Oracle® Essbase

Essbase Cube Designer

G38725-01
December 2025

Oracle Essbase Essbase Cube Designer,

G38725-01

Primary Author: Essbase Information Development Team

Contents

1

Set up Cube Designer

You might find it easier to work with application workbooks in Excel using the Cube Designer extension for Smart View.

Workflow to Set up Cube Designer

This is the workflow for setting the Smart View Cube Designer extension:

1. Install Smart View.
2. Set up a data source connection to Essbase.
3. Install Cube Designer Smart View extension.
4. Update Cube Designer Smart View extension.

Download and Run the Smart View Installer

Using Smart View, you can view and manipulate Essbase data in Microsoft Excel.

Smart View Prerequisites

- For Smart View version support and supported versions of Windows operating system, .NET, and Microsoft Office, see the Smart View readme, at Oracle Applications Enterprise Performance Management
- On the Oracle Technical Resources **Downloads** page, the latest release for Smart View is always certified.

Installing Smart View

1. Log into Essbase.
2. Click **Console.**
3. Navigate to the Smart View download page:
    - In the Redwood Interface, select **Desktop Tools**, expand **Smart View**, and click the Browse icon on the **Smart View for Essbase** tile.
    - In the Classic Web Interface, on the **Desktop Tools** tab, click the Browse icon to the right of **Smart View for Essbase.**
4. Click **Download Smart View for Office**.

5. Click **Download Now**.
6. From the **Platforms** drop-down menu, select your platform.
7. Check the box, **I reviewed and accept the Oracle License Agreement**, and then click **Download**.

   If the Oracle sign-in page is displayed, then sign in with your Oracle user name (usually your email address) and password.
8. Follow the steps for your browser to download the `.zip` file, and save it to a folder on your computer.
9. Go to the folder that you used in Step 8, and then double click the `.exe` file to start the installation wizard.
10. Select a destination folder for Smart View, and then click **OK.** For new installations, Smart View is installed by default in: `C:\Oracle\smartview`.

    If you are upgrading an installation of Smart View, then the installer defaults to the folder where you previously installed Smart View.
11. When the installation is complete, click **OK.**

Continue the setup process with Connect to Essbase.

Connect to Essbase

After you install Smart View, you can create connections to Essbase.

Connections require information about the server and port. Your Essbase administrator should provide you with the information you need to create the connection.

See Connect to a Cube in Smart View.

Continue the setup process with Install the Smart View Cube Designer Extension.

Install the Smart View Cube Designer Extension

You can install Cube Designer either from Smart View or from the Console in the Essbase web interface.

Before you perform this procedure, you must complete the steps in Connect to Essbase.

Install Cube Designer from Smart View

1. On the Smart View ribbon, select **Options**, and then **Extensions**.
2. Click the **Check for updates** link.

   Smart View checks for all extensions that your administrator has made available to you.
3. Locate the extension named **Oracle Cube Designer** and click **Install** to start the installer.
4. Follow the prompts to install the extension.

> **Note**
>
> - This method is for installing or updating an existing Cube Designer installation. To downgrade, uninstall and then install again.
> - Oracle recommends that you install Cube Designer by downloading from the Essbase web interface and running the `.svext` file.

Install Cube Designer from Essbase web interface

1. In the Essbase web interface, on the Home page, click **Console.**
2. Navigate to the Cube Designer download option:
   - In the Redwood Interface, select **Desktop Tools**, expand **Smart View**, and click the Download icon on the **Cube Designer Extension** tile.
   - In the Classic Web Interface, on the Desktop Tools tab, to the right of **Cube Designer Extension**, click Download .
3. Follow the steps for your browser to download the Cube Designer installer, and save it to a folder on your computer.
4. Close all Microsoft Office applications and make sure Microsoft Office applications are not running in the background.
5. Double click the installation file.
6. Restart Microsoft Office applications.

Connect to Essbase from Cube Designer

1. Create a private connection to the Essbase Server from Smart View.
   After you do this, the private connection will be available in the **Connections** dialog box.
2. On the Cube Designer ribbon, click **Connections** 
3. In the **Connections** dialog box, select your Essbase URL and click **Save**.
   This saves the Essbase URL as your default Essbase connection. To switch to a different Essbase instance, repeat the steps, using the new URL.

Set WebView2 to Enable for UI from Smart View

Cube Designer works correctly only when the WebView2 Runtime is either enabled for UI or completely disabled. To make the necessary changes to the WebView2 Runtime settings in Smart View:

1. Open Excel and go to the Smart View ribbon.
2. Click the **Help** dropdown menu and select **Health Check**.
3. Under **Smart View Options**, locate **WebView2 Runtime**.
4. In the **Tab** section, ensure that **WebView2** is set to **Enable for UI**, or use the **Fix/Update/Info** dropdown menu next to **WebView2** and select **Enable for UI**.
5. Click **Update Settings**, then restart Excel.

Update the Smart View Cube Designer Extension

If an extension is available for you to update, you can update it from Smart View Excel, on the **Extensions** tab of the Options dialog box.

To check for Cube Designer Smart View extension updates and install them:

1. From the Smart View ribbon, select **Options** and then **Extensions.**
2. Click the **Check for Updates, New Installs, and Uninstalls** link to check for updates.

   You are prompted to log in.

   If an update is available, the **Update Available** icon is displayed in the **Cube Designer** row.

   > **Note**
   >
   > This process uses a server locations list, which was created by previous Smart View connections . If there are connection definitions that are no longer valid, you receive errors when the process tries to connect to those servers. See Delete Smart View Connection URLs.

3. Click **Remove** to uninstall the extension.
4. Close Excel.
5. Restart Excel.
6. From the Smart View ribbon, select **Options** and then **Extensions**.
7. Click **Check for Updates, New Installs, and Uninstalls.**

   You are prompted to log in.

8. In the Cube Designer row, click **Install.** 

.
9. Close Excel.
10. Open Excel.
11. Ensure that the Cube Designer ribbon is displayed in Excel.

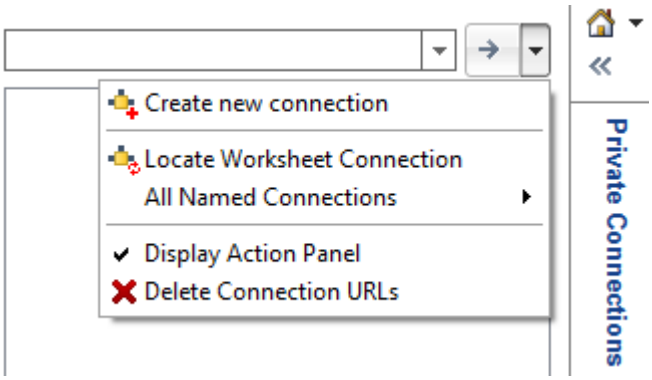
Delete Smart View Connection URLs

You can reset the list of connection definitions to remove those that are unwanted or invalid in Cube Designer.

When you connect to Essbase from Cube Designer, the list of server locations is created by previous Smart View connections . If there are connections that are no longer valid, you receive errors.

To reset the list of server locations,

1. Click the down arrow next to the **Private Connection** drop down list and select **Delete Connection URLs**.



2. In the Delete Connection URLs dialog box, select **Extension Update URLs** from the drop down menu.
3. Select all of the URLs except the one you want to use, and click **Delete.**

2

Top Tasks for Oracle Essbase

Learn the most common administrative tasks in Essbase. Download sample application workbooks and use them to build cubes, provision users, and connect to Smart View for data analysis. Edit the cube outline by adding members. Run a calculation, export data, and explore cube designer.

Before you start working through these tasks, be sure that you can log in to Essbase and that Smart View and cube designer are installed on client computers. See Set up Cube Designer.

- Download the Sample Dynamic Application Workbook and Examine Its Structure
- Create an Application in the Essbase Web Interface and Provision a User to Access and Query the Cube
- Analyze an Application in Smart View
- Modify an Essbase Outline
- Analyze Forecast Data in Smart View
- Create an Application and Cube in Cube Designer
- Analyze Data and Perform an Incremental Update in Cube Designer
- Transform Tabular Data into a Cube
- Export and Modify Metadata and Data

Download the Sample Dynamic Application Workbook and Examine Its Structure

In the Block Storage Sample (Dynamic) application workbook, all non-leaf level members in the cube are dynamically calculated. Dynamically calculated values are not stored in the cube; the values are recalculated and rendered for each user retrieval.

Now you'll download the application workbook from the Files catalog in Essbase, save it to a local drive, and examine its structure.

Download the Sample Dynamic Application Workbook

To download the Block Storage Sample (Dynamic) application workbook:

1. On the Home page, click **Files**, then click `Gallery > Applications > Demo Samples > Block Storage`.
2. On the Block Storage tab, click the Actions menu next to **Sample_Dynamic_Basic.xlsx.**
3. Save the application workbook file, `Sample_Dynamic_Basic.xlsx`, to a local drive.

Examine the Structure of the Sample Dynamic Application Workbook

Application workbooks contain a number of worksheets that define the metadata for the cube, including an Essbase.Cube worksheet that names all of the dimensions in the cube and defines other information about them, separate worksheets for each dimension, and a data worksheet.

1. In Microsoft Excel, open `Sample_Basic_Dynamic.xlsx`.
2. On the Essbase.Cube worksheet, the application name (Sample_Dynamic), cube name (Basic), the names of 10 dimensions, and other information about the dimensions, are defined.

	A	B	C	D	E
1	Application Name	Sample_Dynamic			
2	Database Name	Basic			
3	Version	1.0			
4					
5	**Dimension Definitions**				
6					
7		Dimension Type	Storage Type	Outline Order	Base Dimension
8	Year	Time	Dense	1	
9	Measure	Accounts	Dense	2	
10	Product	Regular	Sparse	3	
11	Market	Regular	Sparse	4	
12	Plan	Regular	Dense	5	
13	Caffeinated	Attribute-Boolean		6	Product
14	Ounces	Attribute-Numeric		7	Product
15	Pkg Type	Attribute-Text		8	Product
16	Population	Attribute-Numeric		9	Market
17	Intro Date	Attribute-Date		10	Product

3. Each dimension has a separate worksheet, Dim.*dimname*, in which the dimension is further defined with information such as the build method and incremental mode. Because the build method for each dimension in this sample application workbook is PARENT-CHILD, members are defined in PARENT and CHILD columns.

   On the Dim.Year worksheet, months roll up to quarters, and quarters roll up to years. For example, child members Jan, Feb, Mar roll up to parent member Qtr1. Child member Qtr1 rolls up to parent member Year.

	A	B	C
1	Dimension Name	Year	
2			
3	**Definitions**		
4	File Name	Dim_Year	
5	Rule Name	Dim_Year	
6	Build Method	PARENT-CHILD	
7	Incremental Mode	Merge	
8			
9	**Members**		
10	Columns	PARENT	CHILD
11			Year
12		Year	Qtr1
13		Qtr1	Jan
14		Qtr1	Feb
15		Qtr1	Mar

The Dim.Product and Dim.Market worksheets are similarly structured. In Dim.Product, SKUs roll up to product families, and product families roll up to Product. For example, child members 100-10, 100-20, and 100-30 (SKUs) roll up to parent member 100 (product family). Child member 100 rolls up to parent member Product.

	A	B	C
1	Dimension Name	Product	
2			
3	**Definitions**		
4	File Name	Dim_Product	
5	Rule Name	Dim_Product	
6	Build Method	PARENT-CHILD	
7	Incremental Mode	Merge	
8			
9	**Members**		
10	Columns	PARENT	CHILD
11			Product
12		Product	100
13		100	100-10
14		100	100-20
15		100	100-30

4. This sample application workbook includes data. Scroll to the last worksheet, Data.Basic, to review the structure of the columns and the data.

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	**Definitions**												
2	File Name	Cube_Basic											
3	Rule Name	Basic											
4	Data Load Option	Add											
5													
6	**Data**												
7	Columns	Dimensio	Dimensio	Dimensio	Dimensio	Measure.S	Measure.(	Measure.I	Measure.I	Measure.I	Measure.(	Measure./	Measure.E
8		100-10	New York	Jan	Actual	678	271	94	51	0	2101	644	2067
9		100-10	New York	Jan	Budget	640	260	80	40	#Missing	2030	600	1990
10		100-10	New York	Feb	Actual	645	258	90	51	1	2067	619	2041
11		100-10	New York	Feb	Budget	610	240	80	40	#Missing	1990	600	1980
12		100-10	New York	Mar	Actual	675	270	94	51	1	2041	742	2108
13		100-10	New York	Mar	Budget	640	250	80	40	#Missing	1980	700	2040

In this topic, you learned how to download an application workbook and examine its structure. Next, learn how to access additional templates using the Gallery section of the File Catalog.

Create an Application in the Essbase Web Interface and Provision a User to Access and Query the Cube

In Download the Sample Dynamic Application Workbook and Examine Its Structure, you learned about the structure of an application workbook by exploring `Sample_Basic_Dynamic.xlsx`.

Now, you use this workbook to learn how to create an application in the Essbase web interface and provision a user to access and query the cube.

Create an Application in the Essbase web interface

1. In the Essbase web interface, on the Home page, click **Import**.
2. On the Import dialog box, click **File Browser** (as the workbook was downloaded to the local file system). Open the Block Storage Sample (Dynamic) application workbook, `Sample_Basic_Dynamic.xlsx`, that you saved in Download the Sample Dynamic Application Workbook and Examine Its Structure.
3. In **Build Option**, select **Create Database**, and then check the box to load data. You do not need to select **Execute Scripts**, because all measures and aggregations along hierarchies in the cube are dynamically calculated at query time.

> **Note**
>
> In the Classic Web Interface, expand **Advanced Options** and **Build Option**, and then select to create a database and load data.

4. Click **OK**. In a few moments, the Sample_Dynamic application and Basic cube are created.
5. Open the outline:
   In the Redwood Interface,
   a. On the Home page, navigate to the **Sample_Dynamic** application, and select the Actions menu ⋯.
   b. Choose **Databases** > **Basic** > **Launch Outline**.

   The outline is a representation of the dimensions in the Basic cube as defined in the application workbook. The outline opens in a separate tab in the application window, allowing you to navigate between the outline and other web interface actions.

   In the Classic Web Interface,
   a. On the Home page, expand the **Sample_Dynamic** application, and select the cube, **Basic**.
   b. In the Actions list for the cube, select **Outline**.

   The outline is a representation of the dimensions in the Basic cube as defined in the application workbook. The outline opens in a separate browser tab, allowing you to navigate between the outline and other web interface actions.
6. View a cube dimension, and then drill down into the children of that dimension:
   a. Expand the **Year** dimension to view the quarters.
   b. Expand the individual quarters to view months.

Now all of the information from the application workbook is represented in the new cube.

Provision a User to Access and Query the Cube

1. Log in as a power user. This allows you to provision other users to the applications you have created.
2. Go to **Permissions**.
   In the Redwood Interface:
   a. On the Home page, select the **Sample_Dynamic** application.

b. Click **Permissions** and click **Add**.

> **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.

c. Click the + next to each user to assign their access.

d. Click **Close** to close the list of users in the right hand panel.

In the Classic Web Interface:

a. Return to the Essbase web interface browser tab and go to **Applications**.

b. Select the application for which you want to provision the user; in this example, select **Sample_Dynamic**. If you select the cube instead of the application, then you won't be able to provision user roles.

c. Use the Actions menu to open the application inspector.

d. Select the **Permissions** tab within the application inspector.

e. Click + to list users on the system, and click the + next to each user to assign their access.

3. Use the option controls next to each user to assign their access. Select **Database Manager** for each added user. The Database Manager has full control of the cube, but no control over the application.

In Analyze an Application in Smart View, you'll go to Smart View, log in as the user you just provisioned, and then query a cube.

Analyze an Application in Smart View

In Create an Application in the Essbase Web Interface and Provision a User to Access and Query the Cube, you created an application and a cube with data, and provisioned users.

Now you'll learn how to connect to the cube from Smart View, and perform ad hoc analysis by zooming and pivoting data.

This task assumes that you installed Smart View. See Download and Run the Smart View Installer.

Connect to the Cube from Smart View

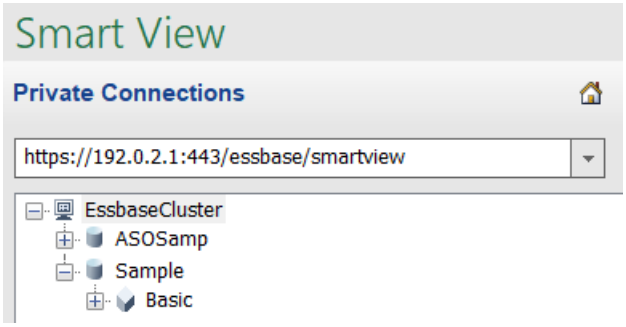
1. Open Microsoft Excel.

   If Smart View is installed, the Smart View ribbon is displayed in Excel.

2. On the Smart View ribbon, click **Panel.**

3. On the Smart View Home dialog box, click the arrow next to the **Home** button, then select **Private Connections.**

4. Make a private connection using the same URL that you used to connect to Essbase, and append `/essbase/smartview` to the end of that URL. For example, `https://192.0.2.1:443/essbase/smartview`.
5. Log in as the user you created.
6. Expand EssbaseCluster.



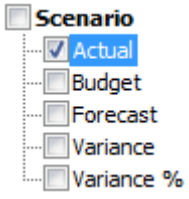
7. Highlight the Basic cube, and click **Connect**.

Perform an Ad hoc Analysis

1. On the **EssbaseCluster** tree, under **Sample_Dynamic,** select the **Basic** cube, then click **Ad hoc analysis.**
2. In the resulting grid, there is one aggregated data value for all five dimensions of this dynamic cube.

	Product	Market	Scenario
	Measures		
Year	105522		

3. Navigate into the member Scenario and narrow it down to a specific scenario type of Actual data.
   a. Click the cell containing Scenario.
   b. On the Essbase ribbon, click **Member Selection.**
   c. In the Member Selection dialog box, check the box next to the Actual member.



   d. Click **Add** to move Actual to the right pane.
   e. If Scenario is already included in the right pane, highlight it and use the left arrow to remove it, and then click **OK.**

On the Essbase ribbon, click **Refresh.** The grid should now have Scenario narrowed down to Actual.

	Product	Market	Actual
	Measures		
Year	105522		

4. Navigate into Measures and narrow it down to the Sales member, to analyze sales data.
   a. Highlight the cell containing Measures.
   b. On the Essbase ribbon, click **Zoom In.**
   c. Highlight the cell containing Profit, and click **Zoom In.**.
   d. Highlight the cell containing Margin, and click **Zoom In.**
   e. Highlight the cell containing Sales, and click **Keep Only.**

   The grid should now look like this:

	Product	Market	Actual
	Sales		
Year	400855		

5. Zoom in to Year by double-clicking the cell containing Year.

   The grid should now look like this:

	Product	Market	Actual
	Sales		
Qtr1	95820		
Qtr2	101679		
Qtr3	105215		
Qtr4	98141		
Year	400855		

6. Zoom in to Product by double-clicking the cell containing Product.

   The grid should now look like this:

		Market	Actual
		Sales	
Colas	Qtr1	25048	
Colas	Qtr2	27187	
Colas	Qtr3	28544	
Colas	Qtr4	25355	
Colas	Year	106134	
Root Beer	Qtr1	26627	
Root Beer	Qtr2	27401	
Root Beer	Qtr3	27942	
Root Beer	Qtr4	27116	
Root Beer	Year	109086	
Cream Soda	Qtr1	23997	
Cream Soda	Qtr2	25736	
Cream Soda	Qtr3	26650	
Cream Soda	Qtr4	25022	
Cream Soda	Year	101405	
Fruit Soda	Qtr1	20148	
Fruit Soda	Qtr2	21355	
Fruit Soda	Qtr3	22079	
Fruit Soda	Qtr4	20648	
Fruit Soda	Year	84230	
Water Beve	Qtr1	#Missing	
Water Beve	Qtr2	#Missing	

7. Enhance your data display to show time periods per product. Pivot Qtr1 of Colas by highlighting it, right-clicking and holding, then dragging it from B3 to C3.

   The grid should now look like this:

	Market	Actual			
	Sales	Sales	Sales	Sales	Sales
	Qtr1	Qtr2	Qtr3	Qtr4	Year
Colas	25048	27187	28544	25355	106134
Root Beer	26627	27401	27942	27116	109086
Cream Soda	23997	25736	26650	25022	101405
Fruit Soda	20148	21355	22079	20648	84230
Water Beve	#Missing	#Missing	#Missing	#Missing	#Missing
Product	95820	101679	105215	98141	400855

8. Analyze each product by region. Double-click Market in B1.

   The grid should now look like this:

		Actual				
		Sales	Sales	Sales	Sales	Sales
		Qtr1	Qtr2	Qtr3	Qtr4	Year
East	Colas	6292	7230	7770	6448	27740
East	Root Be	5726	5902	5863	6181	23672
East	Cream S	4868	5327	5142	4904	20241
East	Fruit So	3735	3990	4201	3819	15745
East	Water E	#Missing	#Missing	#Missing	#Missing	#Missing
East	Product	20621	22449	22976	21352	87398
West	Colas	6950	7178	7423	6755	28306
West	Root Be	8278	8524	8885	8513	34200
West	Cream S	8043	8982	9616	8750	35391

9. Drill in to a region to view product sales by state. Double-click East in A4.

   Because not every product is sold in every state, some cells have the #Missing label instead of a data value.

You can specify the ancestor position for a hierarchy as top or bottom, in the Member Options tab of the Options dialog in Smart View. SSANCESTORONTOP must first be enabled by an administrator in application configuration in the Essbase web interface. The grid changes shape when you perform a zoom-in operation.

In this task, you navigated through a data grid easily, zooming in and pivoting by clicking in the grid itself. You can also use the tools on the Essbase ribbon to perform the same actions. For more help on using Smart View, click the Smart View tab, and then click **Help**.

In Modify an Essbase Outline, you'll go back to the web interface and modify an outline.

Modify an Essbase Outline

In Analyze an Application in Smart View, you analyzed an application in Smart View.

In this example of modifying an Essbase outline, you will create a new forecast member, seed it with data, and execute a calculation script.

Create a New Member

You start by creating a new member.

- Redwood
- Classic

Redwood

1. On the Home page, navigate to the **Sample_Dynamic** application, and select the Actions menu ⋯.
2. Choose **Databases** > **Basic** > **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. Expand the **Scenario** dimension.
6. Select the **Budget** member.
7. On the outline toolbar, under the Add member icon , select **Add sibling member below**.
8. Enter the member name, **Forecast**, and press **Add**, and close the add member slider.
9. Select the new **Forecast** member and select the tilde (~) consolidation operator from the Consolidation list on the outline toolbar.
10. Click **Validate** .
11. Click **Save** .
12. Choose a restructure database option, and click **Save**.

Classic

1. In the Essbase web interface, on the Applications page, select the **Basic** cube in the **Sample_Dynamic** application.
2. Click the **Actions menu**, and select **Outline**.
3. Click **Edit**.
4. Expand the Scenario dimension by clicking the arrow next to **Scenario**.
5. Insert a member:
   a. Click **Edit** to put the outline in edit mode.
   b. Expand the **Scenario** dimension.
   c. Select the **Budget** member.
   d. On the outline toolbar, under **Actions**, select **Add a sibling below the selected member**.
6. Enter the member name, **Forecast**, and press **Tab**.
7. Select the tilde (~) consolidation operator from the list.

   The Forecast member does not aggregate with the other members in its dimension.
8. Leave the data storage type as **Store Data** because we want users to be able to input forecast data.
9. Click **Save**.

Seed the Forecast Member with Data

To seed the Forecast member with data, we'll create a calculation script and calculate forecast data.

- Redwood
- Classic

Redwood

1. In the Essbase web interface, on the Home page, open the **Sample_Dynamic** application, and open the **Basic** database (cube).
2. Click **Scripts**, click **Calculation Scripts**, and click **Create**.
3. In the **New Calculation Script** field, enter `salesfcst`.
4. In the script content box, enter a simple formula:

   ```
   Forecast(Sales=Sales->Actual*1.03;)
   ```

   Forecast for sales is equal to actual sales multiplied by 1.03, which seeds the Forecast member for Sales with a value 3% higher than the actual sales.
5. Validate the script.
6. Save the script and close it.

