 'DROP HIERARCHY "' || c.HIER_NAME || '" ';
    DBMS_OUTPUT.PUT_LINE('HIERARCHY ' || c.HIER_NAME || ' dropped
successfully.');
  END LOOP;

  FOR c IN ( SELECT distinct DIMENSION_NAME FROM
USER_ATTRIBUTE_DIM_TABLES_AE WHERE DIMENSION_NAME like prefix_str || '%' )
  LOOP
    EXECUTE IMMEDIATE 'DROP ATTRIBUTE DIMENSION "' || c.DIMENSION_NAME ||
'" ';
    DBMS_OUTPUT.PUT_LINE('ATTRIBUTE DIMENSION ' || c.DIMENSION_NAME || '
dropped successfully.');
  END LOOP;

  FOR c IN ( SELECT distinct VIEW_NAME FROM USER_VIEWS WHERE VIEW_NAME
like prefix_str || '%' )
  LOOP
    EXECUTE IMMEDIATE 'DROP VIEW "' || c.VIEW_NAME || '" ';
    DBMS_OUTPUT.PUT_LINE('VIEW ' || c.VIEW_NAME || ' dropped
successfully.');
  END LOOP;

  FOR c IN ( SELECT distinct TABLE_NAME FROM USER_TABLES WHERE TABLE_NAME
like prefix_str || '%' )
  LOOP
    EXECUTE IMMEDIATE 'DROP TABLE "' || c.TABLE_NAME || '" purge';
    DBMS_OUTPUT.PUT_LINE('TABLE ' || c.TABLE_NAME || ' dropped
successfully.');
  END LOOP;

  FOR c IN ( SELECT distinct TYPE_NAME FROM USER_TYPES WHERE TYPE_NAME
like prefix_str || '%' )
  LOOP
    EXECUTE IMMEDIATE 'DROP TYPE "' || c.TYPE_NAME || '" FORCE';
    DBMS_OUTPUT.PUT_LINE('TYPE ' || c.TYPE_NAME || ' dropped
successfully.');
  END LOOP;

END;
/
```

7. Drop and update associated metadata-related tables.

   a. Run a stored PL/SQL procedure to drop metadata-related tables associated with the *OTL_ID*.

      Example

      Replace 987 with your *OTL_ID*.

```
SET SERVEROUTPUT ON;
BEGIN
  FOR c IN ( SELECT distinct TABLE_NAME FROM USER_TABLES WHERE
```


Copyright © 2019, 2026, Oracle and/or its affiliates.

```
TABLE_NAME like 'ESSAV_MTD_987_%' )
  LOOP
    EXECUTE IMMEDIATE 'DROP TABLE "' || c.TABLE_NAME || '" purge';
    DBMS_OUTPUT.PUT_LINE('TABLE ' || c.TABLE_NAME || ' dropped
successfully.');
  END LOOP;
END;
/
```

b. Run an UPDATE statement to set the ESSAV_OTL_MTD_VERSION table to inactive status.

Example

Replace `Sample` with your *AppName*.

```
UPDATE "ESSAV_OTL_MTD_VERSION" SET  "OTL_STATUS" = 'INACTIVE'  where
APPNAME ='Sample';
commit;
```

> **Note**
>
> As only one Essbase instance should be using the Autonomous AI Database schema, you should not need to specify the ESSBASE_INSTANCE_UNIQUE_ID when you are performing the cleanup tasks. Refer to Restrictions for Federated Cubes.

Recreate the Connection and Federated Partition

1. Re-create the connection to Autonomous AI Lakehouse. This may be a global connection (under the main Sources icon in Essbase web interface), or it may be in the Sources defined just for the application. Follow the instructions in connection. Make sure to Test and Save the connection.
2. Re-create the federated partition, as described in Create a Federated Partition.
3. If you continue to see a connection error such as `Essbase Error(1350012): Attempt to connect to OCI failed`, check https://support.oracle.com/rs?type=doc&id=2925030.1.

Back up and Restore a Federated Cube

Federated partitions are not migrated with Essbase applications. When preparing to move your federated cube to another server or to migrate to another Essbase version, you need to delete the federated partition and recreate it in the new environment.

To back up your federated cube,

1. Back up the application, without the data, but including everything else you may need (such as configuration properties, filters, variables, calculation scripts, and other artifacts). To do this, use LcmExport (or the Export LCM job in the Essbase web interface).
2. Back up the fact table. See Backing Up and Restoring Autonomous Database.
3. Delete the federated partition definition from the cube, following the steps in Remove a Federated Partition.

To restore your federated cube from backup,

Copyright © 2019, 2026, Oracle and/or its affiliates.

1. Re-create the application, using LcmImport (or the Import LCM job in the Essbase web interface).
2. If necessary, restore the fact table on Autonomous AI Lakehouse.
3. Re-create the connection to Autonomous AI Lakehouse. It is recommended to use a new connection name to avoid encountering errors.
4. Re-create the federated partition.

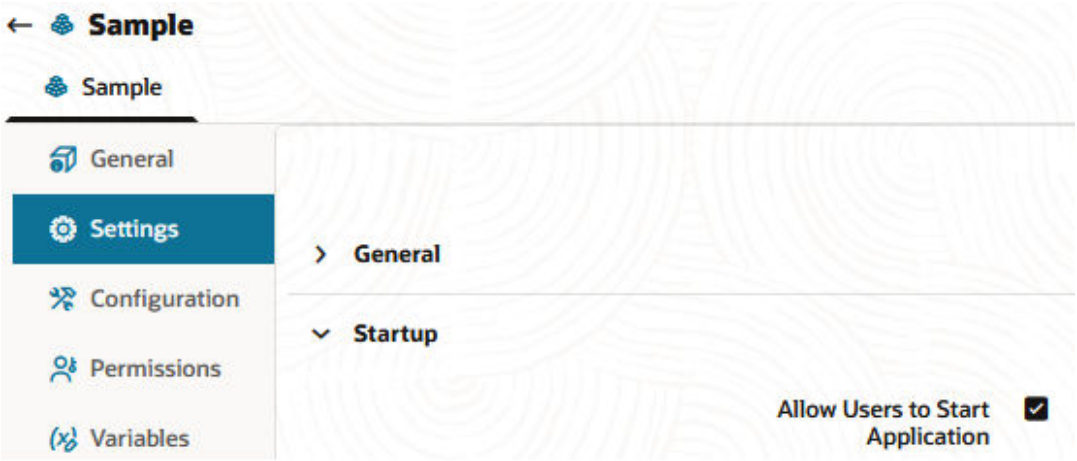
Remove a Federated Partition

Federated partitions are not migrated, so when moving your application to another server or version, you need to delete the federated partition and recreate it in the new environment.

When you need to remove a federated partition, take the following actions to ensure that associated tables are cleaned up in Autonomous AI Lakehouse.

1. Ensure that in the application settings, startup is enabled.

   In the Essbase web interface, the setting is controlled by the **Allow Users to Start Application** check box.



   In MaxL, the setting is controlled by:

```
alter application APP-NAME enable startup;
```

2. Delete the federated partition from the application. This action removes all of the Essbase helper tables and associated artifacts from Autonomous AI Lakehouse (but does not remove the fact table).
   a. Log in to the Essbase web interface as a database manager or higher.
   b. On the Home page, open the application and open the cube.
   c. Select **Partitions**.
   d. Click the Actions menu to the right of the partition definition, and click **Delete**.
   e. Click Yes to confirm you want to delete the partition and allow the application to restart.
3. Remove the connection, if one was created at application level, and was intended only for the federated cube. If the federated cube was designed using a global connection, it is possible that the connection may be in use for additional purposes in your organization. If you aren't sure, check with a system administrator.

Copyright © 2019, 2026, Oracle and/or its affiliates.

4. If the federated partition had to be removed due to a Database port change, you may need to use SQL Developer to manually delete Essbase-generated tables and other artifacts, if they failed to be removed with the federated partition. The table names begin with ESSAV_. For more details, refer to What to Do if the Database Connection Details Changed.

Differences between Aggregate Storage, Block Storage, and Federated Cubes

If you already have an Essbase ASO or BSO application, you can make it into a federated partition application.

Use this page in planning stages, if you need to compare characteristics of Essbase block storage (BSO) and aggregate storage (ASO) cubes, side by side with federated partition cubes.

	Aggregate Storage (ASO)	**Block Storage (BSO)**	**Federated Partition Cube**
Data storage model	Data is stored in Essbase.	Data is stored in Essbase.	Data is stored in a relational table in Autonomous AI Lakehouse. Elsewhere in the documentation, it is referred to as the *fact table*.

Copyright © 2019, 2026, Oracle and/or its affiliates.

	Aggregate Storage (ASO)	Block Storage (BSO)	Federated Partition Cube
How it works	Number of dimensions can be very high, containing millions of members, but the cube has relatively sparse data slices (many dimensional intersections contain no data). Data is input at level 0 only. Cubes are optimized for rapid aggregation.	The number and scale of dimensions are typically smaller as compared with ASO. BSO accomodates dense data sets. Some of the dimensions are defined as dense, with data at most intersections, and others are defined as sparse. This helps Essbase store data efficiently and optimize dependency analysis (so as not to overcalculate). Data can be input at any level.	The Essbase outline is mapped to the fact table, allowing data storage to remain in Autonomous AI Lakehouse, while being accessible for analysis using the logic you build into your Essbase application. The analytical capabilities of your Essbase outline enable you to analyze the flat relational table as hierarchies, employing whatever complex procedural math you may need for your multidimensional analysis. Calculations and aggregations, when possible, are converted by Essbase into SQL and pushed to Autonomous AI Lakehouse, so that processing occurs closer to where data is stored. You can find the SQL Essbase writes in the platform log, located in `<DOMAIN_HOME>/servers/essbase_server1/logs/essbase`.

Copyright © 2019, 2026, Oracle and/or its affiliates.

	Aggregate Storage (ASO)	Block Storage (BSO)	Federated Partition Cube
Typical use cases	ASO cubes are commonly used for highly aggregational analytics, custom calculations, and allocations. Data loads can be broken into slices for frequent, highly parallelized updates.	BSO cubes are commonly used for financial and operational planning, and interactive reporting on aggregate data relative to the source. BSO cubes are designed for complex analytical requirements requiring formulas/ math, and frequent procedural calculations.	Data does not leave Autonomous AI Lakehouse, eliminating the need for refreshing and restructuring in Essbase. Since you create the federated partition over an existing ASO or BSO cube, you can use either of those Essbase options and benefit from its style of calculations and queries, without ever having to load the data into Essbase or restructure the outline. If your organization already has a fact table stored in Autonomous AI Lakehouse, a federated partition enables you to use Essbase functionality such as: • Querying tables using Smart View in Excel • Powerful calculation and query capabilities including financial and time intelligence • "What-if" modeling and forecasting • Writeback capabilities If your organization already uses Essbase, a federated partition enables you to access these benefits of storing data in Autonomous AI Lakehouse: • Elimination of data latency caused by Essbase data load processes • Ability to handle larger volumes of data than possible in Essbase • Other operational benefits of Autonomous AI Lakehouse, including auto-

Copyright © 2019, 2026, Oracle and/or its affiliates.

	Aggregate Storage (ASO)	Block Storage (BSO)	Federated Partition Cube
			scaling and automated backups

Restrictions for Federated Cubes

Some functionality is not supported for Essbase federated cubes.

Calculation Restrictions

- Block calculation mode (enabled when Essbase configuration setting CALCMODE is set to BLOCK) is not applicable for federated cubes. Calculation processing is pushed to Autonomous AI Lakehouse. If an exception exists and the calculation is processed on the Essbase Server instead, then solve order determines the dependency analysis.
- When performing custom allocations on a federated cube based on ASO, you can only override existing values. You cannot add to, nor subtract from, existing values.
- Aggregate storage custom calculations and allocations are supported for federated cubes using only the MDX Insert logic. All restrictions documented for MDX Insert also apply to custom calculations and allocations in a federated cube.
- If you need to run Essbase block storage (BSO) calculation scripts, select a dense dimension as the pivot dimension. Calculation scripts are not supported for federated cubes if the pivot dimension is sparse.
- The following calculation commands are not supported for federated cubes, and return an error if used:
  - CALC AVERAGE
  - CALC FIRST
  - CALC LAST
  - CCONV
  - DATAEXPORTCOND
  - DATAIMPORTBIN
  - SET AGGMISSG OFF (Essbase always consolidates `#MISSING` for federated cubes)
  - SET CLEARUPDATESTATUS
  - SET CREATEBLOCKONEQ OFF (Essbase calculation of sparse dimensions is always top-down for hybrid and federated cubes, resulting in calculation of upper-level parents. In other words, the default behavior is SET CREATEBLOCKONEQ ON for federated cubes as well as hybrid cubes.)
  - SET FRMLRTDYNAMIC
  - SET REMOTECALC
  - SET UPTOLOCAL
  - SET UPDATECALC ON (Intelligent calculation, with its markers for dirty/clean blocks, is applicable only to non-federated, block storage cubes)
  - THREADPARVAR

  For more about calculation support, see Calculate and Query Federated Cubes.

Copyright © 2019, 2026, Oracle and/or its affiliates.

- Calculation scripts using the @MDALLOCATE function are not supported and will fail with an error message.
- Some long running calculations using IF/ELSEIF/ELSE logic may fail on federated cubes, returning either or both of the following ORA error(s) from Autonomous AI Lakehouse:

  ```
  ORA-04036: PGA memory used by the instance or PDB exceeds
  PGA_AGGREGATE_LIMIT
  ```

  ```
  ORA-12805: parallel query server died unexpectedly
  ```

  If you encounter such errors, you may need to increase Autonomous AI Lakehouse hardware configuration to 16 CPU cores and 128G RAM. Refer to OCI documentation: Changing the Shape of an Instance.

Data Load Restrictions

- Before performing a data load through Essbase to the fact table, Oracle recommends you upload the data file to the Essbase Server. Client side data load is supported, but takes longer.
- If you do not need to load data through Essbase to Autonomous AI Lakehouse, you can use Data Tools in Autonomous AI Database to load data to the fact table and perform other management tasks. However, ensure that the cube outline and fact table do not get out of sync – see Metadata Precautions for Federated Partition Cubes.
- Loading Essbase-formatted data export files into federated cubes can be time consuming. To optimize data loads, use a DBMS formatted source file. You can make one using the DATAEXPORT calculation command with DataExportCSVFormat option. CSV formatted files can be loaded faster because they are in accordance with DBMS_CLOUD package format options for source files.
- The pivot dimension used in data load input files must be the same as the pivot dimension of the fact table.

  See Set up Credentials for Federated Cubes.
- Importing data from multiple files in parallel using a MaxL **import** statement with wildcard characters is not supported for federated cubes.
- Aggregate storage incremental data loads using buffers are not supported in a federated cube.

Feature Restrictions

- Block storage cubes must be in hybrid mode to support federated partitions. Do not configure ASODYNAMICAGGINBSO to any setting other than FULL for the application containing the federated partition, or else query results may be incorrect, and a warning message will be written to the log.
- Scenario management is not supported.
- Transparent or replicated partitions against the federated cube are not applicable/not supported.
- MaxL does not support creating or altering federated partitions, but you can use REST API.
- MaxL statements and APIs for clearing/resetting data, clearing data regions, or clearing aggregates are not supported.

Copyright © 2019, 2026, Oracle and/or its affiliates.

- Text lists (a.k.a smartlists) are not supported
- Currency cubes are not supported.
- Data audit trail is not supported.
- Triggers on cube events are not supported.

General Restrictions

- Federated partitions are not migrated with Essbase applications. When preparing to move your application and cube to another server or to migrate to another Essbase version, you need to delete the federated partition and recreate it in the new environment. Refer to Back up and Restore a Federated Partition Application.
- The cube must be within its own uniquely-named application. Federated cubes should not share an application with other cubes. Do not use the same Autonomous AI Lakehouse schema for multiple instances of Essbase.
- Exporting a federated cube to an application workbook is not supported (does not export the data nor the partition definition).
- Lifecycle Management (LCM) import operations (and Migration Utility import) are not supported for federated cubes. Federated partitions must be recreated manually on the target.
- For federated cubes based on ASO, Oracle recommends that dimensions containing multi-level stored member hierarchies not be selected as the pivot dimension. Select a pivot dimension with dynamic hierarchies, or a stored hierarchy that is a flat, single-level hierarchy (where all members are level 0 stored members).
- Oracle Database has a limit of 1,000 columns, and the pivot dimension inherits this limit. Determine the number of eligible column members in the pivot dimension to ensure that you do not encounter the limit. The number of potential stored member combinations in the pivot dimension plus the number of dimensions in the cube should be less than or equal to 1,000.
- A federated cube with Essbase managed data storage supports duplicate members, whereas a user managed federated cube does not.
- Varying attributes, and any default attribute calculation other than Sum are not supported.
- MDX Sub Select is not supported.
- Building aggregate views (MaxL statements **execute aggregate process**|**build**|**selection**) is not supported.
- Merging data regions/slices is not applicable (because data is in Autonomous AI Lakehouse).
- Information returned from the MaxL statement `query application APP-NAME list aggregate_storage storage_info` (or equivalent API) is not complete/accurate.
- Asymmetric queries may have slower performance.
- Writeback performance (for example, the speed of submitting data updates from Smart View) can be slow if there is a large amount of data to submit.
- Copying federated cubes or their applications is not supported.
- The following Essbase application or server configuration settings are ignored:
  - AUTOMERGE
  - AUTOMERGEMAXSLICENUMBER
  - DATACACHESIZE

Copyright © 2019, 2026, Oracle and/or its affiliates.

  - CALCCACHE
  - CALCCACHEDEFAULT
  - CALCCACHEHIGH
  - CALCCACHELOW
  - CALCLOCKBLOCK