Classic

1. In the Essbase web interface, on the Applications page, select **Basic** cube in the **Sample_Dynamic** application, click the **Actions menu**, and select **Inspect**.
2. In the Basic dialog box, select the **Scripts** tab, with **Calculation Scripts** selected, click + to add a calculation script.
3. In the **Script Name** field, enter `salesfcst`.
4. In the **Script Content** box, enter a simple formula:

   ```
   Forecast(Sales=Sales->Actual*1.03;)
   ```

   Forecast for sales is equal to actual sales multiplied by 1.03, which seeds the Forecast member for Sales with a value 3% higher than the actual sales.
5. Click **Save and Close**.
6. Close the database inspector by clicking **Close** until all tabs are closed.

Execute the Script

Calculation scripts are executed as jobs.

1. In the Essbase web interface, from the Home page, click Jobs.
2. From the **New Job** drop-down menu, select **Run Calculation**.
3. On the Run Calculation dialog box, in the **Application** field, select **Sample_Dynamic** application.

   Notice that the **Database** field automatically populates the **Basic** cube.
4. On the **Scripts** menu, select the **salesfcst** calculation script that you created.
5. Click **Submit**.
6. Click **Refresh** to see that the job completes.

In Analyze Forecast Data in Smart View, you'll analyze this new forecast data in Excel. But first, let's take a closer look at managing jobs.

Analyze Forecast Data in Smart View

In Analyze an Application in Smart View, you learned to analyze data in Smart View. In Modify an Essbase Outline, you added a Forecast member to the outline, and seeded it with data.

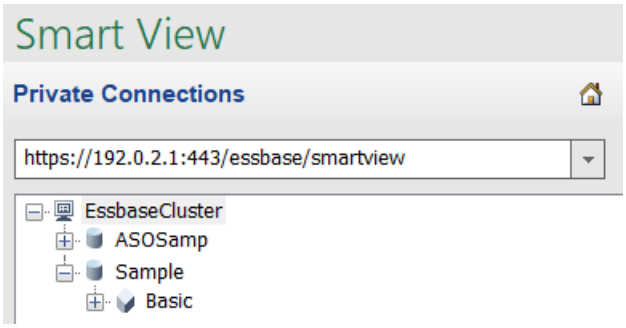
In this example of analyzing forecast data in Smart View, you'll reconnect to the cube, do analysis of the data, create a grid in Excel, and do an ad hoc analysis. Then you'll test that the calculation is correct, revise the grid, and submit monthly forecast values.

Now you'll reconnect to the cube in Smart View, and do further analysis of the data.

1. Open Excel and create a worksheet like the following one, by typing the member names in these cells: A3=Market, B3=Product, C1=Year, C2=Actual, D1=Sales, D2=Forecast.

	A	B	C	D
1			Year	Sales
2			Actual	Forecast
3	Market	Product		

2. On the Smart View ribbon, reconnect to Basic cube in the Sample_Dynamic application.



Your previous connection URL should be shown in the list of Private Connections.

3. When prompted to log in, connect as the user you provisioned.
4. To populate cells with data values, click **Ad hoc analysis.**
In the resulting grid, you should be able to see the results of your calculation. The yearly sales data refreshes for both Actual and Forecast, and the forecast is about 3% higher than the actual:

	A	B	C	D
1			Year	Sales
2			Actual	Forecast
3	Market	Product	400511	412526.3

5. To test that the calculation is correct, create this Excel formula, =D3/C3, in cell E3, which divides the forecast data by the actual data, to ensure that D3 is 3% higher than C3.

	A	B	C	D	E
1			Year	Sales	
2			Actual	Forecast	
3	Market	Product	400511	412526.3	=D3/C3

The test result should confirm the 3% increase, in which Actual is 400511, Forecast is 412526.3, and E3 is 1.0.

	A	B	C	D	E
1			Year	Sales	
2			Actual	Forecast	
3	Market	Product	400511	412526.3	1.03

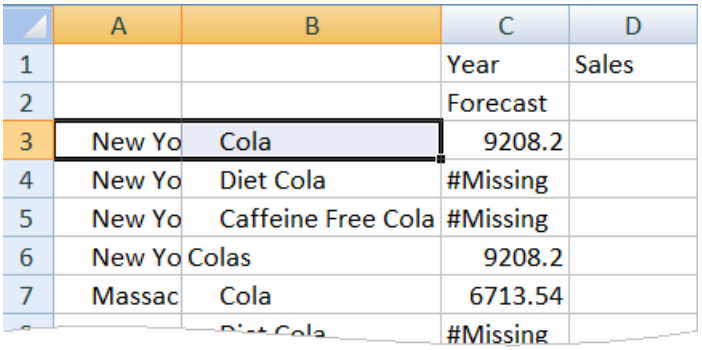
6. Zoom in on Product and Market. You can see that for all products and all markets, the forecast data is present and is 3% higher than the actual.

	A	B	C	D
1			Year	Sales
2			Actual	Forecast
3	East	Colas	27740	28572.2
4	East	Root Beer	23672	24382.16
5	East	Cream Soda	20241	20848.23
6	East	Fruit Soda	15745	16217.35
7	East	Diet Drinks	7919	8156.57
8	East	Product	87398	90019.94
9	West	Colas	28306	29155.18
10	West	Root Beer	34200	35226
11	West	Cream Soda	35391	36452.73
12	West	Fruit Soda	35034	36085.02
13	West	Diet Drinks	36423	37515.69
14	West	Product	132931	136918.9
15	South	Colas	16280	16768.4

7. Now, build a worksheet that you will use to do a data analysis on the forecast, and make some changes.
   a. Click the cell containing Forecast, then click **Keep Only.**
   b. Select cells A3-B3 containing East and Colas, then click **Keep Only.**
   The grid should now look like this:

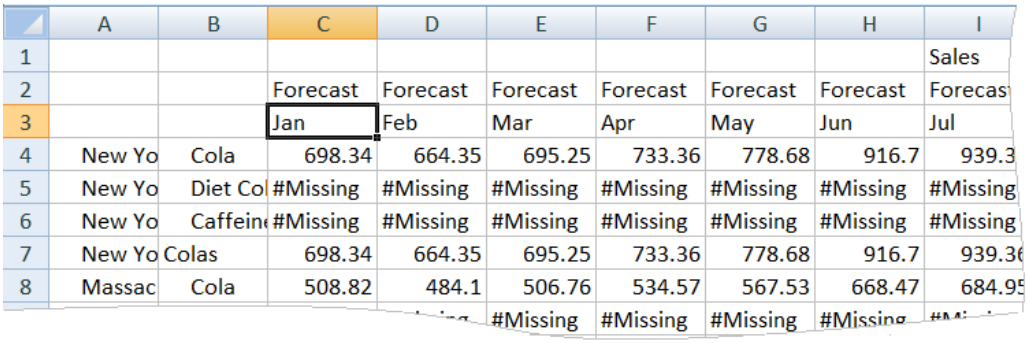
	A	B	C	D
1			Year	Sales
2			Forecast	
3	East	Colas	28572.2	

c. With cells A3-B3 still selected, click **Zoom In** to view per-state information for detailed product SKUs.
The grid should now look like this:



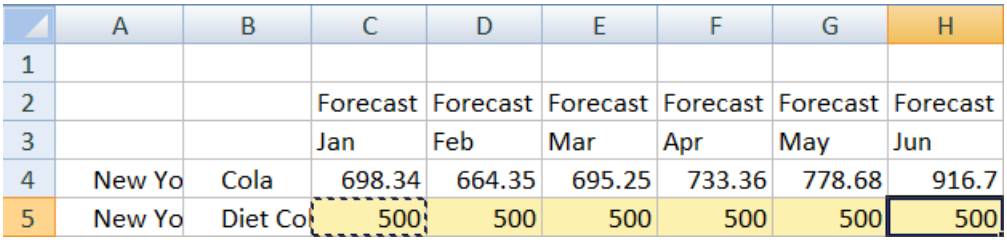
	A	B	C	D
1			Year	Sales
2			Forecast	
3	New Yo	Cola	9208.2	
4	New Yo	Diet Cola	#Missing	
5	New Yo	Caffeine Free Cola	#Missing	
6	New Yo	Colas	9208.2	
7	Massac	Cola	6713.54	
		Diet Cola	#Missing	

d. Pivot the Year dimension down into the columns. Highlight member **Year**, and select the arrow next to **zoom in** on the Essbase ribbon. Select **Zoom to bottom** to see the bottom level of the months.
The grid should now look like this:



	A	B	C	D	E	F	G	H	I
1									Sales
2			Forecast	Forecast	Forecast	Forecast	Forecast	Forecast	Forecas
3			Jan	Feb	Mar	Apr	May	Jun	Jul
4	New Yo	Cola	698.34	664.35	695.25	733.36	778.68	916.7	939.3
5	New Yo	Diet Co	#Missing	#Missing	#Missing	#Missing	#Missing	#Missing	#Missing
6	New Yo	Caffein	#Missing	#Missing	#Missing	#Missing	#Missing	#Missing	#Missing
7	New Yo	Colas	698.34	664.35	695.25	733.36	778.68	916.7	939.36
8	Massac	Cola	508.82	484.1	506.76	534.57	567.53	668.47	684.95
					#Missing	#Missing	#Missing	#Missing	

e. Enter some monthly values to create a Diet Cola forecast. For example, enter 500 in each of the cells in the range C5:H5.



	A	B	C	D	E	F	G	H
1								
2			Forecast	Forecast	Forecast	Forecast	Forecast	Forecast
3			Jan	Feb	Mar	Apr	May	Jun
4	New Yo	Cola	698.34	664.35	695.25	733.36	778.68	916.7
5	New Yo	Diet Co	500	500	500	500	500	500

f. Click **Submit Data**, and notice that the full year forecast in cell O5 changes to 3000, which is the sum of 500 in each of 6 months.

In this task, you learned how easy it is to analyze and edit the cube in Smart View, as long as you have the correct provisioning.

In Create an Application and Cube in Cube Designer, you'll get familiar with Cube Designer.

Create an Application and Cube in Cube Designer

In Analyze Forecast Data in Smart View, you analyzed data in Excel. Users working in Excel can design and deploy applications using cube designer.

Open an application workbook in cube designer. Deploy, load, and calculate a cube. View the cube in the Essbase web interface.

Open the Application Workbook in Cube Designer

Log in as a Power User and download the Sample_Basic.xlsx application workbook from the Gallery.

1. In Excel, on the Cube Designer ribbon, click **Catalog**  .

   If you are prompted to log in, then log in as a Power User.
2. Click **Gallery**, then navigate to `Applications > Demo Samples > Block Storage`, and double-click **Sample_Basic.xlsx.**

The Sample Basic application workbook is different from the Sample Basic Dynamic application workbook in that the Product and Market dimensions do not have dynamically calculated members.

For example, go to the Dim.Market worksheet in `Sample_Basic.xlsx`. Look at the **Storage** column. There are no X characters, which indicates that the members are stored. X characters in the **Storage** column indicate dynamically calculated members.

Therefore, after creating the dimensions and loading the data, you also need to calculate the cube.

Creating, loading, and calculating the cube can all be done in one step in the Build Cube dialog box.

Create, Load, and Calculate the Cube

Use Cube Designer to create, load, and calculate a cube from the `Sample_Basic.xlsx` application workbook.

1. On the Cube Designer ribbon, with the Sample Basic application workbook

   (`Sample_Basic.xlsx`) still open, click **Build Cube**  .
2. On the **Build Option** menu, select **Create Cube.**
3. Click **Run.**

   If there is an existing application with the same name, you are prompted to overwrite the application and cube. Click **Yes** to delete the original application and build this new application.
4. Click **Yes** to confirm your selection.

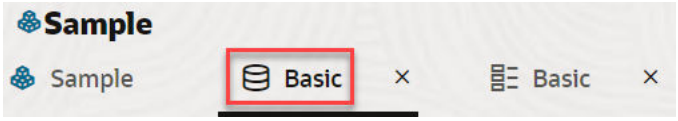
   The **View Jobs** icon displays an hourglass while the job is in progress. The job runs in the background, and Cube Designer notifies you when the job is completed, which should display **Success**.
5. Click **Yes** to launch the Job Viewer and see the status of the job.

View the Application in the Web Interface

View and inspect the new application in the Essbase web interface, check that both level zero and upper-level blocks exist to confirm that the cube is fully calculated.

In the Redwood Interface:

1. Log into the Essbase web interface.
2. On the Home page, navigate to the **Sample** application, and select the Actions menu ⋯.
3. Choose **Databases** > **Basic** > **Launch Outline**.
   View the outline, and see that the expected dimensions are present.
4. Go to the Basic cube tab.



5. On the **General** page, under **Statistics**, you see that both level 0 and upper-level blocks exist, showing that the cube is fully calculated.

In the Classic Web Interface:

1. Log into the Essbase web interface.
2. On the Applications page, expand the **Sample** application and select the **Basic** cube.
3. Click the Actions menu to the right of the **Basic** cube and select **Outline.**

   View the outline, and see that the expected dimensions are present.
4. Return to the Applications page, expand the **Sample** application, and select the **Basic** cube.
5. Click the Actions menu to the right of the **Basic** cube and select **Inspect.**
6. In the inspector, select **Statistics**.
7. On the **General** tab, in the **Storage** column, you see that both level 0 and upper-level blocks exist, showing that the cube is fully calculated.

In Analyze Data and Perform an Incremental Update in Cube Designer, you'll analyze data in this cube and perform incremental updates from Excel.

Analyze Data and Perform an Incremental Update in Cube Designer

In Create an Application and Cube in Cube Designer, you executed a cube build, loaded data, and ran the calculation script defined in the workbook.

Analyze data from a query worksheet in cube designer. Add members to the cube.

Analyze Data in the Sample Basic Cube

Validate that the cube build was successful and see how to analyze data.

1. In Excel, on the cube designer ribbon, click **Analyze**  .

2. On the **Analyze** menu, select **Connect Query Sheets.**

   If you are prompted to log in, then enter your Essbase user name and password.
3. You're connected to the Basic cube in the Sample application.
4. You can now analyze the data.
   a. Use the Essbase ribbon to zoom in on **Cream Soda** to see all of the low-level products that are part of the Cream Soda family.
   b. Zoom out on **New York** to see all of the East region, and zoom out again to see all Markets.

Perform an Incremental Update on the Sample Basic Cube

Add a hierarchy to the product dimension and see the results in Smart View.

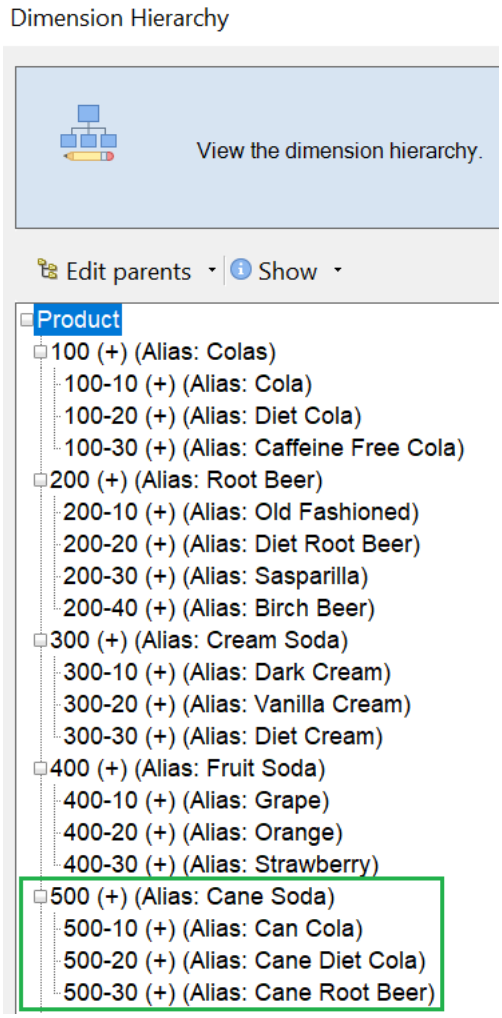
1. Go to the Dim.Product worksheet, where you'll update the product dimension with some extra products.
2. Insert new members into the workbook, following the 400 product family.
   a. Create a new parent Product with child 500 and give it the Alias Default name Cane Soda.
   b. Create three new SKUs with parent 500: 500-10, 500-20, and 500-30.
   c. Give aliases to the new SKUs. Call them Cane Cola, Cane Diet Cola, and Cane Root Beer.

Product	400			Fruit Soda
400	400-10			Grape
400	400-20			Orange
400	400-30			Strawberry
Product	500			Cane Soda
500	500-10			Cane Cola
500	500-20			Cane Diet Soda
500	500-30			Cane Root Beer
Product	Diet		~	Diet Drinks
Diet	100-20			Shared Diet Cola
Diet	200-20			Diet Root Beer
Diet	300-30			Diet Cream

3. Save the updated workbook.
4. Using the cube designer ribbon, click **Build Cube**  .

   The build option will default to **Update Cube – Retain All Data** since the application already exists on the server and you are the application owner who created it.
5. Click **Run**.
6. When the job completion notice is displayed, click **Yes** to launch the **Job Viewer**.
7. You should see **Success**. If the job returns **Error**, then you can double-click the job for more information.
8. Close the **Job Viewer.**
9. With the Dim.Product sheet active, click **Hierarchy Viewer** in the cube designer ribbon.

10. On the Dimension Hierarchy dialog box, see that the Cane Soda product group was created.



11. Go to the query worksheet, Query.Sample.
12. Navigate to the top of the Product dimension by highlighting Dark Cream and zooming out using the Essbase ribbon. Then zoom out on Cream Soda.
13. Select Product again and click **Zoom In.**
14. Select Cane Soda and click **Keep Only.**
15. Select Cane Soda and **Zoom In** to see the child members.

Adding members to the Product dimension does not populate those members with data. Data can be submitted using Smart View or by performing a data load.

Application workbooks are convenient tools for designing Essbase cubes when you already understand the elements needed to build a cube or when you have a sample.

In Transform Tabular Data into a Cube, you will create an application using a columnar Excel worksheet without any Essbase-specific structure.

Transform Tabular Data into a Cube

Although data from external source systems such as ERP tables or a data warehouse is not formatted as an Essbase application workbook, you can still use Cube Designer to transform it into an Essbase cube.

To transform some sample data into an Essbase database using Cube Designer, use the following workflow:

1. In Excel, select the Cube Designer ribbon, then click **Catalog**  .
2. On the Essbase Files dialog box, in the **Gallery**, navigate to the `Technical > Table Format` folder, and double click **Sample_Table.xlsx**.

   The `Sample_Table.xlsx` file contains a worksheet, Sales, which represents a common, simple sales report that you might receive from someone in your organization. The column headings indicate that there are measures (such as Units and Discounts), time representations (such as Time.Month and Time.Quarter), geographic regions (such as Regions.Region and Regions.Areas), and products (such as Product.Brand and Product.LOB).

   From this report, you can create an application and cube by using introspection, which is a method of inspecting a physical data source (in this case, the `Sample_Table.xlsx` file) for Essbase metadata elements.
3. On the Cube Designer ribbon, click **Transform Data.**
4. On the Transform Data dialog box, you can accept the default names for the application (Sample_Table) and cube (Sales) or you can change them.
5. Cube Designer inspects the tabular data to detect relationships that determine appropriate dimensionality.
6. Click **Run** and, when prompted to create the cube, click **Yes.**
7. When the job is completed, you'll see the Job Viewer dialog box.

   Click **Yes** until the status is Success.
8. Close the Job Viewer.
9. Log into the Essbase web interface.
10. View cube statistics:

    In the Redwood Interface:

    a. On the Applications page, open the **Sample_Table** application and then open the **Sales** database (cube).

    b. On the **General** page, click **Statistics**.

    c. In the **Storage** column, the number 4,928 for **Existing level 0 blocks** indicates that data has been loaded into the cube.

    In the Classic Web Interface:

    a. On the Applications page, expand the **Sample_Table** application and select the **Sales** cube.

    b. Click the Actions menu to the right of the **Sales** cube and select **Inspect**.

    c. Select **Statistics**, and on the **General** tab, under **Storage**, the number 4928 for **Existing level 0 blocks** indicates that data has been loaded into the cube.

11. Launch the outline and view dimensions:
    - In the Redwood Interface, still on the **General** page, select **Details**, and then select **Launch Outline**.
    - In the Classic Web Interface, use the **General** tab at the top of the database inspector to launch the outline.

    In the outline editor, you can see that the Sales cube has the following dimensions: Measures, Time, Years, Geo, Channel and Product.
12. Expand **Measures**.

    You'll notice that Units, Discounts, Fixed Costs, Variable Costs, and Revenue are in a flat hierarchy.

In Export and Modify Metadata and Data, you'll create a hierarchy for these Measures so that you can see Revenue net of Discounts, and total costs (fixed and variable).

Export and Modify Metadata and Data

In Transform Tabular Data into a Cube, you created an application and cube from tabular data.

Export a cube, including its data to Excel from the Essbase web interface, and then open the exported application workbook to examine the format.

1. Export to Excel.

   In the Redwood Interface:

   a. In the Essbase web interface, on the Home page, open the **Sample_Table** application, and then open the **Sales** database (cube).

   b. From the **Actions** menu, select **Export to Excel**.

   In the Classic Web Interface:

   a. In the Essbase web interface, on the Applications page, expand the **Sample_Table** application, and select the **Sales** cube.

   b. From the **Actions** menu, select **Export to Excel**.
2. On the Export To Excel dialog box, select the Parent-Child **Export Build Method**.
3. Select **Export Data** and click **OK**.
   - If the data size is less than 400 MB, this exports the metadata and data to an Excel file called an application workbook. Save the application workbook, `Sales.xlsx`, to your Downloads area. The application workbook defines the cube that you exported.
   - If the data size exceeds 400 MB, the data file is saved in a compressed file and is not included in the exported Excel file. The ZIP file containing the data and the application workbook can be downloaded from **Files** page.
4. Open `Sales.xlsx`.
5. Scroll to the Data.Sales worksheet to view it. This is the data worksheet for the cube.

   Examine the worksheets for each of the dimensions. The dimension worksheets begin with `Dim`, including the worksheet for the Measures dimension.
6. Using the exported application workbook, you can make further incremental updates. For example, you can add or remove hierarchies, append a formula to a measure, change aliases, and develop calculations, among many other tasks.

The sequenced tasks in this chapter are intended to show you how you can design and deploy cubes from application workbooks or tabular data. You can incrementally improve the design of your cubes by exporting them to application workbooks, making modifications, and rebuilding.

3

Work with Cubes in Cube Designer

You can create or modify application workbooks and then deploy cubes to Essbase using Cube Designer, a Smart View extension.

About Cube Designer

Cube designer helps you to design, create and modify application workbooks to meet their strict layout and syntax requirements.

The basic components of Cube Designer are the Cube Designer ribbon and the Designer Panel. See About the Cube Designer Ribbon and About the Designer Panel.

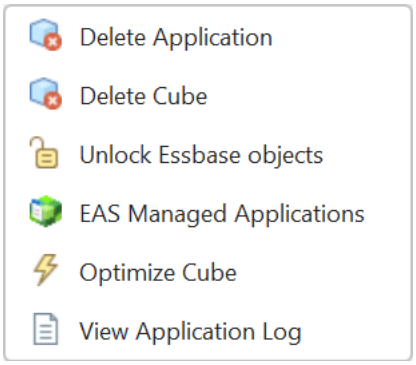
About the Cube Designer Ribbon

You can use options on the Cube Designer ribbon to perform a number of cube management tasks, such as loading data, editing formulas and viewing jobs.