  - CALCMODE
  - CALCNOTICE
  - CALCOPTFRMLBOTTOMUP
  - CALCREUSEDYNCALCBLOCKS
  - CALCPARALLEL
  - CALCTASKDIMS
  - DATACACHESIZE
  - DYNCALCCACHEBLKRELEASE
  - DYNCALCCACHEBLKTIMEOUT
  - DYNCALCCACHECOMPRBLKBUFSIZE
  - DYNCALCCACHEMAXSIZE
  - DYNCALCCACHEONLY
  - DYNCALCCACHEWAITFORBLK
  - ENABLE_DIAG_TRANSPARENT_PARTITION
  - EXPORTTHREADS
  - FORCEGRIDEXPANSION
  - GRIDEXPANSION
  - GRIDEXPANSIONMESSAGES
  - INDEXCACHESIZE
  - INPLACEDATAWRITE
  - PARCALCMULTIPLEBITMAPMEMOPT
  - SSAUDIT
  - SSAUDITR
  - SSLOGUNKNOWN
  - SUPNA
  - TARGETASOOPT
  - TARGETTIMESERIESOPT
- Creating a federated partition may fail with the following error if too many levels exist in the Essbase outline: `Remote warning from federated partition on Analytic View: [ORA-04063: hierarchy has errors]`.
- Federated partition creation may fail if characters or name lengths used in Essbase dimension names or member names in the pivot dimension are not supported or are considered special by Autonomous AI Lakehouse. These limitations should be considered

Copyright © 2019, 2026, Oracle and/or its affiliates.

in addition to the documented Essbase Naming Conventions for Dimensions, Members, and Aliases.

- You can only delete a federated partition from the Essbase web interface. You cannot delete it from Cube Designer.

Copyright © 2019, 2026, Oracle and/or its affiliates.

16
Configure Oracle Essbase

Oracle Essbase is preconfigured with properties that you may never need to modify.

You can add or modify Provider Services properties at the Essbase server level.

- Set Provider Services Configuration Properties

Set Provider Services Configuration Properties

If you have the Service Administrator role, you can customize network-related settings for Oracle Essbase using the Provider Services configuration properties.

To set the values for Provider Services configuration properties,

1. Log in to the Essbase web interface as a Service Administrator.
2. Click **Console**.
3. In the Console, click **Configuration**.
4. On the Provider Services tab, click **Add** to add a new property and set its value. If the property you want to configure is already listed, double-click the **Value** field to edit the value.
5. When you are finished editing properties, click **Save**.

See Provider Services Configuration.

Copyright © 2019, 2026, Oracle and/or its affiliates.

17

Download Utilities and Perform Tasks in the Console

In the Essbase web interface, on the **Console** page, as a user or service administrator, you can access various tools and tasks that you will need.

Topics:

- Access Tools and Tasks from the Console
- Enable Antivirus Scanning in Essbase

Access Tools and Tasks from the Console

Users and administrators have access to Console actions from the Essbase web interface. Note that the bold terms below represent the options listed on the Console.

As a non-service administrator user, you can:

- Download **desktop tools** that you will install locally and use for administration, import, and export. See Set Up Your Client.
- Monitor your own user **sessions**.
- View **database size statistics** for applications for which you're a provisioned user.

As a service administrator, you can:

- Download **desktop tools** that you install locally and use for administration, import, and export. See Set Up Your Client.
- Set platform-based **email configuration** for email notifications of scenario status changes. See Enable Email Notifications for Scenario Status Changes.
- Enable **file scanner** to scan files and ensure they're virus-free before they're uploaded into Essbase.
- Monitor and manage all user **sessions**.
- View **database size statistics** for all applications.
- View agent and server **configuration** and add Provider Services.
- Add substitution **variables** that apply to all Essbase applications. See Use Substitution Variables.
- Enable **Performance Analyzer** to capture incremental log data according to the interval you set in the Console. See About Performance Analyzer and System Performance Templates.

Enable Antivirus Scanning in Essbase

If your network uses an antivirus scanner, enable it in Essbase to ensure that files uploaded to Essbase are scanned for viruses.

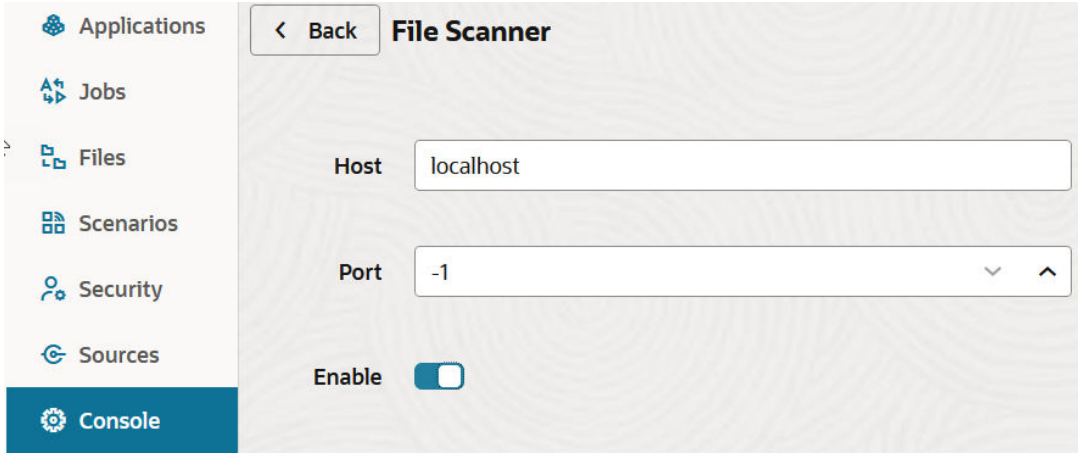
Copyright © 2019, 2026, Oracle and/or its affiliates.

Requirements:

- You must be the system administrator.
- The virus scanner software must be compatible with the ICAP protocol.
- Essbase supports Symantec and ClamAV virus scanner software. The Symantec and ClamAV virus scanners are not included with the Essbase installation. Install the software separately and confirm that it can be reached by the Essbase Server.

To enable virus scanning in Essbase,

1. Log in to the Essbase web interface.
2. Go to the Console.
3. Click **File Scanner**.
4. Enter the hostname and port for the virus scanner ICAP server.
5. Enable the virus scanner using the toggle switch.



If the file scanner detects a virus, a message stating "File is infected by a virus" is displayed, and you will be unable to upload the file.

ClamAV has a known limitation on file sizes. Consult ClamAV documentation for details.

Copyright © 2019, 2026, Oracle and/or its affiliates.

18

Use Logs and View Notifications