Cube Designer Ribbon Options

- **Connections**: Opens the Connections dialog box, in which you choose the Essbase URL.
- **Catalog**: Opens the Essbase Files dialog box, which contains a selection of prebuilt application workbooks, from which you can build sample applications and cubes.
  Also, a catalog toolbar is available in this dialog box from which that you can perform many file operations within the catalog, such as upload, download, cut, copy, paste, delete, rename, and create a new folder.
- **Local**: Provides a drop-down menu with options to open or save an application workbook locally, or to export a cube to an application workbook.

- **Designer Panel**: Opens the Designer Panel, a series of panels in which you can design and edit application workbooks.
- **Calculation Editors** cube designer ribbon icon is context aware. When you've selected a formula cell on a dimension worksheet, the **Member formula editor** option is enabled. When you've selected a calculation worksheet, the **Calculation script editor** option is enabled.
- **Hierarchy Viewer**: Opens the Dimension Hierarchy dialog box, in which you can view the hierarchy for the selected dimension worksheet in an application workbook, and perform tasks, such as renaming members and changing storage settings. See Work with Dimension Worksheets in Cube Designer.
- **Federated Partition**: Opens the Federated Partition wizard, where you can create a federated partition in Cube Designer by creating a Cube.FederatedPartition worksheet in the application workbook for your Essbase cube, and then validating the sheet and saving the partition to the server.
- **Build Cube**: Opens the Build Cube dialog box, where you can build a cube from the active application workbook. In this dialog box, cube designer automatically detects existing data and calculation worksheets, and then pre-selects options to load the data and run the worksheets.
- **Load Data**: Opens the Load Data dialog box, which contains options to clear all data and to load data.
- **Calculate**: Opens the Calculate Data dialog box, in which you can select an application, a cube, and a calculation script to execute.
- **Analyze**: Provides a drop-down menu with options to create a Smart View ad hoc grid, or connect application workbook query worksheets (Query.*query_name* worksheets) to Smart View.
- **View Jobs**: Opens the Job Viewer dialog box, in which you can monitor the status of jobs, such as data loads, calculations, imports, and exports.
- **Transform Data**: Opens the Transform Data dialog box, which lets you build a cube from tabular data.
- **Options**: Provides options to specify the default working folder and to activate the cube designer log.
- **Admin Tasks**: Opens a menu from which you can delete an application, delete a cube, unlock objects, set applications to EAS Managed, open the expert mode (optimize cube) dialog box, or view the application log.



- ***Server name***: Shows the currently defined connection location. When you click ***Server name*** and log in (if prompted to do so), the server name and the client and server versions are displayed.

About the Designer Panel

The Designer Panel uses a manual system of reading and writing to the worksheets in an application workbook. The **From Sheet** button at the bottom of the Designer Panel reads the entire application workbook's data and populates the panel with the data. The **To Sheet** button updates the entire application workbook with the data from the Designer Panel. The **Reset** button clears the data from the Designer Panel.

One common use of panel is to populate it with information from one application workbook using **From Sheet**, open a new blank workbook, and then use **To Sheet** to make a clone of the first application workbook.

You can design and edit application workbooks in the Designer Panel. Each of its five tabs correspond to one of the five types of worksheets in an application workbook. See Design and Create Cubes Using Application Workbooks.

To open the panel, click **Designer Panel**  on the Cube Designer ribbon.

If the Smart View panel displays when you click **Cube Designer**, then click **Switch To**  , and select **Cube Designer** from the drop down menu.

The Designer Panel contains the following tabs:

- **Cube**: You can design and modify the Essbase.Cube worksheet in an application workbook.

  See Work with the Essbase.Cube Worksheet in Cube Designer.
- **Settings**: You can design and modify the Cube.Settings worksheet in an application workbook.

  See:

  – Work with the Cube.Settings Worksheet: Alias Tables in Cube Designer.
  – Work with the Cube.Settings Worksheet: Properties in Cube Designer.
  – Work with the Cube.Settings Worksheet: Dynamic Time Series in Cube Designer.
  – Work with the Cube.Settings Worksheet: Attribute Settings in Cube Designer.
  – Work with Typed Measures Worksheets in Cube Designer
- **Dimensions**: You can design and modify the Dim.*dimname* worksheets in an application workbook.

  See Work with Dimension Worksheets in Cube Designer.
- **Data**: You can design and modify the Data.*filename* worksheet in an application workbook.

  See Work with Data Worksheets in Cube Designer.
- **Calc**: You can design and modify the Calc.*scriptname* worksheet in an application workbook.

  See Work with Calculation Worksheets in Cube Designer.

Manage Files in Cube Designer

Your access to view and work with Essbase files and artifacts from Cube Designer depends on your permissions. In Cube Designer, you access the file folders in the Catalog using the **Catalog** option in the Cube Designer ribbon.

To access the **Applications** folder, you need to have at least Database Manager application permission.

The **Gallery** folder is read-only access for all users.

The **Shared** folder is read-write access for all users.

The **Users** folder is read-write access for the logged in user.

According to your permissions, you can create, move, rename and delete custom folders. Similarly, users with access can import, export, copy, move, rename and delete files.

Related topic: Manage Essbase Files and Artifacts

Download Sample Application Workbooks

Using the sample application workbooks provided in the Essbase Files dialog box, you can quickly create sample applications and cubes. The cubes are highly portable, because they are quickly and easily imported and exported.

1. On the Cube Designer ribbon, click **Catalog**  .
2. If prompted to connect, enter your user name and password.
3. On the Essbase Files dialog box, choose the sample application workbook you want to open.

You can then edit the application workbook to fit your requirements in the Designer Panel. See Work with Application Workbooks in Cube Designer.

You can save this modified application workbook to your private inventory. See Build a Private Inventory of Application Workbooks.

You can upload this modified application workbook to either the user or shared catalog locations. If uploaded to the shared catalog location, the application workbook will be available to all users.

You can build an application and cube using this application workbook. See Create an Application and Cube in Cube Designer.

Build a Private Inventory of Application Workbooks

Cube Designer allows you to create and store application workbooks on the client computer. This lets you keep a private inventory of completed and in-progress application workbooks.

Using the **Local** icon menu items on the Cube Designer ribbon, you can manage your private application workbook inventory.

Open an Application Workbook

Open an existing application workbook from your inventory.

1. On the Cube Designer ribbon, click **Local**  .
2. Select **Open Application Workbook.**
3. Browse to the application workbook and click **Open.**

Save an Application Workbook

Save a new or updated application workbook to your inventory.

1. Open the application workbook.
2. On the Cube Designer ribbon, click **Local**  .
3. Select **Save Application Workbook.**
4. Browse to your inventory location and click **Save.**

Export to an Application Workbook

Export a cube to an application workbook and add it to your inventory.

1. On the Cube Designer ribbon, click **Local**  .
2. Select **Export Cube to Application Workbook.**
3. If prompted to log in to Essbase, enter your user name and password.
4. In the **Export Cube** dialog box, select the application and cube you want to export, and from the **Export Build Method** menu, select either the **Parent-Child** or **Generation** build method; indicate if you want to export input level data and calculation scripts, and click **Run.**
5. To add the application workbook to your private inventory, click **Save Application Workbook.**

Work with Application Workbooks in Cube Designer

Each application workbook contains a number of worksheets that, taken together, define the cube. Using Designer Panel, you can modify an application workbook, and then you can use the modified workbook to create an updated cube, reflecting your changes.

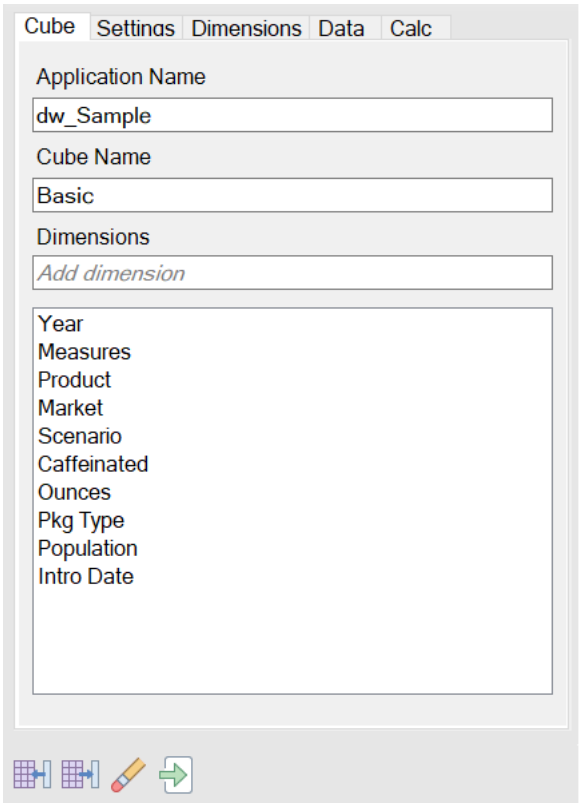
- Work with the Essbase.Cube Worksheet in Cube Designer
- Work with the Cube.Settings Worksheet: Alias Tables in Cube Designer
- Work with the Cube.Settings Worksheet: Properties in Cube Designer
- Work with the Cube.Settings Worksheet: Dynamic Time Series in Cube Designer
- Work with the Cube.Settings Worksheet: Attribute Settings in Cube Designer
- Work with the Cube.Settings Worksheet: Substitution Variables in Cube Designer
- Work with Dimension Worksheets in Cube Designer
- Work with Data Worksheets in Cube Designer
- Work with Calculation Worksheets in Cube Designer
- Work with MDX Worksheets in Cube Designer
- Work with Typed Measures Worksheets in Cube Designer
- Create a Cube from a Local Application Workbook in Cube Designer

Work with the Essbase.Cube Worksheet in Cube Designer

Using the Cube tab in the Designer Panel, you can modify the Application Name, Cube Name, and Dimension Definitions fields on the Essbase.Cube worksheet. You can change the application name and cube name, and delete one or more dimensions.

1. On the Cube Designer ribbon, select **Designer Panel** 

.
2. In the Designer Panel, select the **Cube** tab.



3. Select **From Sheet**  to populate the Designer Panel with the contents of the application workbook.
4. Change the application name or the cube name, if you want to.
5. Add one or more dimensions by typing the name in the text box and pressing the enter key after each one.
6. In the Dimensions list
   - If you want to delete a dimension, right click the dimension name and select **Delete Dimension**.
     Alternatively, you can select a dimension name and press the delete key.
   - If you want to rename a dimension, right click the dimension name and select **Rename Dimension**.
7. Select **To Sheet**  to propagate the changes to the application workbook.

8. Examine the updated application workbook to see your changes.

See also: Understand the Essbase.Cube Worksheet.

Work with the Cube.Settings Worksheet: Alias Tables in Cube Designer

You can add new alias tables in the Cube.Settings worksheet.

1. In the Designer Panel, select the **Settings** tab.
2. Select **From Sheet**  to populate the Designer Panel with the contents of the application workbook.
3. In the **Alias Tables** field, enter a name for the new alias table.
4. Press **Enter.**
5. Select **To Sheet**  .

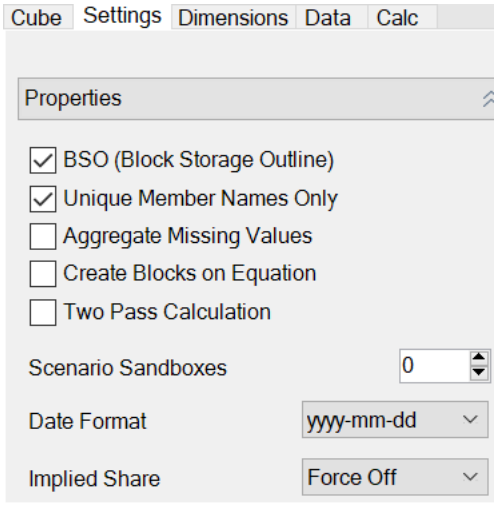
   A new alias table name is added on the Cube.Settings worksheet in the application workbook. To add the alias table to a dimension worksheet, open the Dimensions tab in the Designer Panel, and add the alias table to the selected dimension worksheet. See Work with Dimension Worksheets in Cube Designer. After you add the alias table to the dimension worksheet, you must populate the aliases manually, or by copying from a source.

> **Note**
>
> Changes to the Cube.Settings worksheet cannot be applied incrementally. Instead, you must rebuild the cube to apply those changes.

Work with the Cube.Settings Worksheet: Properties in Cube Designer

You can add new properties in the Cube.Settings worksheet.

1. In the Designer Panel, select the **Settings** tab.
2. Select **From Sheet**  to populate the Designer Panel with the contents of the application workbook.
3. Expand the **Properties** section.



4. Make your selections.
5. Select **To Sheet** to propagate the changes to the application workbook.

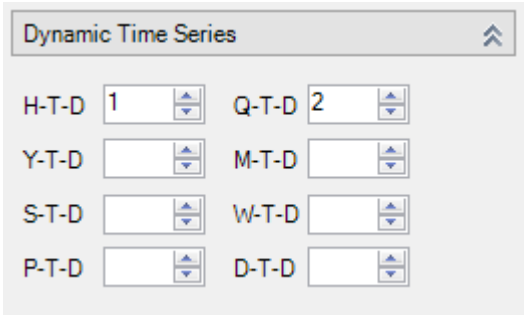
> **Note**
>
> Changes to the Cube.Settings worksheet cannot be applied incrementally. Instead, you must rebuild the cube to apply those changes.

See also: Understand the Cube.Settings Worksheet: Properties.

Work with the Cube.Settings Worksheet: Dynamic Time Series in Cube Designer

You can add dynamic time series members in the Cube.Settings worksheet.

1. In the Designer Panel, select the **Settings** tab.
2. Select **From Sheet** to populate the Designer Panel with the contents of the application workbook.
3. Expand the **Dynamic Time Series** section.



4. Make the changes that you want.

5. Select **To Sheet** to propagate the changes to the application workbook.

There are reserved generations names used by dynamic time series. For example, using the generation name of “Year” activates dynamic time series for “Y-T-D.”

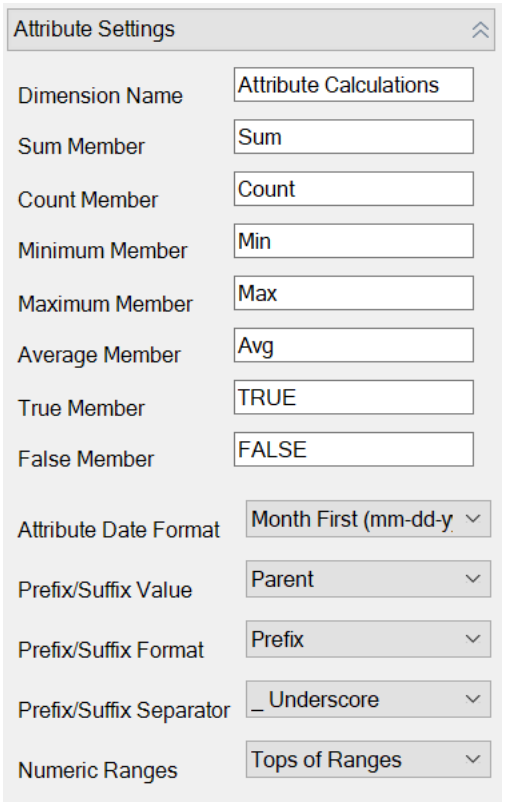
> **Note**
>
> Changes to the Cube.Settings worksheet cannot be applied incrementally. Instead, you must rebuild the cube to apply those changes.

See also: Understand Dimension Worksheets.

Work with the Cube.Settings Worksheet: Attribute Settings in Cube Designer

You change attribute settings on the Cube.Settings worksheet.

1. In the Designer Panel, select the **Settings** tab.
2. Select **From Sheet** to populate the Designer Panel with the contents of the application workbook.
3. Expand the **Attribute Settings** section.



4. Make the changes that you want.
5. Select **To Sheet** to propagate the changes to the application workbook.

> **Note**
>
> Changes to the Cube.Settings worksheet cannot be applied incrementally. Instead, you must rebuild the cube to apply those changes.

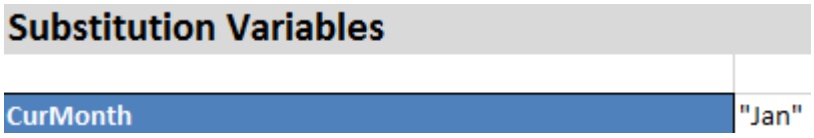
See also: Understand the Cube.Settings Worksheet: Attribute Settings.

Work with the Cube.Settings Worksheet: Substitution Variables in Cube Designer

You can add cube-level substitution variables on the Cube.Settings worksheet.

Enter the name of the substitution variable in column A. Enter the corresponding value of the substitution variable in column B.

You must enclose member names in double quotes.

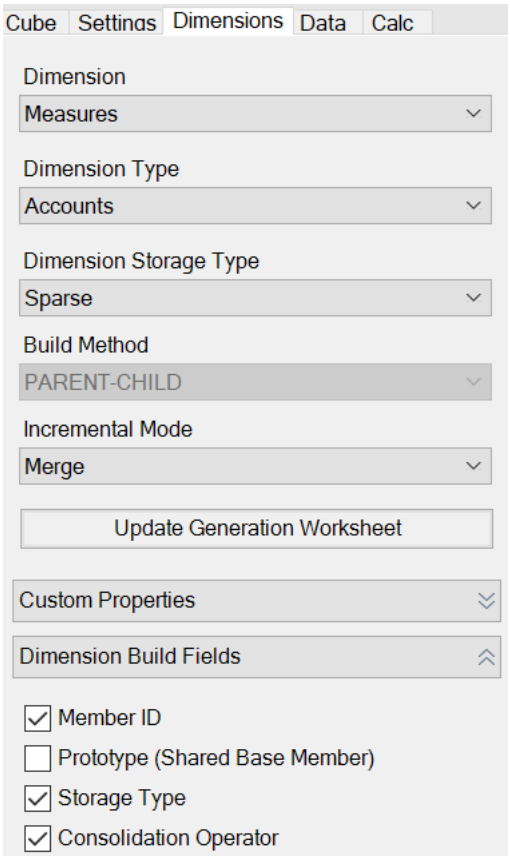


> **Note**
>
> Changes to the Cube.Settings worksheet cannot be applied incrementally. Instead, you must rebuild the cube to apply those changes.

Work with Dimension Worksheets in Cube Designer

On dimension worksheets, you can define dimensions, including name, type, storage type, build method, and more.

1. In the Designer Panel, select the **Dimensions** tab.



2. Select **From Sheet** to populate the Designer Panel with the contents of the application workbook.
3. Make the changes that you want.

   For descriptions of the options and valid values, see Understand Dimension Worksheets.
4. (Optional) If you want to update the Cube.Generations worksheet in the application workbook for this dimension, click the **Update Generation Worksheet** button.

   The **Update Generation Worksheet** button creates a section in the Cube.Generations worksheet for the dimension selected in the **Dimension** drop down list on the **Dimensions** tab of the Designer Panel.

   The Dimension section of the Cube.Generations worksheet changes if you add or delete members on the dimension worksheet (Dim.*dimname*), causing the number of generations in the dimension to change. If you make changes to the dimension worksheet by adding or deleting members, you should always press the **Update Generation Worksheet** button as part of the editing process.
5. Select **To Sheet** to propagate the changes to the application workbook.

- After adding alias tables using Designer Panel, populate the alias table column with alias names manually, or by copying them from a source.
- Use no more than 1024 characters when naming dimensions, members, or aliases.
- The length limit for the dimension worksheet is 30 characters, including 3 characters for the "Dim." at the beginning of the sheet name. So, the name following "Dim." can contain up to 27 characters.

See Understand the Cube.Generations Worksheet.

Work with Data Worksheets in Cube Designer

You can create data worksheets in the Designer Panel for new or existing application workbooks. You can also edit the display of dimensions and members in new or existing data worksheets.

To create a new data worksheet:

1. In the Designer Panel, select the **Data** tab.
2. Enter a name for the new data worksheet in the **Data Sheets** field.
3. In **Sheet Type**, select the type of sheet to create:
   - **Data Dimension**
     In a data dimension sheet, dimensions are represented in the column headers. These indicate the member combinations to which to load the data. The right-most columns are the data columns. The data column headers specify members of the data dimension, which is the dimension to which you are loading data. The data values reside in the data column rows.
   - **Flat**
     In a flat sheet, each dimension is represented in a column header. The last column, ***Data*** contains the data values for the specified member combinations.
   - **Sandbox**
     In a sandbox sheet, each dimension is represented in a column header. The first dimension is **Dimension.Sandbox**. The last three column headers display members of the CellProperties dimension: **EssValue**, **EssStatus**, and **EssTID**. Do not modify the EssValue, EssStatus, and EssTID columns, as they are for internal purposes and it is important that they not be changed.
4. Press **Enter.**
5. Optional: Edit the data layout. Change the order of the dimension columns, and (for the Data Dimension sheet type only) select members and set their order. See subsequent sections in this topic for instructions.
6. Select **To Sheet**  .
   After you select a sheet type, and then select **To Sheet** or **From Sheet**, you cannot change the sheet type from the Designer Panel.

A new data worksheet is created in the application workbook.

To change the order of dimensions in the data worksheet:

1. In the Designer Panel, select the **Data** tab.
2. In **Data Sheets**, select the sheet that you want to edit.
3. In **Dimension Column Order**, select the dimension that you want to move.
4. Use the up and down arrows to move the dimension.
5. Select **To Sheet**  to add your changes to the selected **Data** tab in the worksheet.

To change the order of members on the data worksheet (Data Dimension sheet type only):

1. In the Designer Panel, select the **Data** tab.
2. In **Data Columns**, select the member that you want to move.
3. Use the up and down arrows to move the member.

4. Select **To Sheet**  to add your changes to the selected **Data** tab in the worksheet.

To select the members to display in a data worksheet (Data Dimension sheet type only):

1. In the Designer Panel, select the **Data** tab.
2. Click **Member Selection**.
3. In the **Member Selector**, check the members you want to display and clear the members you don't want to display.
4. Click **OK**.
5. Select **To Sheet**  to add your changes to the selected **Data** tab in the worksheet.

To add data worksheets to an existing application workbook, go to the **Data** tab in the Designer Panel, click **From Sheet**  , and proceed with the steps in this topic.

Work with Calculation Worksheets in Cube Designer

You can create new calculation worksheets in the Designer Panel.

1. In the Designer Panel, select the **Calc** tab.
2. Select **From Sheet**  to populate the Designer Panel with the contents of the application workbook.
3. In the **Calculation Sheets** field, enter a name for the new calculation worksheet.
4. Press **Enter.**
5. Select **To Sheet**  .

   A new calculation worksheet is created in the application workbook.

Cube Designer calculation worksheets apply only to block storage cubes.

Work with MDX Worksheets in Cube Designer

You can create new MDX worksheets in the Designer Panel.

1. In the Designer Panel, select the **Calc** tab.
2. Select **From Sheet**  to populate the Designer Panel with the contents of the application workbook.
3. In the **MDX Insert Sheets** field, enter a name for the new MDX worksheet.
4. Press **Enter**.
5. Select **To Sheet**  .

A new MDX worksheet is created in the application workbook.

See Understand MDX Worksheets.

Work with Typed Measures Worksheets in Cube Designer

Using Cube Designer, you can add date measures or text list definitions to application workbooks, to build typed measures in Essbase.

1. Open an application workbook.

2. On the Cube Designer ribbon, open the Designer Panel.
3. Click the **Settings** tab.
4. Click **From Sheet** to populate the Designer Panel with the contents of the application workbook.
5. To add date measures:
   a. In the **Cube.Settings** worksheet, under **Properties**, modify the **Date Format** to the format you want to load into the cube.
   b. If a Cube.TypedMeasures sheet does not exist in the application workbook, add one:
      i. On the **Designer Panel**, **Settings** tab, expand **Text Lists**.
      ii. In the **Text Lists** field, type a name.
      iii. Press **Enter**.
   c. Identify the members in Accounts dimension and add them in cells to the right of **Associated Members** in the **Date Measures** section. These are the members that will allow dates to be loaded into the cube as data.
   d. Rebuild the cube.
6. To add text lists:
   a. If a **Cube.TypedMeasures** sheet does not exist in the application workbook, add one:
      i. On the **Designer Panel**, **Settings** tab, expand **Text Lists**.
      ii. In the **Text Lists** field, type a name.
      iii. Press **Enter**.
      The text list name is moved to the text box below the **Text Lists** field.
   b. If a **Cube.TypedMeasures** sheet already exists in the application workbook, you can create additional text list tables to that sheet by following the steps in 6a and using a new name in the **Text Lists** field.
   c. After you add the text list, you must enter the text list information manually. This includes the associated members for the text list, the valid text items in the list and their related numeric values.
   d. Rebuild the cube.

- Understand the Cube.TypedMeasures Worksheet
- Working with Typed Measures
- Performing Database Operations on Text and Date Measures

Create a Cube from a Local Application Workbook in Cube Designer

Using a sample local application workbook, you can create an Essbase cube from Cube Designer.

1. In Excel, on the Cube Designer ribbon, select **Local**  , and then select **Open Application Workbook**.
2. Select an application workbook, then select **Open.**
3. On the Cube Designer ribbon, select **Build Cube**  .

4. On the Build Cube dialog box, verify that you want to use the selected options. Cube Designer detects data worksheets and calculation worksheets in the application workbook, and pre-selects those options for you, however you can deselect those options if you want to:
   - **Load Data Sheets Contained within Workbook** is pre-selected if data worksheets exist in the workbook. You can de-select this option if you do not want to load data.
   - **Run Calculation Sheets Contained within Workbook** is pre-selected if calculation worksheets exist in the workbook. You can de-select this option if you do not want to run the calculations.
5. Click **Run.**
6. After the asynchronous job completes, a dialog box is displayed. Click **Yes** to launch Job Viewer and view the status of the Excel import, or click **No** if you don't want to launch Job Viewer.

   See Work with Jobs in Cube Designer.

Load Data in Cube Designer

You can use cube designer to load either aggregate storage or block storage data into Essbase.

At times, you may need to clear and reload data during cube development. The data and rules files used in the data load process must be stored in Essbase. If a data worksheet is included in the application workbook, then the data files and rule files are automatically generated during the cube build process. You can also upload individual data and rule files to the cube.

Each data file you select to load starts a separate data load job. By default, the number of jobs that can be run at one time is ten, but you can increase that number. To determine the appropriate limit, consider your calculations, application, restructures, and batch windows, and compare the timing of administrative and user activities so as not to overwhelm your shape size.

To increase the jobs limit, set the essbase.jobs.maxCount Provider Services configuration property to the value that you want. See Set Provider Services Configuration Properties.

It is important to understand the differences between aggregate storage and block storage data loads.

Load Block Storage Data in Cube Designer

To load block storage data using Cube Designer, select the **Load Data** icon on the Cube Designer ribbon, and follow prompts to select job type, source data, and rule file, then run and monitor the job.

1. In Excel, on the Cube Designer ribbon, select **Load Data**  .
2. In the Load Data dialog box, select the application and cube in which you want to load data.
3. Under **Select a Job Type**, select an option:
   - **Load Data**: to load data to the cube.
   - **Clear all Data**: to clear all data from the cube.
4. Click **Select Data**.

5. In the **Select Data** dialog box, click **Add**.
6. In the **Essbase files** dialog box, browse to the data files you want to add and select them. These files can be in the cube directory or another directory that you choose. You can add multiple files at once, or you can add them one at a time.
7. Under each data file that you selected in step 6, click the **Select a load rule file** icon to select the matching rule file.
8. In the **Essbase files** dialog box, browse to the rule file for that data file and select it.

> **Note**
>
> In the case of non-SQL data loads, you must always select a data file. If you select only a rule file (one that is designed for non-SQL loads), and no data file, an error is returned stating that a connection could not be established with the SQL database. The solution is to select the appropriate data file.

9. Click **OK**.
10. Select whether to **Abort on Error**.
    If you select **Abort on Error**, the data load is stopped when an error is encountered.
11. Click **Run** to start the data load.
    One job is created for each data file. The jobs are run in parallel, so they finish more quickly than if they were run separately.
12. Click **Yes** to launch the **Job Viewer** and view the status of each job, or click **No** if you do not want to start the **Job Viewer**.

See Understanding Data Loading and Dimension Building.

Load Aggregate Storage Data in Cube Designer

To load aggregate storage data using Cube Designer, select the **Load Data** icon on the Cube Designer ribbon, and follow prompts to select a job type, settings, source data, and rule file, then run and monitor the job.

1. In Excel, on the Cube Designer ribbon, select **Load Data**  .
2. In the **Load Data** dialog box, select the application and cube in which you want to load data.
3. Click **Select data**.
4. In the **Select Data Files** dialog box, select the settings for your data load.

Property or Field	Values
Duplicate Aggregation Method	Specify how to handle cases where multiple values for the same cell are being loaded from the data stream in the load buffer. • Add – Add values when the buffer contains multiple values for the same cell. • Verify that multiple values for the same cells are identical; if they are, ignore the duplicate values. If the values for the same cell differ, stop the data load with an error message. • Use last – Combine duplicate cells by using the value of the cell that was loaded last into the load buffer. This option is intended for relatively small data loads of up to 10,000s of cells.
Load buffer options	Specify how to handle missing and zero values in the data stream from the load buffer. • Ignore none – Ignore no values in the incoming data stream. • Ignore missing values – Ignore #Missing values in the incoming data stream. • Ignore zero values – Ignore zeroes in the incoming data stream. • Ignore missing and zero values - Ignore both #Missing and zero values in the incoming data stream.
Commit option	Specify load buffer commit options to use when committing the contents of the data load buffer to the cube. • Store data – Store values in the load buffer. • Add data – Add the values in the load buffer to the existing stored data values. • Subtract data – Subtract the values in the load buffer from the existing stored data values. • Override all data – Store incoming data instead of the existing stored data values. • Override incremental data – Reset incremental data (stored in incremental slices). In other words, remove the current contents of all incremental data slices in the cube and create a new data slice with the contents of the specified data load buffer. The new data is created with the data load property **add values** (aggregate_sum). If there are duplicate cells between the new data and the primary slice, their values are added together when you query for them.

Property or Field	Values
Term option	Specify final options for committing data slices to the cube from the data load buffer. • Incr. to main slice – Store data into the main slice, and don't create an incremental slice. • Incr. to new slice – Write the data currently stored in the buffer to a new slice. This operation speeds up the data load. • Incr. to new slice lightweight - Write the data currently stored in the buffer to a new slice in the cube, as a lightweight operation. This option is intended only for very small data loads of up to 1,000s of cells that occur concurrently (for example, grid client data-update operations).
Concurrency	• Sequential – Load data sequentially instead of in parallel. • Parallel – Loads data in parallel. Multiple data load buffers can exist on an aggregate storage cube. To save time, you can load data into multiple data load buffers simultaneously. Although only one data load commit operation on a cube can be active at any time, you can commit multiple data load buffers in the same commit operation, which is faster than committing buffers individually.

5. Click **Add** to select data and rule files.
6. In the **Select Data Files** dialog box, browse to the data files you want to add and select them. These files can be in the cube directory or another directory that you choose. You can add multiple files at once, or you can add them one at a time.
7. Under each data file that you selected in step 6, click the **Select a load rule file** icon to select the matching rule file.
8. In the **Essbase files** dialog box, browse to the rule file for that data file and select it.
9. Select whether to **Abort on Error**.
   If you select **Abort on Error**, the data load is stopped when an error is encountered.
10. Click **Run** to start the data load.
    One job is created for each data file. The jobs are run in parallel, so they finish more quickly than if they were run separately.
11. Click **Yes** to launch the **Job Viewer** and view the status of each job, or click **No** if you do not want to start the **Job Viewer**.

See Loading Data into Aggregate Storage Databases.

Calculate Cubes in Cube Designer

You can create member formulas, create calculation scripts, and execute calculation jobs in Cube Designer.

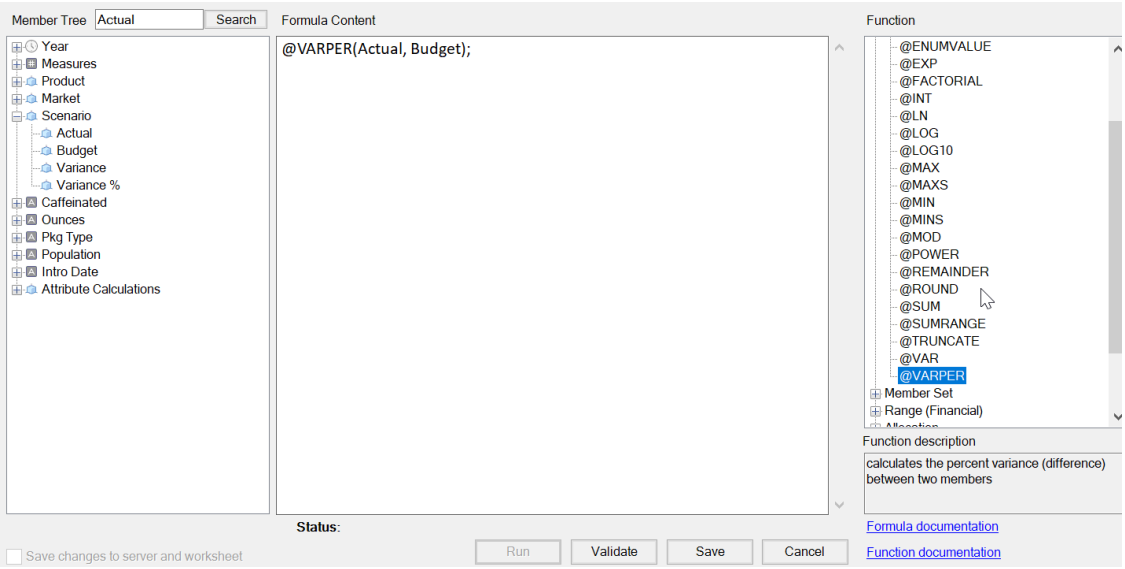
- Create and Validate Member Formulas in Cube Designer
- Create and Validate Calculation Scripts in Cube Designer
- Calculate Data in Cube Designer

Create and Validate Member Formulas in Cube Designer

In the Cube Designer Formula Editor, you can write formulas for specific outline members. You can construct member formulas from operators, functions, dimension names, member names, substitution variables, and numeric constants.

- The Cube Designer Formula Editor applies to both aggregate storage and block storage cubes. In aggregate storage, the functions will be MDX functions. In block storage, the functions will be calc script functions.
- Validation works against existing block storage cubes in Essbase (validation is disabled for aggregate storage cubes). It does not detect application workbook changes that have not been applied to the cube.
- Member selection works with existing cubes only.

Formula Editor provides a formula editing pane in which you can enter a formula. You can use the Tab and arrow keys to move focus within Formula Editor. You can also use a point-and-click approach to select and insert formula components into the formula editing pane. A member selection tree helps you place the correct member names into the formula.



1. Open the application workbook for the cube that you want to modify.
2. If a dimension worksheet has been defined with the Formula property, select the cell in the Formula column for the member you wish to create a formula.
3. On the Cube Designer ribbon, click **Calculation Editors** .
4. From the drop-down menu, select **Member formula editor**.
5. Enter your login credentials for Essbase, if prompted to do so.
6. In the Formula Editor, create the formula.
   - Use the keyboard to enter formula text. Enclose in quotation marks any member names containing blanks or special characters.

- Select a cell containing a member name or alias from any dimension worksheet. Place the cursor in the appropriate location of the editor and right-click to paste that name surrounded by quotes into the editor.
- Double-click on a member in the member selection tree to have that member pasted into the editor.
- To search for a specific member in the tree, enter the member name in the **Member Tree** text box, and click **Search**.
- Double-click on a function to have that function syntax pasted into the editor.

7. Optional: on block storage cubes, click **Validate** to check formula syntax.

   If the validation fails, edit the formula and try again. Be sure to check the error message for guidance.

   The **Validate** button is disabled for aggregate storage cubes.

See:

- Developing Formulas for Block Storage Databases
- Understanding Formula Syntax
- Reviewing Examples of Formulas

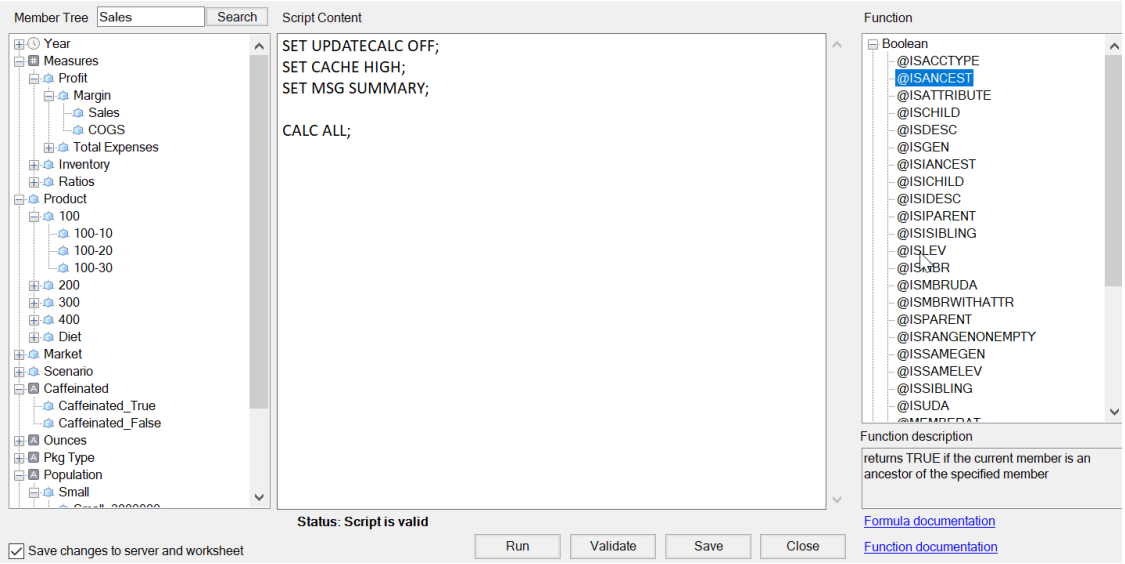
Create and Validate Calculation Scripts in Cube Designer

In the Calculation Script Editor of Cube Designer, you can write calculation scripts for specific block storage cubes. Calculation scripts specify how cubes are calculated and, therefore, override outline-defined cube consolidations.

See Developing Calculation Scripts for Block Storage Databases.

- The Cube Designer Calculation Script Editor applies only to block storage cubes.
- If the cube of the currently open application workbook exists on the server and the communication to the server works, you are in online mode. Otherwise, you are in offline mode. When you are in offline mode, the member tree is disabled.
- If you are editing a script locally, and a script with the same name exists on the server, the **Save changes to server and worksheet** check box is enabled. If you are editing a remote script (one that exists only on the server), the check box is disabled.
- Validation works against existing cubes in Essbase. It does not detect application workbook changes that have not been applied to the cube.

The Calculation Script Editor provides a calculation script-editing pane in which you can enter a script. You can use the Tab and arrow keys to move focus within Calculation Script Editor. A member selection tree helps you place the correct member names into the script.



1. Open the application workbook for the cube you want to modify.
2. If a calculation worksheet has been defined, select it. If not, create one. See Work with Calculation Worksheets in Cube Designer.
3. On the Cube Designer ribbon, click **Calculation Editors** .
4. Select **Calculation script editor** from the drop-down menu.
5. Enter your login credentials for Essbase, if prompted to do so.
6. In the **Calculation Script Editor**, create the calculation script.
   - Use the keyboard to enter text. Enclose in quotation marks any member names containing blanks or special characters.
   - Double-click on a member in the member selection tree to have that member pasted into the editor. To search for a specific member in the tree, enter the member name in the **Member Tree** text box, and click **Search**.
   - Double-click on a function to have that function syntax pasted into the editor.
7. Click **Validate** to check calculation script syntax. If the validation fails, edit the script and try again. Be sure to check the error message for guidance.
8. If you want to synchronize changes to the server and the application workbook, you can check **Save changes to server and worksheet** prior to saving the script.

> **Note**
>
> This option works in online mode.

9. Click **Save**.
10. Optional: Click **Run** to run the script.

Calculate Data in Cube Designer

You can run and manage Essbase calculations using Cube Designer.

Calculation scripts let you specify how block storage (BSO) cubes are calculated and, therefore, override outline-defined cube consolidations. For example, you can calculate cube subsets or copy data values between members. See Developing Calculation Scripts for Block Storage Databases.

During cube development, it is common to recalculate a cube many times when validating the data and formulas. The calculation script files used in the calculation process must be stored in Essbase. If a Calc worksheet is included in the application workbook, then the calculation script files are automatically generated during the cube build process. Individual calculation script files can also be uploaded to Essbase. See Work with Files and Artifacts.

1. In Excel, on the Cube Designer ribbon, select **Calculate**  .
2. On the Calculate Data dialog box, select an application and a cube, and select the calculation script you want to use. To view or edit the calculation script, click **Edit**. Right click on the calc script name to display options: **rename**, **delete**, **run**, **copy**, and **lock**, or **unlock**.
3. Click **Run** to start the calculation.
4. When the asynchronous job completes a dialog box is displayed. Click **Yes** to start Job Viewer and view the status of the calculation, or click **No** if you do not want to start Job Viewer.
5. (Optional) View the status in Job Viewer.

   See Work with Jobs in Cube Designer.

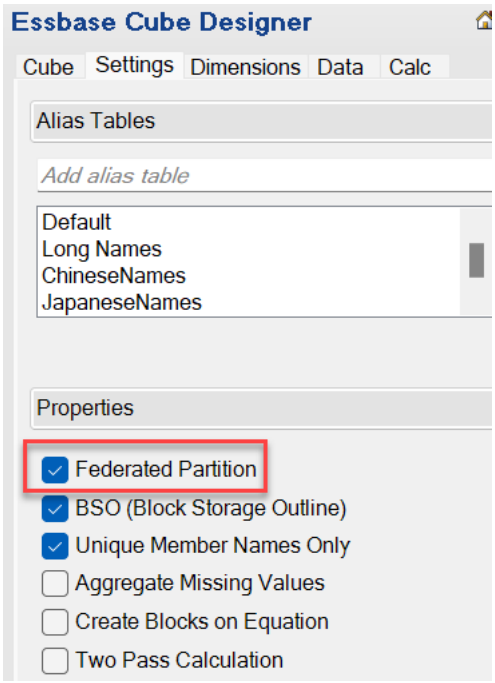
Create a Federated Partition using Cube Designer

To create a federated partition using Cube Designer, add a Cube.FederatedPartition worksheet to the application workbook for your Essbase cube, validate the sheet, and build the federated partition to the Essbase Server.

The federated partition is between Essbase and Autonomous AI Lakehouse Serverless.

This topic assumes you have completed the prerequisites and reviewed the information detailed in Federated Cubes: Integrate Essbase with Autonomous AI Database.

1. Build an Essbase application and cube without a federated partition.
2. Open the application workbook for the cube. If you don't have one, see Export a Cube to an Application Workbook.
3. On the Cube Designer ribbon, click **Cube Designer** to open the Designer Panel.
4. Click **From Sheet**  to populate the Designer Panel with the contents of the sheet.
5. Click the **Settings** tab.
6. On the **Settings** tab, expand **Properties** and select **Federated Partition**.



7. Click **To Sheet** to create a Cube.FederatedPartition sheet in the application workbook.
8. Click **Yes** to edit the new Cube.FederatedPartition worksheet.

   The Federated Partition wizard opens in Cube Designer.
9. For **Connection name**, enter the connection to Autonomous AI Lakehouse that was previously created by an administrator as shown in Create a Connection for Federated Cubes.

> **Note**
>
> When creating a federated partition using Cube Designer, the connection must be a global connection.

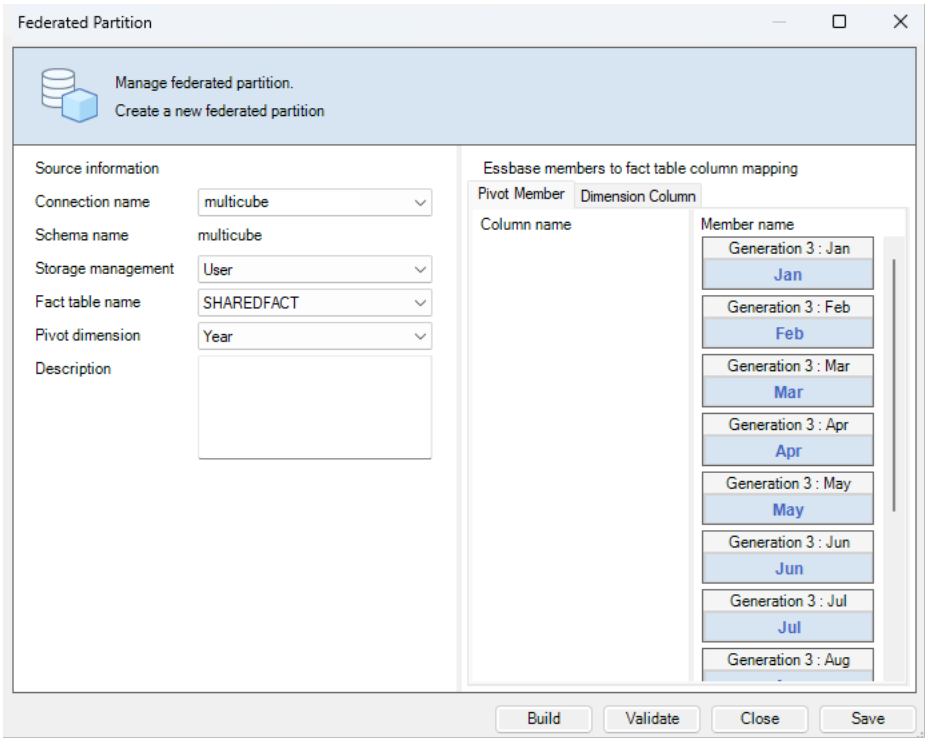
10. For **Schema name**, ensure that it matches the name of the database schema (user name that you entered when you created the connection).
11. For **Storage management**, select **User** or **Essbase** managed.

> **Note**
>
> To let Essbase create and manage a fact table for you, select Essbase managed. See Data Load Options for Federated Cubes.

12. For **Fact table name**, select the name of the fact table in Autonomous AI Lakehouse that stores numeric values and keys.

> **Note**
>
> Skip this step if you selected Essbase managed in the previous step.

13. 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.

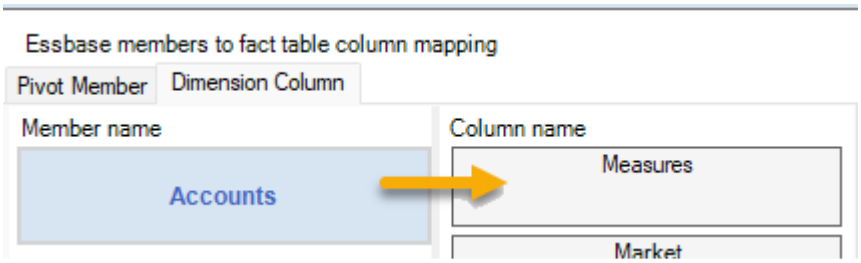
If a member of the pivot dimension (or a non-measures dimension name) includes a special character, such as &, Oracle recommends renaming it.

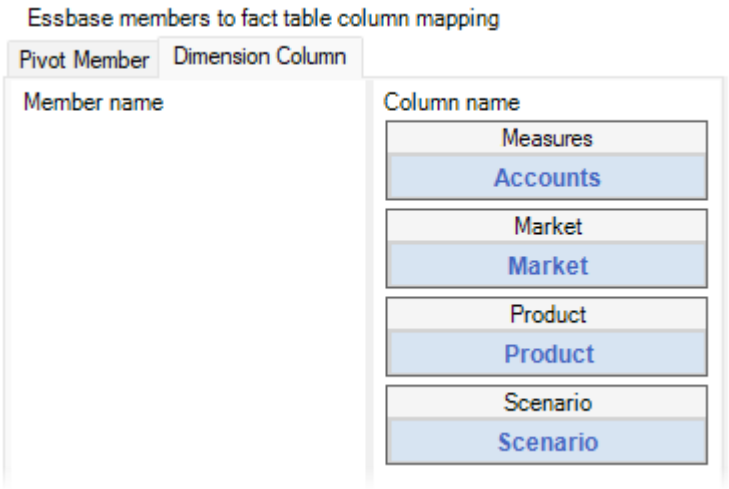
In the **Pivot Member** column, row values are numeric values, or the data. The headers for these columns are member names.