You can download and view logs at the applications level, and you can view notifications in the Essbase web interface. You can also use Performance Analyzer, which analyzes Essbase logs and provides usage and performance statistics.

- Download Application Logs
- About Performance Analyzer
- View and Manage Notifications in the Essbase Web Interface

Download Application Logs

If you have Application Manager permission or higher, you can download application logs using the Essbase web interface. You can download the latest log, as well as rolled over logs. You can also view logs without downloading them.

To download an application log,

1. On the Home page, select the application.
2. Navigate to the **Logs** tab:
   - In the Redwood Interface, on the General page, click the **Logs** tab in the upper right-hand corner.
   - In the Classic Web Interface, to the right of the application name, click the Actions menu, select **Inspect**, and click the **Logs** tab.
3. On the **Logs** tab, click the **Download** icon under **Latest**, the **View** icon under **Latest**, or the **Download** icon under **All.**
4. If you're downloading, save the file locally.

About Performance Analyzer

The Performance Analyzer, available in the Console of the Essbase web interface, helps you monitor usage and performance statistics of your Essbase service.

Performance Analyzer reads log files behind the scenes, scanning them at intervals that you specify. From the log files it creates .csv files of Essbase activity data. The data comes from the application ODL log, agent log, and WebLogic logs.

After a Performance Analyzer file grows to 10 MB, a new file is created. By default, Essbase keeps a total of 112 files, at which point Essbase deletes the oldest file first. The most recent file is called EssbaseHpa_Data.csv. The older files are named numerically; for example, EssbaseHpa_*n*_Data.csv.

A template in the Essbase file catalog, in **Files** > `gallery > System Performance > Health and Performance Analyzer`, can help you learn more about Performance Analyzer. To use the gallery template, you copy and paste CSV data into the template.

Copyright © 2019, 2026, Oracle and/or its affiliates.

Because each .csv file contains time-stamped information from your logs in chronological order, you can use a database or reporting utility of your choice to:

- combine .csv files or file portions to create performance analysis for precise time intervals.
- build charts or other visualizations of the data.

Enable Performance Analyzer and Choose Settings

If you are a Service Administrator, you can enable the Performance Analyzer in the Console of the Essbase web interface to capture information from log files about usage and performance.

You can also set the interval at which Essbase captures the CSV data, and indicate the maximum number of files you want Essbase to retain.

1. In the Essbase web interface, on the Home page, click **Console**.
2. Click **Performance Analyzer**.
3. Click **Settings**.
4. In the **Settings** dialog box, use the toggle switch to enable **Performance Analyzer**.
5. In the **Interval** field, choose the interval at which you want new .csv files to be created. The value can be between 2 and 100 minutes.
6. In the **Max files count** field, choose the maximum number of .csv files you want Essbase to retain. The value can be between 1 and 1000 files.

Understand and Work With Performance Analyzer Data

Performance Analyzer generates CSV data based on logs and organizes it into columns. First, you gather the CSV data and open the .csv files in Excel, and then you can examine and work with the data using Excel filtering tools.

To gather the CSV data:

1. Locate the .csv files you want to analyze.
   a. In the Essbase web interface, on the Home page, click **Console**.
   b. Select **Performance Analyzer**.
   c. Find the .csv file or files matching the time period you are interested in.
2. Download the files:
   a. Select the download icon under **Actions** to download each file.
   b. Repeat for additional files you want to download.

Open the files in Excel and examine the columns at the top of the files. Most of the columns are self-explanatory. They contain data helpful for filtering performance analysis, such as application and cube name, time stamp, and date.

Columns N and O need further discussion, as they contain key information. Column N contains information such as configuration settings, database settings, and user logins. Column O contains specific entries within those categories. In Excel, you can filter on column N and choose a category, and then filter on column O to choose specific entries within those categories.

Column N (Operation.OperationType) describes the type of the log message:

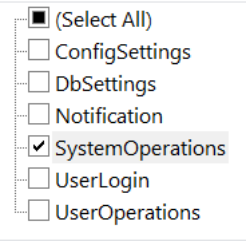
- **UserLogin** shows how long the user was active, and when the user logged out.

Copyright © 2019, 2026, Oracle and/or its affiliates.

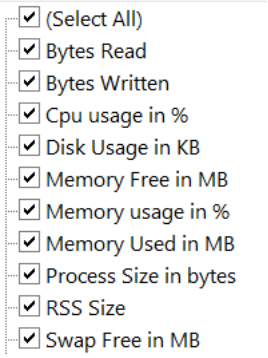
- **UserOperation** shows all user operations, such as data loads, calculations, and restructures. It also shows errors and exceptions.
- **SystemOperation** shows CPU, memory, disk, and I/O usage.
- **DBSettings** shows database statistics.
- **ConfigurationSetting** shows configuration settings.