The values in the **Dimension Column** are textual values. These map to Essbase member names. The headers for these columns map to Essbase dimension names.

14. If Essbase dimension and member names do not exactly match the column names in the fact table, map them by dragging and dropping the Essbase names to the appropriate column names in both the **Pivot Member** column and the **Dimension Column**.

    For example, drag and drop **Accounts** to **Measures**.





15. Complete the federated partition creation process:
    a. Click **Validate** to validate the partition.
    b. Click **Save** to save your changes to the Cube.FederatedPartition worksheet.
    c. Click **Build** to build the federated partition on the server.

> **Note**
>
> The **Build** button in the federated partition wizard does not work on aggregate storage cubes.

Alternatively, you can use the **Build Cube** option on the Cube Designer ribbon to build the cube and create the federated partition.

> **Note**
>
> The federated partition build process is launched as a job which can then be monitored in **View Jobs** on the Cube Designer ribbon.

16. 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.
17. Continue to use the wizard to make changes to just the federated partition or rebuild the application with the new saved federated partition settings on the Cube.FederatedPartition

worksheet in the application workbook. See Understand the Cube.FederatedPartition Worksheet.

Work with Jobs in Cube Designer

Use the Cube Designer Job Viewer to view, monitor and troubleshoot jobs that you run from your particular client. Jobs are operations such as data loads, dimension builds, and calculations.

A record of all Essbase jobs is maintained in the Essbase instance. Each job has a unique ID number.

The jobs listed in the Job Viewer are for one specific user. If a different user logs into the client, then only jobs for that user are displayed.

View Jobs in the Cube Designer Job Viewer

You can view jobs for the specific user that is logged into the client in the Cube Designer Job Viewer.

In Excel, on the Cube Designer ribbon, click **View Jobs**  .

The Job Viewer dialog box opens, showing a list of jobs that have been run from that particular client.

Monitor Essbase Jobs in Cube Designer

The Cube Designer ribbon shows when an Essbase job is in progress. After the job finishes, you can view the status of the job in the Cube Designer Job Viewer.

- While a job is running, the **View Jobs** icon on the Cube Designer ribbon displays an hourglass  .
- When the job finishes running a Job Viewer status dialog box displays, indicating the status of the job.

If you close Excel while the job is running, the job continues to run, but you will not see a status dialog when it finishes. The job is an Essbase Server process, so it runs regardless of whether Excel is open or not.

Troubleshoot Jobs in the Cube Designer Job Viewer

If a job fails, you can view and troubleshoot errors.

1. In Job Viewer dialog box, select a job and click **Details** to see the job details.
2. In the Job Details dialog box, select a file from the **Server Error Files** drop-down menu and click **Open** to view and troubleshoot errors.

Clear and Archive Cube Designer Jobs

Clear the Job Viewer or archive job viewer logs periodically to improve performance.

- Press **Clear All** to remove all jobs from the Job Viewer dialog box.
- To selectively remove individual jobs, select one or more jobs and press the Delete key.

  - Use the Shift key to select multiple contiguous jobs.
  - Use the Ctrl key to select multiple non-contiguous jobs.
- To archive the job viewer logs, copy and rename the log file and then delete the original. Job viewer logs are located in `C:\Users\username\AppData\Roaming\Oracle\SmartView\DBX\Jobs`.

  There is a separate log for each user on the client machine.

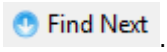
Removing jobs from the Job Viewer dialog box or archiving job viewer logs only affects the client. You can still view all jobs in the web interface.

View Dimension Hierarchies in Cube Designer

You can view dimension hierarchies in the Cube Designer Dimension Hierarchy viewer. To learn more about hierarchies, see Outline Hierarchies.

1. Open the application workbook containing the hierarchy that you want to view.
2. Select the dimension worksheet for the hierarchy that you want to view.
3. On the Cube Designer ribbon, select **Hierarchy Viewer**  .

When you view a hierarchy in Cube Designer, you can perform some actions on the hierarchy. These include:

- To search for a member in the hierarchy, enter a member name in the **Find Next** text box, and click **Find Next** 

.
- To find a member of the dimension in the application workbook dimension worksheet, either double-click a member in the hierarchy or right click on a member in the hierarchy and select **Go To**.

  The corresponding member in the application workbook is highlighted.
- To rename a member:
  1. Right-click a member in the hierarchy and select **Rename**.
  2. Enter the new member name.
  3. Press **Enter.**

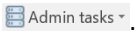
  The corresponding member is renamed wherever found within the Parent and Child columns of the dimension worksheet.
- To set storage for all parents (except members containing formulas or defined as label only) to dynamic calc or to stored:
  1. Select the member in the hierarchy and click **Edit parents** .
  2. On the drop-down menu, select **Set storage to dynamic calc** or **Set storage to stored.**
- To expand or collapse a hierarchy:
  1. Right-click a member in the hierarchy.
  2. Select **Expand All** or **Collapse All.**
- To show or hide aliases, storage, or operators:
  1. Click **Show**.
  2. Click **Alias, Storage,** or **Operator,** to show or hide those items.

Perform Cube Administration Tasks in Cube Designer

You can perform many cube administration tasks in cube designer.

Delete Applications and Cubes in Cube Designer

In Cube Designer, you can delete any application or cube that exists in Essbase. Deleting an application or cube cannot be undone.

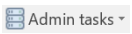
1. In Excel, on the Cube Designer ribbon, select **Admin tasks**  .
2. From the menu, select **Delete Application** or **Delete Cube**.
3. From the Delete Application or Delete Cube dialog box, select the application or cube you want to delete.

Unlock Objects in Cube Designer

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

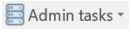
You can view and unlock objects, according to your security role. Service Administrators can unlock any object. Other users can unlock only those objects that they locked.

To unlock an object in Cube Designer:

1. In Excel, on the Cube Designer ribbon, select **Admin tasks** 
2. Select **Unlock Essbase objects**.
3. Enter your login credentials if prompted to do so.
4. Under **Select an application**, select the application containing the object you want to unlock.
5. Under **Select a locked object**, select the object you want to unlock.
6. Click **Unlock**.

View Logs in Cube Designer

In cube designer, you can view the platform log or an application log.

1. In Excel, on the cube designer ribbon, select **Admin tasks** 

2. From the menu, select **View Logs**.
3. Select a log to view:
   - Select **View Platform Log** to view the log for the platform service.
   - Select **View Application Log** to view the log for an individual application.

Manage Applications Using EAS Lite in Cube Designer

You can select in cube designer which applications to manage in Essbase Administration Services (EAS) Lite.

Although the Essbase web interface is the modern administration interface that supports all current platform features, a lite version of Essbase Administration Services is a limited-support option for continued management of your applications if your organization isn't ready to adopt the new interface. This option is available only for Essbase 21c independent installations of Essbase.

See Use Essbase Administration Services Lite for more information about EAS Lite, and to learn how to set applications to EAS managed in cube designer.

Reset a Dimension in Cube Designer

To perform certain dimension editing operations while retaining all data using Cube Designer, you must use the Reset Dimension Incremental Mode on the dimension worksheet in the application workbook.

Using reset dimension clears the members from the dimension and then rebuilds them, retaining the data.

You must update the entire dimension when using reset dimension, otherwise members and data will be lost.

Use Reset Dimension for the following dimension editing operations:

- Re-order members
- Insert a new member in a specific location
- Remove members and maintain the shared members
- Move the members and retain the shared members
- Move parent members and have all the children move with them

Leave Allow Moves set to No, otherwise, you will not be able to build shared members.

Renaming members using this technique is not supported.

To perform a dimension reset in Cube Designer:

1. Open the application workbook.
2. On the Cube Designer ribbon, click **Designer Panel**  .
3. In the designer panel, click **To Sheet**  .
4. In the application workbook, select the dimension you want to reset.
5. In the Designer Panel, in the **Incremental Mode** drop-down menu, select **Reset Dimension**.

6. In the Designer Panel, select **To Sheet**  .
7. In the application workbook, on the dimension sheet, make sure that **Allow Moves** is set to **No**.
8. Save the application workbook.
9. Rebuild the cube. See Create, Load, and Calculate the Cube in Create an Application and Cube in Cube Designer.

Update Cubes Incrementally in Cube Designer

Updating a cube is how you load dimensions and members to a cube outline using a data source and a rule file.

You can also use Essbase to add dimensions and members manually (see Creating and Updating Cubes from Tabular Data).

In an existing cube, you can incrementally update a dimension, or add a new one.

You cannot use Cube Designer to delete dimensions or rename members in an existing cube.

1. In Excel, on the Cube Designer ribbon, select **Build Cube.**  .
2. Choose an **Update Cube** option from the **Build Option** menu.

   When an outline was changed by a dimension build, the database may be restructured. Each of these options specifies how data values are handled during restructures:

   a. **Update Cube - Retain All Data**

      All data values are preserved.

   b. **Update Cube - Retain Input Data**

      All blocks (both upper-and lower-level) that contain loaded data are preserved.

      This option applies only to block storage cubes.

   c. **Update Cube - Retain Leaf Data**

      Only leaf (level 0) values are preserved. If all data required for calculation resides in leaf members, then you should select this option. If selected, then 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.

   d. **Update Cube - Remove All Data**

      All data values are cleared.

      This option applies only to block storage cubes.

- Dimension build definitions are contained within the application workbook and automatically generate the necessary rules files. You do not select a rule file when building dimensions in Cube Designer.
- When making changes to user-defined attributes (UDAs) while updating a cube incrementally using Cube Designer and an application workbook, you must specify all the UDAs in the dimension sheet, both new ones you are adding and existing UDAs in the outline. If you specify some UDAs (such as those you are adding), but not all of them, those that are not specified are deleted.
- When incrementally adding a dimension to an existing cube using an application workbook, the data is automatically mapped to the new top member. There is not a way to

choose a stored member to which to map the existing data. If the new dimension has a top member that is dynamic calc, the data is lost because dynamic members can't store data.

When using an application workbook to add a new dimension in which you want the top member to be dynamic calc, follow these steps:

1. Add the new dimension with the top member as stored.
2. Run a calc script to copy the data from the new top member into another stored member in that dimension.
3. Change the top member to dynamic calc.

Create a Cube from Tabular Data in Cube Designer

This workflow uses two sample tabular data Excel files to demonstrate the concepts of intrinsic and forced-designation headers (hints). See Transform Tabular Data to Cubes.

1. In Excel, on the Cube Designer ribbon, click **Catalog**  .
2. On the Essbase Files dialog box, under **Catalog**, select **Gallery**, then select a sample tabular data file:
   - `Technical > Table Format` > **Sample_Table.xlsx**: Intrinsic headers
   - `Technical > Table Format` > **Unstr_Hints.xlsx**: Forced-designation headers
3. Click **Open.**
4. On the Cube Designer ribbon, select **Transform Data**  .
5. On the Transform Data dialog box, enter an application and cube name, if you want to change the default names that are prepopulated. If the application name already exists, you will not be able to preview data or build a new cube, so you will need to enter a new application name.

   The application name is based on the source file name without the extension and the cube name is based on the worksheet name.

   - Sample_Table.xlsx: Application name is Sample_Table and the cube name is Sales.
   - Unstr_Hints.xlsx: Application name is Unstr_Hints and the cube name is SpendHistory.
6. If you selected Sample_Table.xlsx, do not select **Preview Data**. Skip to step 8 to create the cube.
7. If you selected Unstr_Hints.xlsx, press **Preview Data**. The workbook is sent to Essbase for analysis and the relationships are returned for viewing.
   a. Using the tree view, you can drag and drop members (and their children) to different locations in the tree. This changes the default designations and by doing so, creates different dimension hierarchies, measure hierarchies, and skipped members than the default analysis provided. You can also right click on a member name and designate the property of the member: Generation, Attribute, Alias or UDA.

      In some cases, there is special behavior when you change members from one designation to another:

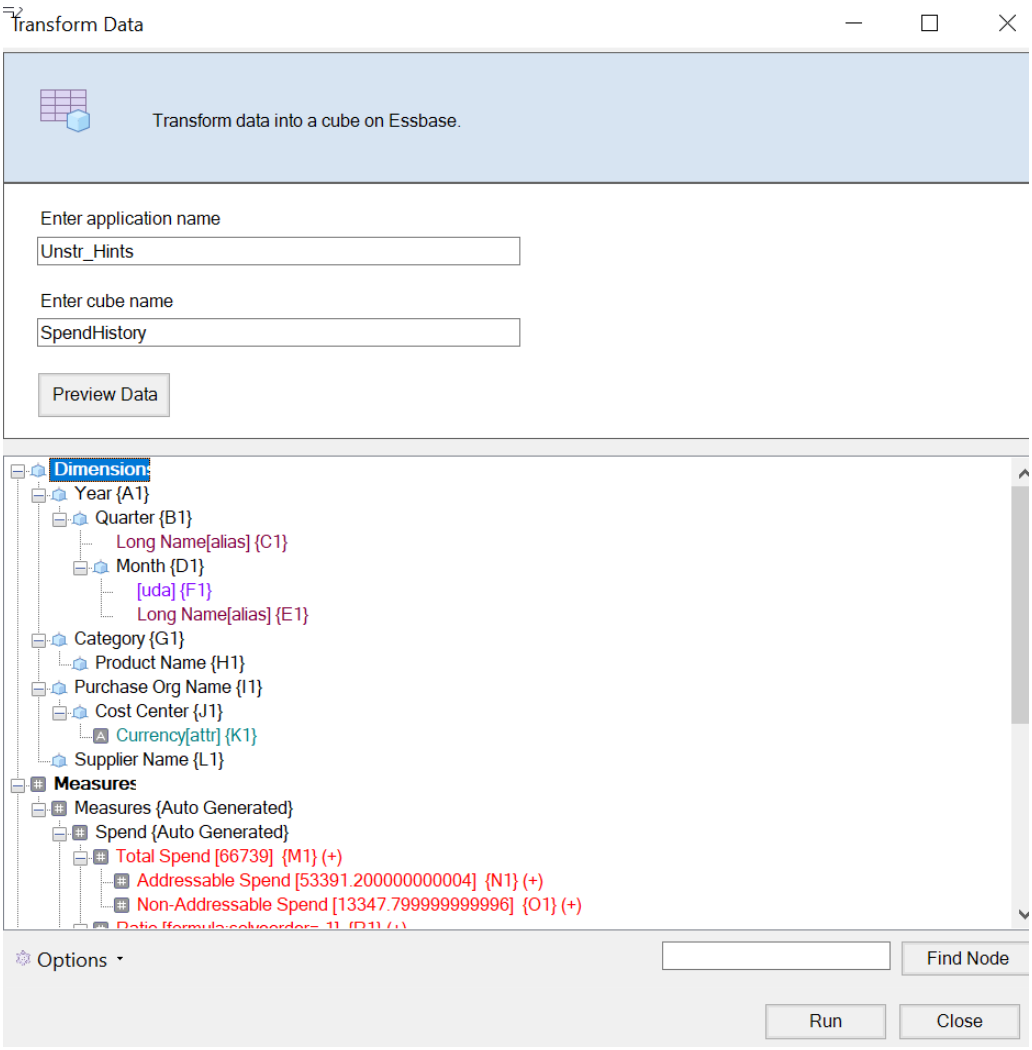
      - When you drag and drop a generation to a measure, all attributes, UDAs, and aliases of the source generation are also moved to measures.
      - When you drag a generation to **Skipped**, all attributes, UDAs, and aliases in that generation are also moved to **Skipped**.

- Dragging and dropping a measure to another designation is only allowed if the measure has no formulas.

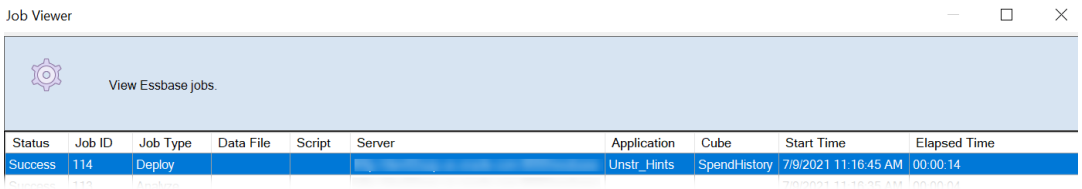
b. If you do not want to save your changes, select **Options** and then select **Reset to Original Header**.

c. If you want to change the cube type and the type of dimensions to be created, before deploying, select **Options**, and then select **Cube Type**. Select **Hybrid BSO** (block storage option) or **ASO** (aggregate storage option).

d. You can guide the analysis to detect two kinds of designations: either measures and hierarchical dimensions, or measures, hierarchical dimensions, and attributes. Define these by selecting **Options**, then **Cube Design**, and then selecting one of the options. After you've made your selection, click **Preview** again.



8. When you are ready to create the cube, click **Run**.
9. When asked if you want to create the cube, click **Yes**.
10. (Optional) When asked if you want to see the cube job status, click **Yes**.



The newly created application and cube are listed on the Applications page in the Essbase web interface and are available in Cube Designer. Now that the cube has been created from the tabular data, you can export the cube to an application workbook.

11. On the Cube Designer ribbon, select **Local** , then select **Export Cube to Application Workbook**.
12. On the Export Cube to Application Workbook dialog box, select the application and cube, and then select **Run** .

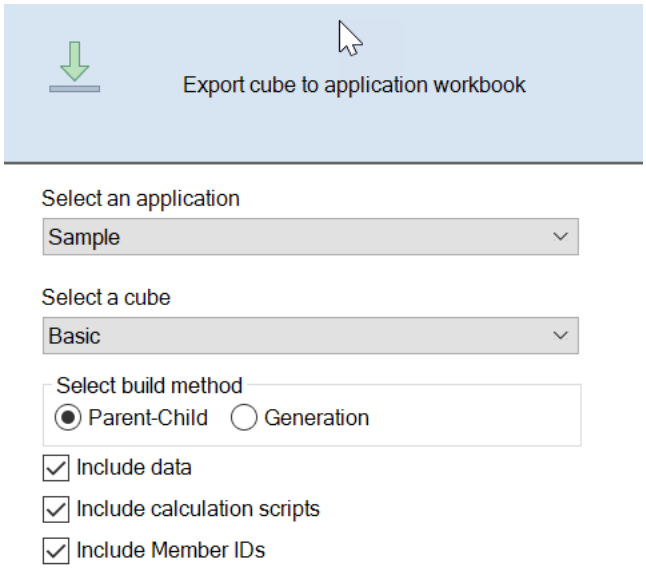
To create a cube using the web interface, see Create and Update a Cube from Tabular Data.

Export Cubes to Application Workbooks in Cube Designer

In Cube Designer, you can export any cube that exists in Essbase.

1. Select the build method, either parent-child or generation format.
2. In Excel, on the Cube Designer ribbon, select **Local** , then select **Export cube to application workbook**.
3. On the Export Cube dialog box, select the application and cube that you want to export.
   - Select **Include Data** if you want input level data included in the application workbook.
     - In block storage cubes, if the size of the data is 400 MB or less, data is exported to the application workbook, on the Data worksheet. If the data size exceeds 400 MB, data is exported to a flat file named *Cubename*`.txt`, which is included in a file named *Cubename*`.zip`. The `.zip` file is created in the specified export directory if the export process is successful.
     - In aggregate storage cubes, regardless of the size of the data, it is always exported to a flat file named *Cubename*`.txt`, which is included in a file named *Cubename*`.zip`. The `.zip` file is created in the specified export directory if the export process is successful.
   - Select **Include Calculation Scripts** if you want calculation scripts in your block storage cube included in the application workbook.

     Aggregate storage cubes do not have calculation scripts.
   - Select **Include Member IDs** if you want to include Member IDs on the dimension sheets in the application workbook.



4. Click **Run.**
5. When the export is completed, click **OK.**

   The application workbook is saved to the local folder location: `C:\Users\`*`username`*`\AppData\Roaming\Oracle\smartview\DBX`. Because it is saved to the local folder location, you can open it using the **Local** icon on the Cube Designer ribbon.

The exported application workbook can be imported to Essbase. See these topics:

- Create a Cube from an Application Workbook
- Create a Cube from a Local Application Workbook in Cube Designer

4

Design and Create Cubes Using Application Workbooks

You can design, create, and modify fully functional cubes using Excel-based application workbooks. You can design the cube within the application workbook, quickly import the workbook to Essbase to create a cube, load data into the cube, and calculate the cube. You can also work with application workbooks in Cube Designer, which is a Smart View extension.

About Application Workbooks

Application workbooks comprise a series of worksheets, which can appear in any order, and define an Essbase cube, including cube settings and dimensional hierarchies. Optionally, you can define data worksheets to be loaded automatically when you create the cube, and calculation worksheets to be executed after you load the data.

There are strict layout and syntax requirements for application workbooks, and there are many validations to ensure that workbook contents are complete and formatted correctly. If the application workbook contents are not correct, then the cube building process will not be successful.

You can modify the worksheets directly in Microsoft Excel or by using the Designer Panel.

In Japanese Excel, if you enter Kanji characters directly on the sheet, the characters are not displayed correctly. Instead, use a text editor to type the Kanji characters and then copy the content into Excel.

Essbase provides application workbook templates for creating block storage and aggregate storage applications and cubes.

- Block Storage Sample (Stored): Block storage application workbook. File name: `Sample_Basic.xlsx.`
- Block Storage Sample (Dynamic): Block storage application workbook. All non-leaf level members are dynamic. File name: `Sample_Basic_Dynamic.xlsx.`
- Block Storage Sample (Scenario): Block storage application workbook with scenarios enabled. All non-leaf level members are dynamic. File name: `Sample_Basic_Scenario.xlsx.`
- Aggregate Storage Sample: Aggregate storage application workbook. File name: `ASO_Sample.xlsx.`
- Aggregate Storage Sample Data: Data for the aggregate storage application workbook. File name: `ASO_Sample_DATA.txt.`

- Tabular Data Sample: Tabular data Excel file. File name: `Sample_Table.xlsx`.

Oracle recommends that you download a sample application workbook and examine the worksheets. See Application Workbooks Reference.

Download a Sample Application Workbook

Using a sample application workbook provided in Essbase, you can quickly create sample applications and cubes. The cubes are highly portable, because they are quickly and easily imported and exported.

1. In the Essbase web interface, on the Home page, click **Files**.
2. Decide if you want to download a sample aggregate storage application workbook, or a sample block storage application workbook:
   a. To download a sample aggregate storage application workbook, under `All Files > Gallery > Applications > Demo Samples > Aggregate Storage`.
   b. To download a sample block storage application workbook, under `All Files > Gallery > Applications > Demo Samples > Block Storage`.
3. From the **Actions** menu to the right of the file you want to download, select **Download.**
4. Optionally, if you download the aggregate storage application workbook, `ASO_Sample.xlsx`, you can also download a data file, `ASO_Sample_Data.txt`.
5. Save the file to a local drive.
6. Open the file and examine the worksheets to understand how you can use the workbook to create an application and cube.

Create a Cube from an Application Workbook

Import an application workbook to create an Essbase cube. Optionally, you can change the application name, select whether to load data and execute calculation scripts, and view dimensions to be created.

1. In the Essbase web interface, on the Home page, click **Import**.
2. In the **Import** dialog box, select **File Browser** to browse to a sample application workbook you previously downloaded.

   You cannot import Excel files that contain spaces in the filename.
3. Your application and cube names are populated based on the names you specified in the application workbook on the Essbase.Cube worksheet.
   - (Optional) You can change the application and cube names on this screen.
   - (Required) If an existing application in Essbase matches the name of the application you are importing, then you must ensure that the cube name is unique. For example, if the name of the application and cube in the Excel workbook is Sample Basic and Essbase already has a Sample Basic cube, then you're prompted to rename the cube.
4. (Optional) Choose a build option and whether to load data and execute calculation scripts.
5. (Optional) Select **View Dimensions,** which allows you to view the mapping of workbook columns to the dimensions to be created.
6. Click **OK**.

   The application is listed on the Applications page.

7. View the outline:

   - On the Home page, navigate to the new application, and select the Actions menu ⋯. Choose **Databases** > *Database Name* > **Launch Outline**.
   - In the Classic Web Interface, expand the application; then click the Actions menu to the right of the cube name, and launch the outline editor.

When you import an application workbook that was created using the command-line 11g Cube Export Utility, some member names might be rejected. See Review Member Names Before you Import an Application Workbook Created by the 11g Cube Export Utility.

If you import an application workbook and then export the cube you created to a new application workbook, the layout of the dimension sheets in the new application workbook might differ from the original, however the new workbook functions the same as the original workbook.

Export a Cube to an Application Workbook

Export an Essbase cube to an Excel application workbook. Choose a build method, and optionally, export data and calculation scripts. The exported application workbook can be imported to create a new cube.

1. Navigate to the **Export to Excel** dialog box.

   In the Redwood Interface:

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

   b. On the **General** page, from the **Actions** menu, choose **Export to Excel**.

   In the Classic Web Interface:

   a. In the Essbase web interface, expand the application that contains the cube that you want to export.

   b. From the Actions menu, to the right of the cube name, select **Export to Excel.**

2. On the Export to Excel dialog box:

   - Select **Export Data** if you want to export the data from the cube. How the data is exported depends on whether the cube is block storage or aggregate storage.
     - In block storage cubes, if the size of the data is 400 MB or less, it is exported to the application workbook, on the Data worksheet. If the data size exceeds 400 MB, data is exported to a flat file named *Cubename*`.txt`, which is included in a file named *Cubename*`.zip` on the **Files** page.
     - In aggregate storage cubes, regardless of the size, data is always exported to a flat file named *Cubename*`.txt`, which is included in a file named *Cubename*`.zip` on the **Files** page.
   - Select a build method, **Generation** or **Parent-Child**.
   - Select **Export Scripts** if you want to export each of the calculation scripts as a separate worksheet within the application workbook.

3. When prompted, save the exported application workbook to your local or network drive or download the exported application workbook and data `.zip` files from the **Files** page.

   File names do not include spaces because files that are imported to Essbase cannot contain spaces in the file name.

If you choose the options to include data, calculation scripts, or both in an export when they do not exist in the cube, the job completes without errors, but no data or scripts are exported.

The exported application workbook can be imported to Essbase. See:

- Create a Cube from an Application Workbook
- Create a Cube from a Local Application Workbook in Cube Designer

Connect to a Cube in Smart View

In Smart View, you can create a private connection using the quick connection method, if you know the URL. The private connection URL is your Essbase login URL with the string `/essbase/smartview` appended to it.

1. From the Smart View ribbon, click **Panel.**
2. From the Smart View panel, click **Home** and then select **Private Connections.**
3. In the text box, enter the login URL ending with `/essbase/smartview`; for example, `https://192.0.2.1:443/essbase/smartview`.
4. Click the connect arrow .
5. On the Login dialog box, enter your Essbase user name and password, then click **Sign In.**

5

Design and Manage Cubes from Tabular Data

You can create a cube from tabular data by extracting fact tables from a relational database into an Excel file and then deploying the cube. You can also export a cube to tabular data.

Topics:

Transform Tabular Data to Cubes

You can create an Essbase cube from tabular data by extracting fact tables from a relational database into an Excel file, and then deploying the cube with Cube Designer or the Essbase web interface.

Essbase detects patterns in the relationships between column headers and data, and uses it to deploy a multidimensional cube. The structures Essbase uses for transforming tabular data into a multidimensional cube include:

- Correlations between columns
- Correlations between column types (such as date, number, and text)
- Header text analysis for common prefixes and business intelligence-related terms (such as cost, price, account)
- Report structure (such as merged cells and empty cells)
- (Optional) Forced-designation headers that are used to explicitly define the shape of a cube and can include formulas to create measures dimensions.
- Measures hierarchies (which can also be generated in Transform Data in Cube Designer).

Sample tabular data Excel files are provided to demonstrate the concepts of intrinsic and forced-designation headers.

When working with tabular data, you should analyze the data before you create a cube from it. Then, after the cube is created, you should determine if the cube outline is the way you want it.

You can create a cube from tabular data in the Essbase web interface or in Cube Designer. See Create and Update a Cube from Tabular Data.

Use Intrinsic Headers to Transform Tabular Data to Cubes

Intrinsic headers use table.column format, which is demonstrated in the `Sample_Table.xlsx` file. In this sample file, the column headers have names such as Units, Discounts, Time.Month, Regions.Region, and Product.Brand.

The transformation process creates this hierarchy:

```
Units
Discounts
Fixed Costs
Variable Costs
Revenue
Time
   Month
   Quarter
Years
Regions
   Region
   Area
   Country
Channel
Product
   Brand
...
```

Use Forced Designation Headers to Transform Tabular Data Into Cubes

When you transform tabular data into an Essbasedatabase, forced-designation headers (also known as hints) can help you specify how tabular data should be handled during the transformation process.

For example, you can force a column to be treated as a measures or an attributes dimension. Most forced-designation headers require a keyword in brackets [ ]. Forced-designation headers are demonstrated in the templates `Unstr_Hints.xlsx` and `Sample_Table.xlsx` (both available in the gallery section of the Essbase file catalog).

Supported forced-designation header formats are listed below:

Table 5-1 Forced-designation Header Formats

Designation	Header Format	Example
Dimension generation	ParentGeneration.CurrentGeneration	Category.Product
Alias	ReferenceGeneration.Generation[alias]	Year.ShortYearForm[alias]
Attribute	ReferenceGeneration.AttributeDimName[attr]	Product.Discounted[attr]
Measures	MeasureName[measure]	Price[measure]
Measure generation	Parent.child[measure]	Measures.profit[measure]
	Top-most parent, if unique, is the account dimension name. If not unique, this member is auto–generated in the account dimension.	profit.cost[measure] cost.price[measure]
Measures formula	MeasureName[=*formula_syntax*;]	profit[="price"-"cost";] profit[="D1"-"E1";] price[=IF ("S1" == #MISSING) "R1"; ELSE "S1"; ENDIF;]

Table 5-1 (Cont.) Forced-designation Header Formats

Designation	Header Format	Example
Measures consolidation	MeasureName[+] : add to parent MeasureName[-] : subtract from parent MeasureName[~] : no consolidation (equivalent to [measure]) The default is no consolidation.	price.shipment[+] Consolidation can be defined only for measure dim
Formula consolidation	FormulaName[+=<formula>] : add to parent FormulaName[-=<formula>] : subtract from parent	profit[+=price-cost] cost.external[+=ExternalWork+ExternalParts]
UDA	ReferenceGeneration[uda]	Product[uda]
Skip The column is not read.	ColumnName[skip]	column[skip]
Recur The last column cell value is used for empty cells Recur can be combined with other forced designations; include a comma separated list of forced designations within a bracket, ColumnName[designationA,recur] .	ColumnName[recur]	Product[recur] Product[uda,recur]

You can specify columns to be measures dimensions and you can use formulas to create measures dimensions with calculated data during the transformation process. The measures and measures formula forced-designation headers are specified with the name for the measures dimension, followed by a keyword or formula that is enclosed in square brackets and appended to the measures dimension name.

You can also consolidate measures and formulas by adding them to, or subtracting from, the parent.

To specify a column to be a measures dimension, in the column header, you enter the name of the measures dimension and then append the keyword [measure]. For example, you can specify the Units and Fixed Costs columns as measures dimensions by using this syntax: Units[measure] and Fixed Costs[measure].

The transformation process creates this hierarchy, with Units, Discounts, Fixed Costs, Variable Costs, and Revenue as measures:

```
Time
   Year
      Quarter
         Month
Regions
   Region
      Area
         Country
...
Product
```

```
   Brand
...
Units
Discounts
Fixed Costs
Variable Costs
Revenue
```

You can create a measure generation hierarchy (parent.child[measure] hierarchy), in a similar way that you create regular dimension generations.

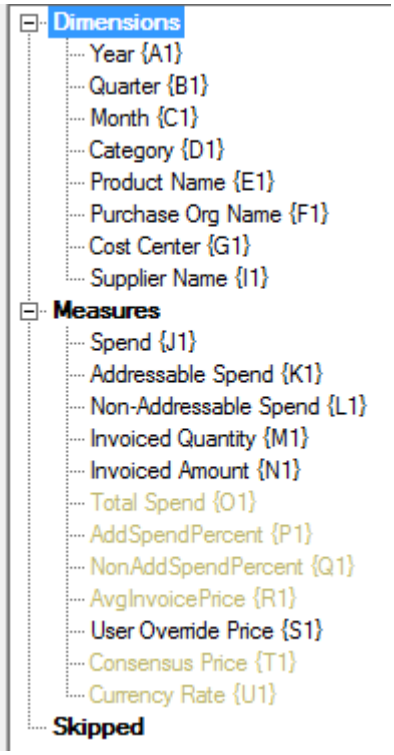
For example, to create a measure hierarchy, you enter Measures.profit[measure], profit.cost[measure] and cost.price[measure], which produces the following hierarchy:

```
Measures
      profit
            cost
                price
```

To create measures dimensions from formulas, in the column header, you enter the name of the measures dimension and then append the formula syntax in brackets [ ]. Within the brackets, start the formula with an equal sign (=) and end the formula with a semicolon (;). The arguments in the formula correspond to column names or cell coordinates, which must be enclosed in quotes. You can use Essbase calculation functions and commands in the formula.

Assume that you have an Excel file named Spend_Formulas.xlsx with tabular data on the SpendHistory worksheet, which has many columns. For example, there are dimensions named Year (column A) and Quarter (column B), and measures dimensions named Spend (column J) and Addressable Spend (column K). These columns have data. Then there are column headers that use formulas to create a measures dimensions. These columns do not have data. For example, to create the Total Spend dimension, the header in column O uses this Essbase formula: Measure.Total Spend[="Addressable Spend" + "Non-Addressable Spend";]. To create the AddSpendPercent dimension, the header in column P uses this Essbase formula: Measure.AddSpendPercent[="Addressable Spend"/"Total Spend";].

The transformation process creates this hierarchy:



The transformation process can also identify measures dimensions when a dimension name is duplicated. Assume that you have a column header that uses this formula, Meas.profit[="a1"-"b1";], which creates the Meas dimension. If, in another column header, you use the Meas dimension name as the top parent, such as Meas.Sales, the Sales dimension is also considered a measures dimension.

Create and Update a Cube from Tabular Data

In this workflow, you're using the sample tabular data Excel file named `Sample_Table.xlsx`, which uses intrinsic column headers. See Transform Tabular Data to Cubes.

1. In the Essbase web interface, on the Home page, click **Files**.
2. On the Files page, open **Gallery**, then **Technical**, and then **Table Format**.
3. From the Actions menu, next to `Sample_Table.xlsx` click **Download.**
4. Save the file to a local drive.
5. To **create** a cube: On the Applications page, click **Import.**
   a. On the **Import** dialog box, click **File Browser** and browse to `Sample_Table.xlsx`.
   b. Open `Sample_Table.xlsx`.

      The application and cube names are pre-populated. The application name is based on the source file name without the extension (in this example, Sample_Table) and the cube name is based on the worksheet name (in this example, Sales).

      - (Optional) You can change the application and cube names on this dialog box.
      - (Required) If an existing application matches the name of the application that you're importing, then you must ensure that the cube name is unique. For example, if there is already an application named Sample_Table with a cube named Sales, then you're prompted to rename the cube.

c. (Optional) Modify the cube type and the type of dimensions to be created.

   In the Redwood Interface, you can:

   - Change the cube type. By default, cubes are set to **Block storage** with the **Hybrid mode** option. You can keep the block storage type but remove the hybrid block storage option, or you can select **Aggregate storage**.
   - Select **Enable sandboxing**, if applicable.
   - Click **Show Transformations** and, on the **Transformations** pane in the Import dialog box, enter names for the dimensions you want to rename.

   In the Classic Web Interface, click **Advanced Options**, and you can:

   - Change the cube type. By default, cubes are set to **BSO** (block storage) with the **Hybrid BSO** option. You can keep the block storage type but remove the hybrid block storage option, or you can select the **ASO** (aggregate storage).
   - Select **Enable Sandboxing**, if applicable.
   - Click **Show Transformations** and, on the **Transformations** pane in the Import dialog box, enter names for the dimensions you want to rename.
   - Change the dimension types.

d. Click **OK**.

   The application and cube are listed on the Applications page.

e. (Optional) View the cube outline:

   On the Applications page, navigate to the new application, and select the Actions menu ••• . Choose **Databases** > *Database Name* > **Launch Outline**.

   In the Classic Web Interface, expand the application. From the Actions menu, to the right of the cube name, launch the outline editor.

6. To **update** a cube with new members or additional data (as an incremental load), from an Excel file: on the Home page, click **Import**.

   The tabular data must have forced designation headers, and the Excel properties must have two custom properties selected: database name and application name. Otherwise, it will use the Excel name as the application name, and sheet name as the cube name.

   a. To do the incremental load, select the file with the incremental data and load it to the cube in the application, which are specified in the Import dialog. On the Import dialog box, click **File Browser,** select the file to add, and click **Open.** A message reminds you that the cube already exists in the application.

   b. In the Classic Web Interface, click **Advanced Options**.

   c. For **Build Option**, select any update cube option, or keep the default, Update Cube — Retain All Data. Click **OK.**

      The cube and corresponding tabular data are updated.

You can't add shared members from tabular data.

Export a Cube to Tabular Data

To facilitate moving and sharing data between Essbase and relational sources, it can be useful to generate flattened output from a block storage (BSO) cube. To accomplish this, you can perform a tabular export.

If you have at least Database Update application permission, you can export a block storage cube from the Essbase web interface into Excel, in tabular format. This exported tabular data is organized into columns with headers that Essbase can use to deploy a new multidimensional cube.

The exported tabular data differs from data exported into an application workbook. Exported tabular data consists of data and metadata, whereas application workbooks are highly structured and contain more information about the cube, such as cube settings and dimensional hierarchies.

The following is an example of CSV output resulting from exporting Sample Basic to table format:

	A	B	C	D	E	F	G	H	I	J	K
1	Product.Cat	Category.Pro	Market.Re	Region.State	Year.Histo	History.Qua	Scenario.G	Margin.Sal	Margin.CO	Profit.Mar	Total
2	100	100-10	East	New York	Qtr1	Jan	Actual	678	271	407	
3	100	100-10	East	New York	Qtr1	Feb	Actual	645	258	387	
4	100	100-10	East	New York	Qtr1	Mar	Actual	675	270	405	

The column headers contain intrinsic relationships, in a logical order, enabling Essbase to detect the relationship patterns needed to build a hierarchy.

The cube you export must meet the following conditions:

- It must be a block storage cube.
- It must not be a scenario enabled cube.
- It must have a measures dimension, and the measures dimension must be dense.
- It must not have asymmetric (ragged) hierarchies. See Hierarchy Shapes.

If you export a cube containing shared members, those members are not added to the exported file.

To export a cube in tabular format in the Redwood Interface:

1. On the Home page, open the application and then open the database (cube).
2. From the **Actions** menu, select **Export to table format**.
3. Select whether to export dynamic blocks, and click **OK**.

To export a cube in tabular format in the Classic Web Interface:

1. In the Essbase web interface, expand the application that contains the cube that you want to export.
2. From the Actions menu, to the right of the cube name, select **Export to Table Format**.
3. Select whether to export dynamic blocks and click **OK**.

The column headers on the exported sheet are of the forced designation headers (hints) type.

You can import the tabular data file to create a new cube. See Transform Tabular Data to Cubes and Use Forced Designation Headers to Transform Tabular Data Into Cubes.

6

Optimize Cubes Using Cube Designer

The Optimize Cube option in Cube Designer provides a set of utilities to help you build and optimize Essbase cubes.

You can use these utilities with hybrid mode cubes, or aggregate storage cubes. Doing so helps you to understand where there are opportunities for optimizing the following processes: building and loading the cube, calculating or aggregating data, running queries, and exporting data.

Create Optimized Hybrid Mode Cubes

The Baseline, Calc Cache, Solve Order, and Data Distribution Optimize Cube utilities help you fine tune your cubes for better performance.

Create optimized hybrid mode cubes using these four Optimize Cube utilities:

Utility	Data Returned
Baseline	Cube performance metrics
Solve Order	Solve order of the members in the cube
Calc Cache	Data to help you choose the best calculator cache value for the cube
Data Distribution	Data to help you choose which dimensions to make sparse and which to make dense

Optimize Baseline Metrics on a Hybrid Mode Cube

The metrics tracked by the Baseline utility in Cube Designer show how the Essbase hybrid mode database is performing. Use these metrics to determine the baseline performance, and then to measure the benefits of the subsequent optimizations that you make.

Before using the Baseline utility, first create an application workbook, including the outline, configuration settings, calculation scripts and queries you want to include in the cube.

When you run the Baseline utility, it builds the cube, loads the selected data files, executes the selected calculation scripts, and runs the queries contained in the application workbook. It is important to have a representative sample of queries from your users.

The Baseline utility creates a dashboard of the application and operational processes, which can help you to design and optimize the cube. As you implement changes and rebuild the

cube, the baseline helps you compare iterations of cube modifications. On the **Essbase.Stats.Baseline** tab of the application workbook, the baseline utility appends new tables with the latest data for each iteration.

Prepare to run the Optimize Cube Baseline Utility on a Hybrid Mode Cube

Complete these tasks before running the Baseline utility:

1. Design and create your application workbook.

   To create an application workbook, you can download a sample application workbook and then modify it to suit your needs. See Explore the Gallery Templates.

2. Clear the query sheets in the application workbook of Smart View metadata:

   a. Go to the Smart View ribbon.

   b. Choose **Sheet Info** and click **Delete**.

   If the query sheets have metadata from a different server, Cube Designer displays a warning and pauses processing until you respond.

3. Modify the **Cube.Settings** worksheet with the following **Application Configuration** settings:

Setting	Value
ASODYNAMICAGGINBSO	FULL
HYBRIDBSOINCALCSCRIPT	NONE
INDEXCACHESIZE	100M
DATACACHESIZE	100M
ASODEFAULTCACHESIZE	100
MAXFORMULACACHESIZE	102400
INPLACEDATAWRITEMARGINPERCENT	20
CALCCACHEDEFAULT	200000
LONGQUERYTIMETHRESHOLD	-1

Run the Optimize Cube Baseline Utility on a Hybrid Mode Cube

The Baseline utility identifies dense and sparse dimensions, data size (PAG and IND file sizes), block size, and the data, index, and calculator cache sizes. Additionally, it provides metrics for the data load, calculation, and query.

To run the Baseline utility:

1. From the Cube Designer ribbon, select **Admin Tasks** > **Optimize Cube**.
2. (Optional) Click **Customize** to choose which baseline operations to run.
   - **Build cube** - Build the cube defined in the application workbook and load the data in the data sheets.
   - **Run calc scripts** - Run the calculation scripts defined in each of the calculation sheets in the application workbook.
     Calculation worksheets run in the order they appear in the application workbook.
     Optimize Cube ignores the **Execute Calc** property on the calculation sheets.

     Only calculation scripts that can be run from Jobs are supported with Optimize Cube. You cannot run calculation scripts that depend on the current Smart View grid context (for example, calculations defined using the @GRIDTUPLES function, or those that use runtime substitution variables defined with <svLaunch> tags).

- **Run queries** - Run the queries on the Query sheets.
- **Export all** - Export all the data in the cube to the cube directory. After the export time and file size are recorded, the export file is deleted automatically.

3. Click **Create Baseline**.

   If you don't have a data sheet in the application workbook, you'll be prompted to select data and load rule files from the catalog. It is a good practice to store the data and rule files in a shared directory in the catalog so that the files won't be lost when you rebuild the cube.

   It will take some time to build the cube.

   Essbase generates the **Essbase.Stats.Baseline** sheet and adds it to the workbook.

4. View the **Essbase.Stats.Baseline** sheet in the application workbook.

   - The first table on the sheet displays the size of the data load files, the number of data load cells, block size, and cache sizes.

Dataload File/s(GB)	140.5 MB
Dataload Cells	15,678,463
Block Size(Bytes)	157,920
Data Cache(MB)	100
Index Cache(MB)	100
Calc Cache(Bytes)	2,500

   - The colors in the baseline table identify the storage type for each dimension:
     - Green - dense dimension
     - Red - sparse dimension with at least one dynamic formula
     - Blue - sparse dimension with aggregations and without all dynamic parents and formulas
     - Gold - other sparse dimension

Baseline			
Dimension	Type	Stored Members	Total Members
Account	DENSE	987	1,515
Period	DENSE	20	142
Entity	SPARSE	12,791	16,133
Currency	SPARSE	2	3
Version	SPARSE	9	9
Initiatives	SPARSE	1	2
Year	SPARSE	13	13
Scenario	SPARSE	11	12
Function	SPARSE	0	35
PG_ATTR	SPARSE	0	163
PL_ATTR	SPARSE	0	134
MG_ATTR	SPARSE	0	10

   - Under **Load and Calc**, the "Script:" rows identify which calculation script takes the longest to complete and thus might need optimizing.

Load and Calc				
Operation	Time (sec)	Blocks	Data (PAG)	Index (IND)
Initial Data Load	87.00	125,063	234,799,155	8,216,576
Script: All	29.00	199,749	641,187,891	16,408,576

- Under **Query**, **Blocks Read**, it shows the amount of data requested by the query. Changing a dynamic dimension to stored reduces that amount.
- Under **Query**, **Formulas**, it shows the number of formulas executed in the query. Review the solve orders of calculated members and make changes to reduce the number of formula executions and improve performance, or consider storing a calculated member containing formulas to reduce the number of formula executions and improve performance.

Query			
Operation	Time (sec)	Blocks Read	Formulas
Query: Test	0.33	275	84

- The last table on the sheet displays the export time and file size.

Export All	
Time (sec)	File Size(MB)
43.00	393.02

Optimize the Solve Order on a Hybrid Mode Cube

The Solve Order utility in Cube Designer gives you a visual representation of the solve order flow used in an Essbase application, which can help you diagnose query performance problems relating to formulas.

To run the Optimize Cube Solve Order utility:

1. From the Cube Designer ribbon, select **Admin Tasks** > **Optimize Cube**.
2. Click **Solve Order**.
3. View the **Essbase.Stats.SolveOrder** sheet of the application workbook.

Use the information in the **Essbase.Stats.SolveOrder** sheet to adjust the solve order to optimize query performance. See Optimize the Cube for Hybrid Mode and Solve Order in Hybrid Mode.

Optimize the Calculator Cache on a Hybrid Mode Cube

The Calc Cache utility in Cube Designer recommends the optimal calculator cache setting for an Essbase hybrid mode, block storage cube.

The calculator cache is applicable for block storage and hybrid mode cubes, but not for ASO cubes. Using the correct calculator cache setting can be an important performance enhancement when calculating entire sparse dimensions in a calculation script. Calculating an entire sparse dimension is a technique for reducing the number of blocks required by a query.

The default value of the calculator cache is 200,000 bytes. The maximum value is 200,000,000 bytes.

The calculator cache should be set to just large enough to contain the sparse dimensions that are calculated in the calculation script. Setting the calculator cache to larger than it needs to be has a negative impact on performance.

To optimize the calculator cache using the Calc Cache utility:

1. In order to reduce the amount of data requested by the query, calculate and store one or more dimensions using a calculation script.

   The best choice is usually the largest dimension.

2. Move that dimension to be the first sparse dimension in the outline.

   The calculator cache algorithm selects the sparse dimensions to place in the cache, beginning with the first sparse dimension.

3. Build the cube without loading data.

   The cube must be built for the Calc Cache utility to work.