- **Notification** identifies when there is a severe error.

If you filter on column N and then choose the specific category you are interested in, you can then view events within that category by filtering on column O.

Example view of a filter on column N:



Example view of a filter on column O:



Other Parameters

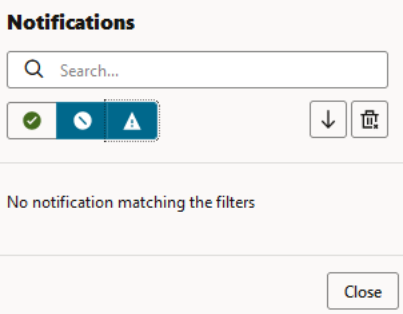
- **Measures.SumValue** is the value of the metric in OperationType.Metric.
- **Measures.LineCount** is the number of reported instances of the OperationType.Metric (currently, always 1).
- **Measures.AverageValue** is not used.

View and Manage Notifications in the Essbase Web Interface

There is a notifications panel in the Essbase web interface for success, warning, and error messages from your active session.

You can access the notifications by clicking the notifications icon in the upper right hand corner of your screen. These messages are stored in the browser session and remain available until your session has ended. After the session expires or you sign out, the messages are automatically cleared. You can filter by message type using the provided icons, and you can sort ascending or sort descending by timestamp using the arrow icon.

Copyright © 2019, 2026, Oracle and/or its affiliates.



Copyright © 2019, 2026, Oracle and/or its affiliates.

A

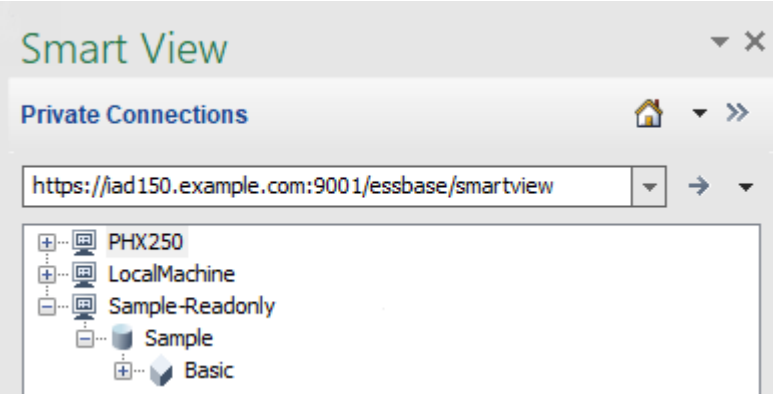
Centralized Smart View URL, and Read-Only Clusters

You can set up access to multiple Essbase Server nodes from the Smart View connection panel using one centralized URL. To provide high-availability and load balancing for cubes that are heavily used for query and reporting, you can create active-active (read-only) clusters of identical Essbase cubes.

> **Note**
>
> This feature is available only for independent deployments.

By default, only a single Essbase server node, usually named EssbaseCluster, is accessible from Smart View. To enable centralized URL access to more than one server node, you must perform some configuration steps.

In the following Smart View image,



- The centralized Smart View URL for this private connection is https://iad150.example.com:9001/essbase/smartview.
- Two Essbase servers, with aliases PHX250 and LocalMachine, are running on separate instances that an administrator configured to be accessible under a centralized Smart View URL.
- The node named Sample-Readonly is an active-active (read-only) cluster. A read-only cluster is not required for centralized Smart View URL access, but is an option available if you want to set up a cube that offers high availability without writeback.

To enable single URL access to multiple Essbase instances from Smart View, select a workflow, depending on your deployment type.

- If Essbase is configured with EPM Shared Services, see Access Multiple Essbase Servers in EPM Shared Services.

Copyright © 2019, 2026, Oracle and/or its affiliates.

- If Essbase is configured in the default WebLogic mode, see Access Multiple Essbase Servers Using a Centralized Smart View URL.

The workflows are mutually exclusive. If Essbase is configured with EPM Shared Services, only Essbase instances that are registered with EPM will be displayed in the centralized Smart View URL.

Access Multiple Essbase Servers Using a Centralized Smart View URL

You can configure a single point of end-user access from Smart View to multiple Essbase Server instances.

For independently deployed Essbase instances *not* registered with EPM Shared Services, you can use Provider Services to configure all the Essbase Servers to be accessible as nodes under one centralized Smart View URL.

After you do so, Smart View users will be able to access all the Essbase Servers using just one URL in their connection panel.

To set up the centralized URL access,

1. On your current Essbase Server machine, navigate to the location of the clone scripts.
   - Linux

     `<Essbase Product Home>/modules/oracle.essbase.sysman/scripts/copyclusterkey`
   - Windows

     `<Essbase Product Home>\modules\oracle.essbase.sysman\scripts\copyclusterkey`

   If you do not know where *<Essbase Product Home>* is in your environment, refer to Environment Locations in the Essbase Platform for an explanation.
2. Copy the **cloneTokenManagerKeys** (`.sh` or `.cmd`) script and the **updatedClusterId.py** file into the `bin` directory of *<Domain Home>* on your current Essbase Server machine. If your Essbase Server is on Windows, you must also copy `cloneTokenManagerKeys.ps1`. If you do not know where *<Domain Home>* is in your environment, refer to Environment Locations in the Essbase Platform for an explanation.

   For Linux,

   a. Copy `cloneTokenManagerKeys.sh` and `updatedClusterId.py` into `$DOMAIN_HOME/bin`. For example:

      ```
      /scratch/<home dir>/Oracle/Middleware/Oracle_Home/domains/essbase_domain/bin
      ```

   b. Open a command prompt in the `$DOMAIN_HOME/bin` directory, and grant execute permission to `cloneTokenManagerKeys.sh`. For example,

      ```
      chmod +x cloneTokenManagerKeys.sh
      ```

   c. Run the script, providing a secondary Admin Server URL to synchronize it (for single sign-on using Provider Services) with the current server.

Copyright © 2019, 2026, Oracle and/or its affiliates.

The syntax is:

```
./cloneTokenManagerKeys.sh t3://<ADMIN-SERVER-NAME>:<ADMIN-PORT>
```

For example:

```
./cloneTokenManagerKeys.sh t3://AdminServer2:7001
```

If there are multiple environments to synchronize, enter the Admin Server URLs for each environment, delimited by spaces. For example:

```
./cloneTokenManagerKeys.sh t3://AdminServer2:7001 t3://AdminServer3:7001
```

If TLS (SSL) is enabled, use the `t3s` protocol to specify the URL. For example:

```
./cloneTokenManagerKeys.sh t3s://AdminServer2:7002
```

For Windows,

a. Copy `cloneTokenManagerKeys.cmd`, `cloneTokenManagerKeys.ps1`, and `updatedClusterId.py` into *%DOMAIN_HOME%*`\bin`. For example:

```
C:\Oracle\Middleware\Oracle_Home\domains\essbase_domain\bin
```

b. Open a command prompt in the *%DOMAIN_HOME%*`\bin` directory.

c. Run the script, providing a secondary Admin Server URL to synchronize it (for single sign-on) with the current server using Provider Services.

The syntax is:

```
.\cloneTokenManagerKeys.cmd t3://<ADMIN-SERVER-NAME>:<ADMIN-PORT>
```

For example:

```
.\cloneTokenManagerKeys.cmd t3://AdminServer2:7001
```

If there are multiple environments to synchronize, enter the Admin Server URLs for each environment, delimited by spaces. For example:

```
.\cloneTokenManagerKeys.cmd t3://AdminServer2:7001 t3://
AdminServer3:7001
```

If TLS (SSL) is enabled, use the `t3s` protocol to specify the URL. For example:

```
.\cloneTokenManagerKeys.cmd t3s://AdminServer2:7002
```

3. After you are finished with the synchronizing process, restart all Essbase Servers that you synchronized with the current Essbase Server. Refer to Start, Stop, and Check Servers.

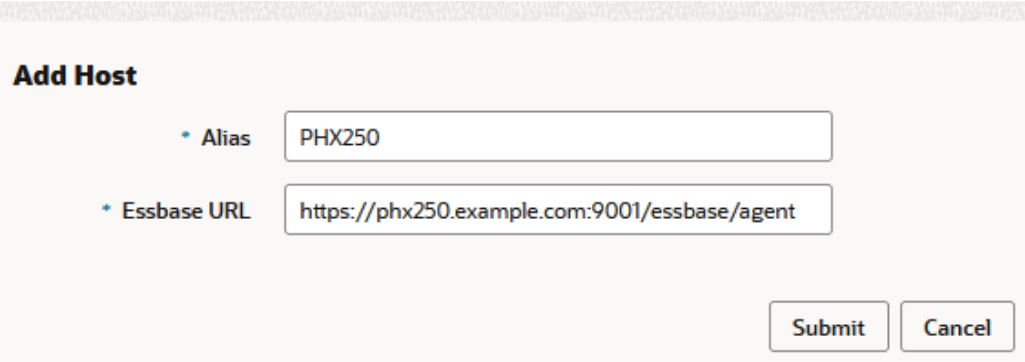
4. Configure your Essbase Servers by adding them to Provider Services management using the Essbase web interface.

   a. In the Essbase web interface, navigate to the **Console** and click **Configuration**.

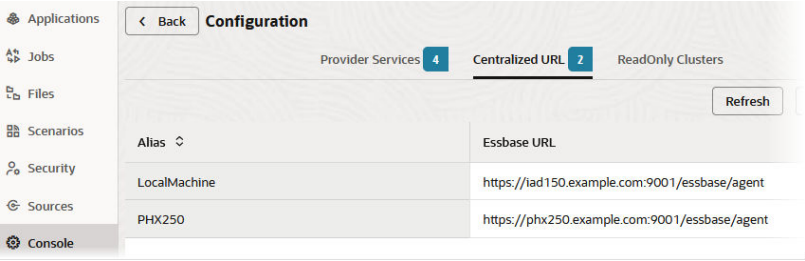
Copyright © 2019, 2026, Oracle and/or its affiliates.

b. Go to the **Centralized URL** tab and click **Add**.

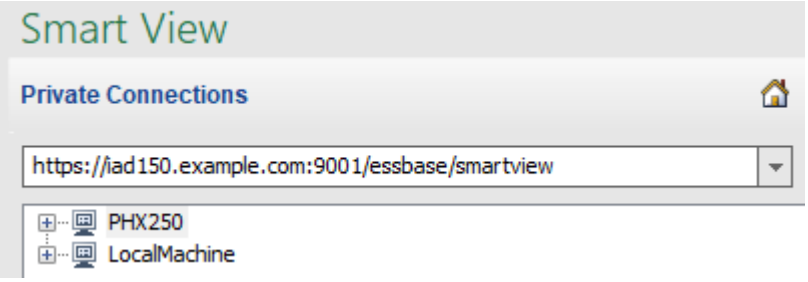
c. In the **Add Host** dialog, enter information about one of the Essbase Servers. Provide an Alias and an Agent URL.



d. Click **Submit**, and click **Add** again to add more Essbase Servers that you want to make accessible from a single Smart View URL.



e. From Smart View, log in to the Essbase Server you just configured. You should be able to connect to all instances that you configured for centralized URL.



f. If you additionally want to set up high availability (failover) access to an application hosted on one or more of the Essbase Servers you configured using a centralized URL, proceed to Configure and Manage Active-Active (Read-Only) Essbase Clusters.

Configure and Manage Active-Active (Read-Only) Essbase Clusters

To provide high-availability and load balancing for cubes that are heavily used for query and reporting, you can create active-active (read-only) clusters of identical Essbase cubes.

Copyright © 2019, 2026, Oracle and/or its affiliates.

This configuration is available for independent deployments of Essbase. The applications and cubes in the cluster may be hosted on a single Essbase Server, or they maybe hosted on more than one Essbase Server.

Whether the cluster is on a single Essbase Server or spans multiple servers, Smart View users can access the cluster by connecting to just one centralized URL.

The benefit of an active-active (read-only) cluster is to provide high-availability and load balancing for cubes that are heavily used for query and reporting, but do not need to be updated frequently. A cluster enables client requests to be distributed amongst the cubes replicas in the cluster. Clusters support only read operations. You cannot update data or modify the outlines.

Configure a Read-only Cluster

To set up an active-active (read-only) cluster,

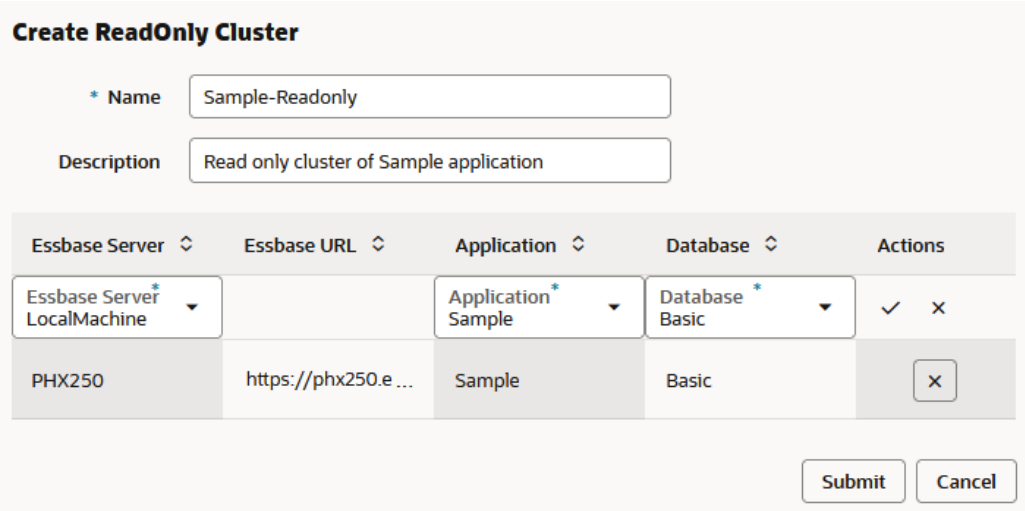
1. If the cluster needs to include applications hosted on more than one Essbase Server, complete steps 1-3 in Access Multiple Essbase Servers Using a Centralized Smart View URL.

   > **Note**
   >
   > The applications within the read-only cluster must all be on the same version of Essbase.
   >
   > All nodes in the read-only cluster must maintain the same application and cube names, to ensure successful connection and availability for Ad Hoc Analysis.

2. Configure your read-only cluster using the Essbase web interface.
   a. On the Home page, navigate to the **Console** and click **Configuration**.

      Go to the **ReadOnly Clusters** tab and click **Create**.
   b. Enter a cluster name; for example, **Sample-Readonly**.
   c. Optionally enter a description; for example, **Read only cluster of Sample application**.
   d. Under **Essbase Server**, select either **LocalMachine** or any other Essbase Server available in the list (for which you already configured centralized URL access).
   e. Under **Application**, select the application containing the cube for which you are configuring this cluster.
   f. Under **Database**, select the cube for which you are configuring this cluster.
   g. Optional: Under **Actions**, click the check mark to add another cube to the cluster. Repeat steps e - g.

Copyright © 2019, 2026, Oracle and/or its affiliates.



h. Click **Submit** to finish the cluster definition.

Manage a Read-only Cluster

To manage an existing active-active (read-only) cluster,

1. In the Essbase web interface, on the Home page, navigate to the **Console** and click **Configuration**.
2. Go to the **ReadOnly Clusters** tab.
3. Under **Actions**, select **Manage**, **Edit**, or **Delete**.
   - Select **Manage** to view status of cubes in the cluster, or to toggle their availability state on or off.
   - Select **Delete** to remove a cluster definition.
   - Select **Edit** to update which cubes are included in the cluster definition.

Access Multiple Essbase Servers in EPM Shared Services

Using EPM Shared Services, you can configure a single point of end-user access from Smart View to multiple Essbase Server instances.

For independently deployed Essbase instances which are registered with EPM Shared Services for user authentication and role assignments, you can make all the Essbase Servers accessible as nodes under one centralized Smart View URL.

After you do so, Smart View users will be able to access all the Essbase Servers using just one URL in their connection panel.

To set up the centralized URL access,

1. Register multiple Essbase servers with EPM Shared Services, and optionally with EAS Lite, using the instructions at: Manage Multiple Essbase 21c Servers in Shared Services and Administration Services
2. Connect to Smart View, as described in Analyze an Application in Smart View. All registered Essbase servers should be listed in the connections panel.
3. If you want to set up active-active/read-only clusters of one cube, see Configure and Manage Active-Active (Read-Only) Essbase Clusters.

Copyright © 2019, 2026, Oracle and/or its affiliates.