4. Run the Calc Cache utility.

   The utility displays the correct cache setting next to each dimension up to 20 MB. Beyond 20 MB, it shows N/A. Generally, settings above a couple of MB are not needed.

   a. From the Cube Designer ribbon, select **Admin Tasks** > **Optimize Cube**.

   b. Click **Calc Cache**.

   c. View the **Essbase.Stats.CalcCache** sheet of the application workbook. You can view the recommended calculator cache settings in the **Essbase.Stats.CalcCache** worksheet, in the **Calc Cache** column.

Dimension	Storage	Total Members	Dependent Parents	Calc Cache (Bytes)
Account	DENSE	1,515		
Period	DENSE	142		
Entity	SPARSE	16,133		2,017
Currency	SPARSE	3		6,050
Version	SPARSE	9		54,449
Initiatives	SPARSE	2		108,898
Year	SPARSE	13		1,415,671
Scenario	SPARSE	12		2,831,342

5. Find the **Calc Cache** setting in the **Essbase.Stats.CalcCache** sheet, next to the sparse dimension(s) you calculated and stored in step 1.

6. If you calculated one dimension in step 1, set the calculator cache default to that value. If you calculated more than one dimension in step 1, choose the highest **Calc Cache** value from among the values you calculated.

   Add this value to the Application Settings section of the **Cube.Settings** worksheet. Alternatively, you can set the value in the application configuration settings in the Essbase web interface. It is a good practice to round up, in order to allow a little more room.

Optimize Data Distribution on a Hybrid Mode Cube

The data distribution utility in Cube Designer helps you better understand the data in an Essbase application, enabling you to make important decisions about how to optimize your cube.

Understanding the data helps you determine the following:

- Which dimensions to make dense and which to make sparse.
  Dense dimensions define the blocks in a block storage application. Ideally, a block should contain dimensions with the most data and represent the predominant query layout for that application. For financial reporting applications, this usually means the Time and Account dimensions should be dense.
- Which dimensions to calculate and store using a calculation script.
  One of the factors that affects query performance is the number of blocks requested by the query. If the number of blocks requested is too high, the query performance suffers. To reduce the number of blocks requested, pre-calculate upper level members of one or more sparse dimensions. First, set the dimension storage attribute of the upper members to a stored attribute (Store or Never Share), and then run a calculation script that aggregates that dimension using either CALC DIM or AGG.
- Which dimensions to use as the task dimension in the FIXPARALLEL command.
  To optimize the calculation script used to aggregate the stored sparse dimensions, use the FIXPARALLEL command. It is important to select the correct task dimensions. A task dimension is the one that determines how the calculation is split into threads and executed in parallel. One or more sparse dimensions should contain the most data to reduce empty tasks, and ideally, that data should be evenly distributed.

To run the Data Distribution utility:

1. From the Cube Designer, select **Admin Tasks** > **Optimize Cube**.
2. Select **Data Distribution**.

   This process can take a long time to run, especially on larger models.
3. View the **Essbase.Stats.DataDist** worksheet.

Dimension	Non-Aggregating	Contains Formulas	Base for attribute	Stored Members	Total Members
Account		X		987	1,515
Period				20	142
Entity			X	12,791	16,133
Currency	X			2	3
Version	X			9	9
Initiatives				1	2
Year	X			13	13
Scenario	X	X		11	12

DataFile	anondata.txt
Dataload Files Size	140.5 MB
Dataload Cells	15,678,463

Blocks	Cells per block
1,103,501	14.21
2,309,337	6.79
265,026	59.16
8,671,759	1.81
10,380,425	1.51
15,678,463	1.00
9,310,087	1.68
13,346,605	1.17

Create Optimized Aggregate Storage Cubes

Create optimized aggregate storage cubes using these two Optimize Cube utilities:

Utility	Data Returned
Baseline	Cube performance metrics
Solve Order	Solve order of the members in the cube

Optimize Baseline Metrics on an Aggregate Storage Cube

The metrics tracked by the Baseline utility in Cube Designer show how the Essbase ASO database is performing. Use these metrics to determine the baseline performance, and then to measure the benefits of the subsequent optimizations that you make.

Before using the Baseline utility, first create an application workbook, including the outline, configuration settings, and queries you want to include in the cube.

When you run the Baseline utility, it builds the cube, loads the selected data files, builds either a default aggregation or a query based aggregation (if enabled in **Customize**), and runs the queries contained in the application workbook. It is important to have a representative sample of queries from your users.

The Baseline utility creates a dashboard of the application and operational processes, which can help you to design and optimize the cube. As you implement changes and rebuild the cube, the baseline helps you compare iterations of cube modifications. On the **Essbase.Stats.Baseline** tab of the application workbook, the baseline utility appends new tables with the latest data for each iteration.

Prepare to Run the Optimize Cube Baseline Utility on an Aggregate Storage Cube

Complete these tasks before running the baseline utility:

1. Design and create your application workbook.

   To create an application workbook, you can download a sample application workbook and then modify it to suit your needs. See Explore the Gallery Templates.

2. Clear the query sheets in the application workbook of Smart View metadata:

   a. Go to the Smart View ribbon.

   b. Choose **Sheet Info** and click **Delete**.

   If the query sheets have metadata from a different server, cube designer displays a warning and pauses processing until you respond.

3. Modify the Cube.Settings worksheet with the following Application Configuration settings:

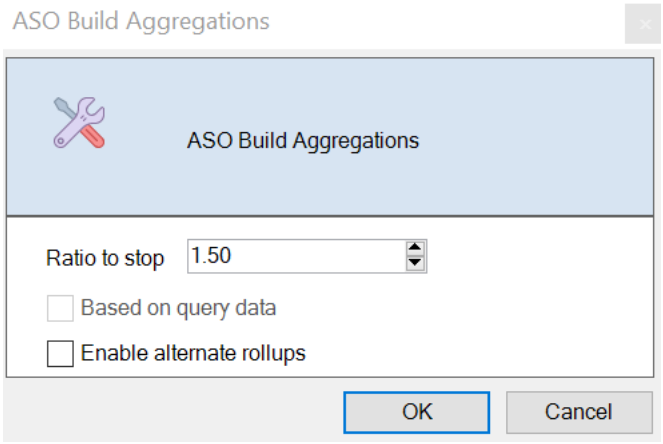
Setting	Value
ASODEFAULTCACHESIZE	100 Designates the default size for the aggregate storage cache. 100 is the default. Start with 100 and adjust if indicated by the output of QUERYTRACE.
LONGQUERYTIMETHRESHOLD	-1 This setting lets you specify the lowest query-time length, in seconds, for which you want to capture statistical information. Oracle recommends setting LONGQUERYTIMETHRESHOLD when using this utility.
QUERYTRACE	-1 Sets a query calculation flow trace to be run and the results to be printed out to a file. Setting QUERYTRACE provides a more in depth analysis.

Run the Optimize Cube Baseline Utility on an Aggregate Storage Cube

On aggregate storage cubes, the Baseline utility identifies dynamic, stored, and multiple hierarchy enabled dimensions. Additionally, it provides metrics for loading data, building aggregations, and performing queries.

To run the Baseline utility:

1. From the **Cube Designer** ribbon, select **Admin Tasks** > **Optimize Cube**.
2. (
   Optional) Click **Customize** to choose which baseline operations to run.
   - **Build cube** - Build the cube defined in the application workbook and load the data in the data sheets.
   - **Build aggregations** – Aggregations are intermediate stored consolidations called aggregate views. Aggregate views store upper-level intersections, which support query performance by avoiding dynamic aggregations on the most commonly queried intersections in the cube. The term aggregation is used to refer to the aggregation process and the set of values stored as a result of the process.
     When you 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. 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.
   - **Run queries** - Run the queries on the Query sheets.
   - **Export all** - Export all the data in the cube to the cube directory. After the export time and file size are recorded, the export file is deleted automatically
3. Click **Create Baseline**.
   - 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 setting this option to a non-zero value 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

- Check or clear 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. The utility runs the queries contained in the workbook first, then it creates the aggregate views based upon those queries.

4. 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.

5. Click **Okay**.

   If you don't have a data sheet in the application workbook, you'll be prompted to select data and rule files from the catalog. It is a good practice to store the data and rule files in a shared directory in the catalog so that the files won't be lost when you rebuild the cube.

   It will take some time to build the cube.

   Essbase generates the **Essbase.Stats.Baseline** sheet and adds it to the workbook.

6. View the **Essbase.Stats.Baseline** sheet in the application workbook.

   - The first table on the sheet displays the number of cells loaded, the default aggregate storage cache size, whether to build aggregations, the ratio to stop, whether it is based on queries in the workbook, and whether alternate rollups are enabled.

Dataload File/s(GB)	12.6 MB
Dataload Cells	1,249,859
ASO Default cache size(MB)	100
Build Aggregation	TRUE
Ratio	1.50
Based on query data	TRUE
Enable alternate rollups	FALSE

- The colors in the **Baseline** table identify the hierarchy type for each dimension.
  - Green – multiple hierarchy dimension
  - Blue – stored hierarchy dimension
  - Gold – dynamic hierarchy dimension

Baseline			
Dimension	**Type**	**Stored Members**	**Total Members**
Measures	Dynamic	7	9
Years	Dynamic	4	5
Time	Multiple	43	46
Transaction Type	Stored	4	4
Payment Type	Stored	5	5
Promotions	Stored	6	6
Age	Stored	13	13
Income Level	Stored	7	7
Products	Multiple	33	38
Stores	Stored	259	259
Geography	Stored	16,904	16,904
Store Manager	Stored	201	201
Square Footage	Stored	8	8
Area Code	Stored	206	206

- Under **Load and Calc**, the rows show load time, input level data size, and aggregate data size for the initial data load and after building aggregations.

Load and Calc			
Operation	**Time (sec)**	**Input-level Data Size (KB)**	**Aggregate Data Size (KB)**
Initial Data Load	65.00	6,688	0
Build Aggregations	5.00	6,688	4,992

- Under **Query**, the **Formulas** column shows the number of formulas executed in the query.
  Review the solve orders of calculated members and make changes to reduce the number of formula executions and improve performance, or consider storing a calculated member containing formulas to reduce the number of formula executions and improve performance.

Query		
Operation	**Time (sec)**	**Formulas**
Before build aggregations		
Query: Test	3.00	3,108
After build aggregations		
Query: Test	0.12	3,108

- The last table on the sheet displays the export time and file size.

Optimize the Solve Order on an Aggregate Storage Cube

The Solve Order utility in Cube Designer gives you a visual representation of the solve order flow used in an Essbase application, which can help you diagnose query performance problems relating to formulas.

To run the Optimize Cube Solve Order utility:

1. From the Cube Designer ribbon, select **Admin Tasks** > **Optimize Cube**.
2. Click **Solve Order**.
3. View the **Essbase.Stats.SolveOrder** sheet of the application workbook.

Use the information in the **Essbase.Stats.SolveOrder** sheet to adjust the solve order to optimize query performance. See Calculation Order.

A

Application Workbooks Reference

Oracle recommends that you download a sample application workbook and examine the worksheets to familiarize yourself on how to design your own application and cube.

- Understand the Essbase.Cube Worksheet
- Understand the Cube.Settings Worksheet
- Understand the Cube.Generations Worksheet
- Understand the Cube.FederatedPartition Worksheet
- Understand the Cube.TypedMeasures Worksheet
- Understand Dimension Worksheets
- Understand Data Worksheets
- Understand Calculation Worksheets
- Understand MDX Worksheets

Also see Download a Sample Application Workbook.

Understand the Essbase.Cube Worksheet

The Essbase.Cube worksheet in an application workbook defines the application and cube name, as well as dimensional information – such as dimension names, types, storage (dense or sparse), and outline order – when you build the cube from Cube Designer or the Essbase web interface.

The following image shows the Essbase.Cube worksheet in a sample application workbook.

Application Name	Sample			
Database Name	Basic			
Version	1.0			
Dimension Definitions				
	Dimension Type	Storage Type	Outline Order	Base Dimension
Year	Time	Dense	1	
Measures	Accounts	Dense	2	
Product	Regular	Sparse	3	
Market	Regular	Sparse	4	
Scenario	Regular	Sparse	5	
Caffeinated	Attribute-Boolean		6	Product
Ounces	Attribute-Numeric		7	Product
Pkg Type	Attribute-Text		8	Product
Population	Attribute-Numeric		9	Market
Intro Date	Attribute-Date		10	Product

Table A-1 Essbase.Cube Worksheet Fields and Values

Property or Field	Valid Values	Description
Application Name	• The application name must not exceed 30 characters. • Do not use spaces. • Application names are not case-sensitive. • The following special characters are not allowed: % $ - { } ( ) ! ~ ` # & @ ^	Enter the name of the application.
Database Name	• The cube name must not exceed 30 characters. • Do not use spaces. • Cube names are not case-sensitive. • The following special characters are not allowed: % $ - { } ( ) ! ~ ` # & @ ^	Enter the name of the cube.
Version	This must be a positive integer.	This is the application workbook version.
Dimension Name	Dimension names cannot be the same as the cube name.	Enter the name of each dimension. There must be at least two dimensions in a cube. For block storage, one dimension must be a dense dimension. Use no more than 1024 characters when naming dimensions, members, or aliases. The following special characters are not allowed: @, ., ,, !, {, }, [,]. /, \, *.
Dimension Type	• Time • Accounts • Regular • Attribute-Boolean • Attribute-Numeric • Attribute-Text • Attribute-Date	Describes the type of dimension. Regular is the Default. Per cube, you can only use one Time and one Accounts dimension type.
Dimension Storage	• Dense • Sparse	Sparse is the default. There must be at least one dense dimension.
Outline Order	This must be a positive integer.	This is the order of the dimension in the outline. Attribute dimensions must be ordered after base dimensions.
Base Dimension	This must be an existing dimension name.	This is the dimension pairing for the attribute dimension.

You can modify the Essbase.Cube worksheet in the Designer Panel. See Work with the Essbase.Cube Worksheet in Cube Designer.

Understand the Cube.Settings Worksheet

The Cube.Settings worksheet in an application workbook defines the Essbase application type (aggregate storage or block storage) and many cube and outline properties such as alias tables, application and database properties, attribute settings, dynamic time series members, and substitution variables.

Each of the five sections in the Cube.Settings worksheet has information about its fields and values, and how to modify those fields and values by using Cube Designer's Designer Panel.

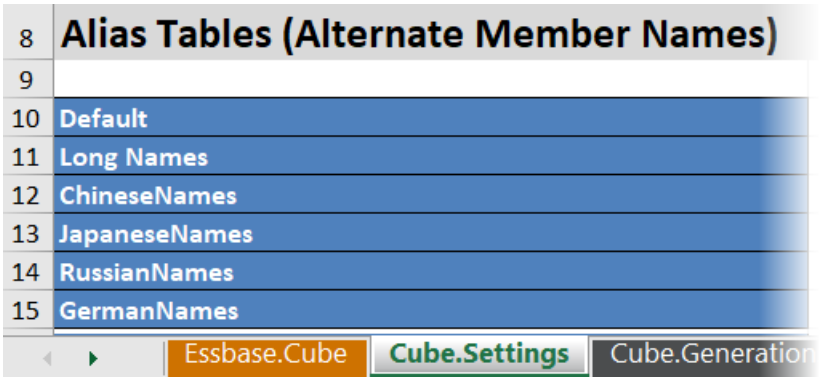
- Understand the Cube.Settings Worksheet: Alias Tables
- Understand the Cube.Settings Worksheet: Properties
- Understand the Cube.Settings Worksheet: Dynamic Time Series
- Understand the Cube.Settings Worksheet: Attribute Settings
- Understand the Cube.Settings Worksheet: Substitution Variables

Understand the Cube.Settings Worksheet: Alias Tables

The Alias Tables section of the Cube.Settings worksheet in an application workbook lists alias tables that need to be created in the Essbase cube, when you build the cube from Cube Designer or the Essbase web interface. It must contain at least the Default row.

Property or Field	Valid Values	Description
Default	Default	Every cube has a table named Default. You can create additional alias tables in the rows following the Default row.
Rows following the default row. These new rows can be created manually, or using the Designer Panel.	Naming conventions for member names apply. See Naming Conventions for Dimensions, Members, and Aliases.	You can set multiple aliases for a member using multiple alias tables.

To define alias tables, add their names on the Cube.Settings worksheet, in the Alias Tables (Alternate Member Names) section. For example, in the workbook for Sample Basic, six alias tables are defined.



An alias table applies to all members in the outline, although you don't have to provide an alias name for every member unless you need it. You may use up to 56 alias tables if you require more than one name for any members in the outline.

When you create a new alias table, it is empty. To make a new set of aliases available to users, you need to populate the alias table with aliases for some of the members.

To define the contents of the alias tables, you must do it per dimension, by adding columns of aliases to the Members section of the dimension worksheets.

bers

PARENT	CHILD	STO	CO	ALIAS.Default	ALIAS.ChineseNames	ALIAS.JapaneseNames	ALIAS.RussianNames	ALIAS.Ge
	Product				商品	商品	Товары	Produkt
Product	100			Colas	可樂類	コーラ類	Колы	Cola Get
100	100-10			Cola	可樂	コーラ	Кола	Cola
100	100-20			Diet Cola	健怡可樂(低熱量可樂	ダイエットコーラ	Диетическая кола	Cola Ligh
100	100-30			Caffeine Free Cola	無咖啡因可樂	コーラ　カフェイン	Кола без коффеина	Koffeinfr
Product	200			Root Beer	麥根沙士	ルートビール	Корнеплодные напит	Kohlensä
200	200-10			Old Fashioned	傳統的	オールドファッショ	Старинный напиток	Orangen
200	200-20			Diet Root Beer	健怡(低熱量)麥根沙士	ダイエットルールビ	Дает. корнеплодный	Zitronen
200	200-30			Sasparilla	黑松沙士	サスパリラ	Саспарилла	Mineralv
200	200-40			Birch Beer	Birch Beer	バーチビール	Березовый напиток	Mineralv
Product	300			Cream Soda	奶精汽水	クリームソーダ	Крем-сода	Milchget
300	300-10			Dark Cream	Dark Cream	ダーククリーム	Темная крем-сода	Schoko
300	300-20			Vanilla Cream	香草奶精	バニラクリーム	Ванильная крем-сода	Vanille
300	300-30			Diet Cream	健怡奶精	ダイエットクリーム	Диетическая крем-со,	Light
Product	400			Fruit Soda	水果汽水	フルーツソーダ	Фруктовые газирован	Fruchtsa
400	400-10			Grape	葡萄	ぶどう	Виноградный напито	Grapefru
400	400-20			Orange	橘子	オレンジ	Апельсиновый напит	Orange
400	400-30			Strawberry	草莓	いちご	Клубничный напиток	Erdbeere

Essbase.Cube | Cube.Settings | Cube.Generations | Dim.Year | Dim.Measures | Dim.Product | Dim.Market | Dim.S

The columns must be named in the format ALIAS.<AliasTableName>.

Notice that there is no ALIAS.Long Names column. Though the Cube.Settings worksheet indicates that an alias table named Long Names will be built, this alias table will be empty if no aliases are defined for it on any dimension worksheets.

See also Setting Aliases.

Understand the Cube.Settings Worksheet: Properties

The Properties section of the Cube.Settings worksheet in an application workbook determine properties related to the Essbase application, database and outline, when you build the cube from Cube Designer or the Essbase web interface.

The following table shows the fields, values and descriptions for the Properties section on the Cube.Settings worksheet:

Table A-2 Properties Section of the Cube.Settings Worksheet

Property or Field	Valid Values	Description
Application Type	• ASO • BSO	This is an application property. Defines whether the cubes in the application use aggregate storage (ASO) or block storage (BSO).

Table A-2 (Cont.) Properties Section of the Cube.Settings Worksheet

Property or Field	Valid Values	Description
Outline Type	• Unique • Duplicate	This is a database property. • Unique: member names in the outline must be unique. • Duplicate: Duplicate member names are permitted in the outline.
Aggregate missing values	• Yes • No	This is a database property. Defines whether missing (#MISSING) values are aggregated during a cube calculation.
Create blocks on equations	• Yes • No	This is a database property. If you enter Yes, then when you assign a nonconstant value to a member combination for which no data block exists, a data block is created. Entering Yes can produce a very large cube. Sometimes, new blocks are not desired; for example, when they contain no other values. In large databases, creation and processing of unneeded blocks can increase processing time and storage requirements. For more specific control, you can use the SET CREATEBLOCKONEQ calculation command within a calculation script to control creation of blocks at the time the command is encountered in the script. See the SET CREATEBLOCKONEQ calculation command.
Two-Pass calculation	• Yes • No	This is a database property. If you enter Yes, then after a default calculation, members that are tagged as two-pass are recalculated, overwriting the aggregation results from the first calculation pass. The two-pass tag is effective on members of the dimension tagged as Accounts and on Dynamic Calc and Dynamic Calc and Store members of any dimension.

Table A-2 (Cont.) Properties Section of the Cube.Settings Worksheet

Property or Field	Valid Values	Description
Date Format	The following date formats are supported in application workbooks: • month dd yyyy • mm/dd/yy • yyyy-mm-dd • mon dd yyyy • yy.mm.dd • dd.mm.yy • mm-dd-yy • mm/dd/yyyy • dd/mm/yy • dd-mm-yy • dd Month yy • Month dd, yy • mon dd, yy • dd mon yyyy • yy/mm/dd • dd Month yyyy • yyyy/mm/dd • dd-mon-yy • dd mon yy	This is a database property. Date measures enable cell values in the form of a formatted date. The date values are stored internally as numeric values, although you load them into Essbase as formatted date strings. When queried, date measures are displayed according to the selected date format.
Implied Share	• Force On • Force Off	If you select Force On, the parent is treated as an implied share when it has only one child or when it has only one child that consolidates to the parent. If you select Force Off, Essbase never uses Implied Share. This is the default behavior.
Scenario Sandboxes	• 0 • A positive integer less than 1000.	This value defines whether the cube contains a sandbox dimension for creating scenarios of the data, and the number of sandbox members within the sandbox dimension. A value of 0 indicates no sandbox dimension.

You can modify the Properties section on the Cube.Settings worksheet in Cube Designer's Designer Panel. See Work with the Cube.Settings Worksheet: Properties in Cube Designer.

Understand the Cube.Settings Worksheet: Dynamic Time Series

The Dynamic Time Series section of the Cube.Settings worksheet in an application workbook determine properties related to Dynamic Time Series (DTS) members you can define when you build the cube from Cube Designer or the Essbase web interface.

Table A-3 Dynamic Time Series Section of the Cube.Settings Worksheet

Property or Field	Valid Values	Description
H-T-D	Integer value representing the generation number	History to date
Y-T-D	Integer value representing the generation number	Year to date
S-T-D	Integer value representing the generation number	Season to date
P-T-D	Integer value representing the generation number	Period to date
Q-T-D	Integer value representing the generation number	Quarter to date
M-T-D	Integer value representing the generation number	Month to date
W-T-D	Integer value representing the generation number	Week to date
D-T-D	Integer value representing the generation number	Day to date

You can modify the Dynamic Time Series section on the Cube.Settings worksheet in Cube Designer's Designer Panel. See Work with the Cube.Settings Worksheet: Dynamic Time Series in Cube Designer.

See Using Dynamic Time Series Members.

Understand the Cube.Settings Worksheet: Attribute Settings

The Attribute Settings section of the Cube.Settings worksheet in an application workbook determines properties related to attribute dimensions and members, when you build the cube from Cube Designer or the Essbase web interface.

The following table shows the fields, values and descriptions for the Attribute Settings section on the Cube.Settings worksheet:

Table A-4 Attribute Settings

Property or Field	Valid Values	Description
Dimension Name	Default: Attributes Calculation	To avoid duplicating names in an outline, you can change the names of members of the attribute calculations dimension. Regardless of the name that you use for a member, the function of the member remains the same. For example, the Sum member always calculates a sum, no matter what you name it. See Changing the Member Names of the Attribute Calculations Dimension.

Table A-4 (Cont.) Attribute Settings

Property or Field	Valid Values	Description
Sum Member	Default: Sum	This is a member of the attribute calculations dimension. The name to use when requesting sum data.
Count Member	Default: Count	This is a member of the attribute calculations dimension. The name to use when requesting count data.
Minimum Member	Default: Min	This is a member of the attribute calculations dimension. The name to use when requesting minimum data.
Maximum Member	Default: Max	This is a member of the attribute calculations dimension. The name to use when requesting maximum data.
Average Member	Default: Avg	This is a member of the attribute calculations dimension. The name to use when requesting average data.
False Member	Default: False	The initial Boolean member names in a cube are set as True and False. See Setting Boolean Attribute Member Names.
True Member	Default: True	The initial Boolean member names in a cube are set as True and False. See Setting Boolean Attribute Member Names.
Prefix/Suffix Value	• None • Dimension • Parent • Grandparent • Ancestors	See Setting Prefix and Suffix Formats for Member Names of Attribute Dimensions.
Prefix/Suffix Format	• Prefix • Suffix	You can define unique names by attaching a prefix or suffix to member names in Boolean, date, and numeric attribute dimensions in the outline. See Setting Prefix and Suffix Formats for Member Names of Attribute Dimensions.

Table A-4 (Cont.) Attribute Settings

Property or Field	Valid Values	Description
Prefix/Suffix Separator	• _ Underscore • | Pipe • ^ Carat	You can define unique names by attaching a prefix or suffix to member names in Boolean, date, and numeric attribute dimensions in the outline. Select a separator (to place between the prefix or suffix and the original name): underscore (_), pipe (|) , or caret.(^).
Attribute Numeric Ranges	• Tops of ranges • Bottoms of ranges	See Setting Up Member Names Representing Ranges of Values.
Date Member	• Month First (mm-dd-yyyy) • Day First (dd-mm-yyyy)	You can change the format of members of date attribute dimensions. See Changing the Member Names in Date Attribute Dimensions.

You can modify the Attribute Settings section on the Cube.Settings worksheet in Cube Designer's Designer Panel. See Work with the Cube.Settings Worksheet: Attribute Settings in Cube Designer.

Understand the Cube.Settings Worksheet: Substitution Variables

The Cube.Settings worksheet in an application workbook enables you to add or modify Essbase substitution variables.

Substitution variables act as global placeholders for information that changes regularly. You create the variable and a corresponding string value, and the value can then be changed at any time.

A substitution variable can be used in a query or calculation script to represent a member in the outline. By default, there are no substitution variables defined for a cube.

There is not an option to add substitution variables in Cube Designer's Designer Panel, however you can add them directly in the application workbook.

1. On the Cube.Settings worksheet, in the Substitution Variables section, create a new row.
2. Enter the variable name in column A and its value in column B, enclosing the value in quotation marks if it represents a member name.
   Example:

   ```
   CurrMonth "Jan"
   ```

See Using Substitution Variables.

Understand the Cube.Generations Worksheet

The Cube.Generations worksheet in an Essbase application workbook is used for naming generations in the Essbase outline when you build the cube from Cube Designer or the Essbase web interface.

Cube.Generations Worksheets

The term "generation" indicates the distance of a member from the root of the dimension. Using a generation number, you can determine the location of members within the database tree. All members in a database that are the same number of branches from their root have the same generation number. The dimension is generation 1, its children are generation 2, and so on.

You can create names for generations in an outline, such as a word or phrase that describes the generation. For example, you might create a generation name called Cities for all cities in the outline.

You can also use generation names in calculation scripts wherever you need to specify a list of generation numbers. For example, you could limit a calculation in a calculation script to all members in a specific generation.

You can specify only one name per generation. The specified name must be unique; that is, it cannot duplicate a generation, level, or member name or an alias or conventional alias.

If you build a cube using an application workbook that has names reserved for Dynamic Time Series on the Cube.Generations sheet for the time dimension, Essbase automatically creates and enables the corresponding Dynamic Time Series member.

> **Note**
>
> The Dimension section of the Cube.Generations worksheet changes if you change the dimension worksheet (Dim.*dimname*) by adding or deleting members in such a way that the number of generations in the dimension is changed. If you make changes to the dimension worksheet by adding or deleting members, you should always press the **Update Generation Worksheet** button on the **Dimensions** tab of Cube Designer's Designer Panel as part of the editing process.

Cube.Generations Worksheet Format

The following image shows a Cube.Generations worksheet in a sample application workbook.

Generation Properties

Dimension Name Year

Generation Number	Generation Name	Unique
1	History	Yes
2	Quarter	Yes
3		Yes

Dimension Name Product

Generation Number	Generation Name	Unique
2	Category	Yes
3	Line	No

Dimension Name Market

Generation Number	Generation Name	Unique
1	Market1	Yes
2	m2	No
3	m3	No

Table A-5 Fields and Valid Values in Generation Worksheets

Property or Field	Valid Values	Description
Dimension Name	For dimension naming restrictions, see Naming Conventions for Dimensions, Members, and Aliasesfor naming restrictions.	The dimension name.
Generation Number	A generation number, 1 or greater.	A root branch of the tree is generation 1. Generation numbers increase as you count from the root toward the leaf member.
Generation Name	You can define only one name for each generation. When you name generations, follow the same naming rules as for members. See Naming Conventions for Dimensions, Members, and Aliases.	The generation name. You can use this field to create or change generation names. Enter the generation name and then build or update the cube using the application workbook. See Update Cubes Incrementally in Cube Designer.

Table A-5 (Cont.) Fields and Valid Values in Generation Worksheets

Property or Field	Valid Values	Description
Unique	• Yes • No	For duplicate member name outlines, enter Yes to require unique member names within the associated generation.

Understand the Cube.FederatedPartition Worksheet

The Cube.FederatedPartition worksheet in an Essbase application workbook defines a federated partition to Autonomous AI Lakehouse, including the connection name, fact table name, pivot dimension name, and storage management type. It also includes mappings for dimensions and the pivot dimension.

Connection Name	multicube	
Fact Table	SHAREDFACT	
Pivot Dimension	Year	
Storage Management	User	
Dimension Map		
Dimension	Fact Column	
Measures	Accounts	
Product	Product	
Market	Market	
Scenario	Scenario	
Pivot Dimension Map		
Member	Generation Number	Fact Column
Jan	3	Jan
Feb	3	Feb
Mar	3	Mar
Apr	3	Apr
May	3	May
Jun	3	Jun
Jul	3	Jul
Aug	3	Aug
Sep	3	Sep
Oct	3	Oct
Nov	3	Nov
Dec	3	Dec

Property or Field	Valid Values	Description
Connection Name	The name of the connection	The global level connection to Autonomous AI Lakehouse that was previously created by an administrator as shown in Create a Connection for Federated Cubes.
Fact Table	The name of the fact table	The name of the fact table in Autonomous AI Lakehouse that stores numeric values and keys.

Property or Field	Valid Values	Description
Pivot Dimension	The name of the pivot dimension	The name of the pivot dimension you decided to use from the Essbase outline, during the Identify the Pivot Dimension process.
Storage Management	• User • Essbase	With User managed, you create and manage the fact table. With Essbase managed, you let Essbase create and manage the fact table.

Dimension Map table fields and valid values

Property or Field	Valid Values	Description
Dimension	Dimension names	Essbase dimension member names that map to fact table column names.
Fact Column	Fact table column names	Fact table column names that are mapped to Essbase dimension member names.

Pivot Dimension Map table fields and valid values

Property or Field	Valid Values	Description
Member	Member names from the pivot dimension	Names of Essbase members in the pivot dimension.
Generation Number	Numerals matching the generation number	The generation number.
Fact Column	Fact table column names.	The fact table column names that are mapped to Essbase members in the pivot dimension.

> **Note**
>
> For the pivot dimension, a one-to-one relationship between the fact table column and Essbase member names is enforced.

You can create federated partition worksheets in Cube Designer's Designer Panel. See Create a Federated Partition using Cube Designer.

To learn more about federated partitions, see Federated Cubes: Integrate Essbase with Autonomous AI Database.

Understand the Cube.TypedMeasures Worksheet

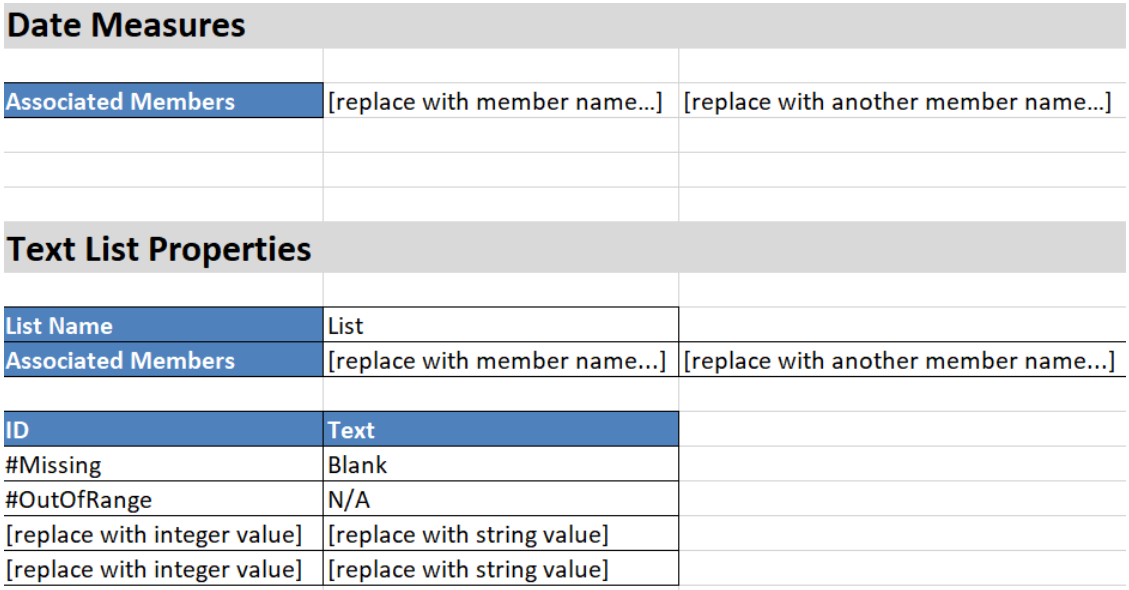
The Cube.TypedMeasures worksheet in an application workbook defines date measures and text lists when you build the cube from Cube Designer or the Essbase web interface.

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

- Date measures are tagged as "date" in the Accounts dimension. Date measures enable cell values in the form of a formatted date. The ability to process dates in the measures dimension can be useful for types of analysis that are difficult to represent using the Time dimension.
- Text lists are used to work with text measures, which are tagged as "text" in the Accounts dimension. They enable cell values to contain one of an enumerated list of text labels. These labels are defined, at the outline level, using a mapping artifact called a text list. Storage and analysis of textual content can be useful when a cell needs to have one of a finite list of textual values; for example, a product may be sold in 5 different colors. The color is a text measure whose value must be one of the 5 colors. The colors are a set of text strings mapped to corresponding numeric IDs.

Date measures and text list mappings are contained in tables in the Cube.TypedMeasures worksheet.

The following image shows the Cube.TypedMeasures worksheet in a sample application workbook.



Date Measures

Associated Members	[replace with member name...]	[replace with another member name...]

Text List Properties

List Name	List	
Associated Members	[replace with member name...]	[replace with another member name...]

ID	Text
#Missing	Blank
#OutOfRange	N/A
[replace with integer value]	[replace with string value]
[replace with integer value]	[replace with string value]

Date Measures table fields and values:

Property or Field	Valid Values	Description
Associated Members	Members from the dimension tagged as “Accounts.”	The Associated Members row contains member names from the Accounts dimension.

Text List Properties table fields and values:

Property or Field	Valid Values	Description
List Name	Must not exceed 80 characters.	A text list must start with a list name followed by its value in the adjacent cell.

Property or Field	Valid Values	Description
Associated Members	Existing member names.	Member names added in adjacent cells. Multiple members can be added in adjacent cells to the right.
ID	The first two values under ID are #Missing and #OutOfRange. These two values must exist in every text list table. The other IDs must be integers.	Each ID, including the #Missing, #OUTOFRANGE and numeric values, must map to a text value. The first two IDs, #Missing and #OUTOFRANGE, are for handling cases where the textual data is invalid or empty. For example, if you try to load an unmapped value such as “Average” to a text measure, the cell value would not be updated, and would display as #Missing in a subsequent query. If you load a numeric cell value that is unmapped, the subsequent query would return N/A.
Text	Up to 80 characters.	The text column contains the text values for each text measure. Each text value must map to an integer in the ID column. Any text value that does not map to an integer in the text list is considered by Essbase to be invalid.

- Working with Typed Measures
- Work with Typed Measures Worksheets in Cube Designer
- Performing Database Operations on Text and Date Measures

Understand Dimension Worksheets

Dimension worksheets in an application workbook define the dimensions in the Essbase cube, when you build the cube from Cube Designer or the Essbase web interface.

Application workbooks contain one dimension worksheet for each of the dimensions listed in the Essbase.Cube worksheet. The name of each dimension worksheet is Dim.*dimname*; for example, the Year dimension worksheet is called Dim.Year. Dimension names can contain up to 1024 characters, but long dimension names (longer than 31 characters, including "Dim.") are truncated in the dimension sheet name.

Dimension worksheets use load rule syntax. For example, an X in the Storage column means that the data value is not stored.

The following image shows a dimension worksheet in a sample application workbook.

Dimension Name **Year**

Definitions

File Name	Dim_Year	Delimiter	,
Rule Name	Year	Header Rows to Skip	0
Build Method	PARENT-CHILD	Allow Moves	No
Incremental Mode	Merge		

Members

Columns	PARENT	CHILD	STORAGE	ALIAS.ChineseNames	IGNORE	ALIAS.JapaneseNames
		Year	X	年	1	年
	Year	Qtr1	X	第一季	2	第一四半期
	Qtr1	Jan		一月	3	１月
	Qtr1	Feb		二月	4	２月
	Qtr1	Mar		三月	5	３月
	Year	Qtr2	X	第二季	6	第二四半期
	Qtr2	Apr		四月	7	４月
	Qtr2	May		五月	8	５月
	Qtr2	Jun		六月	9	６月
	Year	Qtr3	X	第三季	10	第三四半期
	Qtr3	Jul		七月	11	７月
	Qtr3	Aug		八月	12	８月
	Qtr3	Sep		九月	13	９月

Table A-6 Fields and Valid Values in Dimension Worksheets

Property or Field	Valid Values	Description
Dimension Name	The name of the dimension. Do not change the dimension name in this field.	Any dimension or attribute dimension in the outline. Defined on the Essbase.Cube worksheet. Use no more than 1024 characters when naming dimensions, members, or aliases. The following special characters are not allowed: @, ., ,, !, {, }, [,]. /, \, *.
File Name	A valid string. The file name cannot be longer than thirty characters.	The build process creates a data file with a `.txt` extension in Essbase for every data worksheet in the application workbook. You can give them meaningful names so that they are easily recognizable if they need to be used again.
Rule Name	A valid string. See Name and Related Artifact Limits. The rule name cannot be longer than thirty characters.	The build process creates a rule file with a `.rul` extension in Essbase for every dimension worksheet in the workbook. You can give them meaningful names so that they are easily recognizable if they need to be used again.

Table A-6 (Cont.) Fields and Valid Values in Dimension Worksheets

Property or Field	Valid Values	Description
Build Method	• PARENT-CHILD • GENERATION	In Designer Panel, you can build a cube with either build method, but you cannot edit a cube built using the Generation build method using the panel, and you cannot view hierarchies using Cube Designer Dimension Hierarchy viewer.
Incremental Mode	• Merge • Remove Unspecified • Reset Dimension	Incremental dimension builds enable you to update existing dimensions with new members. Merge is the default. This option adds the new members to the dimension while retaining the existing members. Remove Unspecified removes members that are not specified in the source file. Reset Dimension clears the members from the dimension and then rebuilds them, retaining the data. See Reset a Dimension in Cube Designer.
Delimiter	The values can be a tab, a space, or any single character except “.	This value must be updated directly in the Excel sheet. It cannot be updated using the Cube Designer interface.
Header Rows to Skip	A positive number or zero. Zero is the default.	The number of header rows to skip when performing a data load or dimension build. This value must be updated directly in the Excel sheet. It cannot be updated using the Cube Designer interface.
Allow Moves	• Yes • No	Within a dimension, moves members and their children to new parents; recognizes primary members and matches them with the data source; not available for duplicate member outlines. This value must be updated directly in the Excel sheet. It cannot be updated using the Cube Designer interface.
Data Source	A valid Data Source name.	This value is used to retrieve data from the source defined in the data source definition. This value must be updated directly in the application workbook. It can't be updated using the Cube Designer interface.

Table A-6 (Cont.) Fields and Valid Values in Dimension Worksheets

Property or Field	Valid Values	Description
Member ID	Any unique key	Used to uniquely identify a member in an outline.
Prototype	• Member ID of the prototype member • Qualified member name of the prototype member	Indicates the prototype member (member ID or qualified member name) for shared members.
Storage Type	• N Never allow data sharing. • O Tag as label only (store no data). • S Set member as stored (non dynamic calc and not label only). • X Create as dynamic calc.	Uses load rules member property codes. See Using the Data Source to Work with Member Properties.
Consolidation Operator	• + • - • * • / • % • ~ • ^	• + (add) • - (subtract) • * (multiply) • / (divide) • % (percent) • ~ (no operation) • ^ (never consolidate)
IGNORE	Ignore	Data in a column with the heading, IGNORE is ignored during data loads and dimension builds. This value must be updated directly in the Excel sheet. It cannot be updated using the Cube Designer interface.
Two-Pass Calculation	• Yes • No	If you enter Yes, after a default calculation, then members that are tagged as two-pass are recalculated. The two-pass tag is effective on members of the dimension tagged as Accounts and on Dynamic Calc and Dynamic Calc and Store members of any dimension. Two-pass calculation applies only to block storage outlines.

Table A-6 (Cont.) Fields and Valid Values in Dimension Worksheets

Property or Field	Valid Values	Description
Solve Order	Any number, 0 to 127	You can set solve order for dimensions or members, or you can use the default solve order. The minimum solve order you can set is 0, and the maximum is 127. A higher solve order means the member is calculated later; for example, a member with a solve order of 1 is solved before a member with a solve order of 2. Members that are not assigned a solve order are assigned the solve order of their dimension.
Time Balance	• A Treat as an average time balance item (Applies to accounts dimensions only). • F Treat as the first time balance item (Applies to accounts dimensions only). • L Treat as the last time balance item (Applies to accounts dimensions only).	Uses load rules member property codes. See Using the Data Source to Work with Member Properties. Time balance properties provide instructions about how to calculate data in the Accounts dimension. See Setting Time Balance Properties.
Skip Value	• B Exclude data values of zero or #MISSING in the time balance (applies to accounts dimensions only). • M Exclude data values of #MISSING from the time balance (applies to accounts dimensions only). • Z Exclude data values of zero from the time balance (applies to accounts dimensions only).	Uses load rules member property codes. See Using the Data Source to Work with Member Properties. If you set the time balance as first, last, or average, then set the Skip property to indicate what to do when missing values or values of 0 are encountered. See Setting Skip Properties.
Expense Reporting	E	Treat as an expense item (applies to accounts dimensions only)
Comment	Any string	Enter a comment.
Formula	Valid formula syntax.	Enter a member formula.

Table A-6 (Cont.) Fields and Valid Values in Dimension Worksheets

Property or Field	Valid Values	Description
User Defined Attribute	Attribute names, such as specific colors or sizes	Defined attribute names used to aid in the analysis of the data. When making changes to user-defined attributes (UDAs) while updating a cube incrementally using Cube Designer and an application workbook, you must specify all the UDAs in the dimension sheet, both new ones you are adding and existing UDAs in the outline. If you specify some UDAs (such as those you are adding), but not all of them, those that are not specified are deleted.
Number of UDAs	A numeral	The number of UDAs for this member.
Available Alias Tables	Naming conventions for member names apply. See Naming Conventions for Dimensions, Members, and Aliases.	ALIAS.*table_name* After the column heading with ALIAS.*table_name*, the column is populated with the aliases for the cube.

You can modify dimension worksheets in Cube Designer's Designer Panel. See Work with Dimension Worksheets in Cube Designer.

See Working with Rules Files.

Understand Data Worksheets

Data worksheets in an application workbook define data to be loaded into Essbase when you build the cube from Cube Designer or the Essbase web interface. You can include one or more data worksheets in an application workbook.

Data Worksheets

The name of each data worksheet is Data.*name*. For example, for values for the eastern region, the data worksheet might be called Data.East. The *name* can be anything you choose. You can choose meaningful names so that you can recognize them if you need to use them again.

> **Note**
>
> Multiple data worksheets are allowed in an application workbook, but they must share the exact same column layout.

Data Worksheet Format

When loading data, a member from every dimension must be defined before a data value. Therefore, the data worksheet places all but one dimension under the column headings titled,

Dimension.*dimension_name*. One dimension is selected as the Measures dimension and members from that dimension must be added manually under the remaining column headings titled Measure.*member_name*. Only place members that will contain data in the columns titled Measure.*member_name*.

When scenarios are enabled, cubes have a hidden dimension called sandbox. The sandbox dimension, named Dimension.*sandbox*, is the first column in the data worksheet. It contains a member called base that you must define when loading data.

The following image shows a data worksheet in a sample application workbook.

Definitions

File Name	Cube_Basic	Sign Flip Dimensic	Measures
Rule Name	Basic	Sign Flip UDA	Flip
Data Load Option	Replace		
Delimiter	,		
Header Rows to Skip	0		

Data

Columns	Dimension.Product	Dimension.Market	Dimension.Year	Dimension.Scenario	IGNORE	Measure.Sales	Measure.COGS
	100-10	New York	Jan	Actual	1	678	271
	100-10	New York	Feb	Actual	2	645	258
	100-10	New York	Mar	Actual	3	675	270
	100-10	New York	Apr	Actual	4	712	284
	100-10	New York	May	Actual	5	756	302
	100-10	New York	Jun	Actual	6	890	356
	100-10	New York	Jul	Actual	7	912	364
	100-10	New York	Aug	Actual	8	910	364
	100-10	New York	Sep	Actual	9	790	316
	100-10	New York	Oct	Actual	10	650	260
	100-10	New York	Nov	Actual	11	623	249
	100-10	New York	Dec	Actual	12	699	279
	100-10	New York	Jan	Budget	13	640	260

The following table describes the settings on the data.*name* worksheets in application workbooks.

Property or Field	Valid Values	Description
File Name	A valid string. See Name and Related Artifact Limits.	The build process creates a data file with a `.txt` extension in the Essbase web interface for every data worksheet in the application workbook. You can give them meaningful names so that they are easily recognizable if they need to be used again.
Rule Name	A valid string. See Name and Related Artifact Limits.	The build process creates a rule file with a `.rul` extension in the Essbase web interface for every dimension worksheet in the workbook. You can give them meaningful names so that they are easily recognizable if they need to be used again.

Property or Field	Valid Values	Description
Data Load Option	• Add • Subtract • Replace	If you enter Replace, then the existing values of the database are overwritten with the values of the data source. You can also use incoming data values to add to or subtract from existing database values. For example, if you load weekly values, then you can add them to create monthly values in the database.
Delimiter	The values can be a tab, a space, or any single character except “. • Tab • Space • Any single character except “	This value must be updated directly in the Excel sheet. It cannot be updated using the Cube Designer interface.
Header Rows to Skip	A positive number or zero.	The number of header rows to skip when performing a data load or dimension build. This value must be updated directly in the Excel sheet. It cannot be updated using the Cube Designer interface.
Sign Flip Dimension	*Dimension name*	Reverses the values of data fields by flipping their signs. Enter the name of the dimension in the Sign Flip Dimension field, and enter the selected UDA within the specified dimension in the Sign Flip UDA field. This value must be updated directly in the Excel sheet. It cannot be updated using the Cube Designer interface.
Sign Flip UDA	• Flip • Blank	Reverses the values of data fields by flipping their signs. Enter the name of the dimension in the Sign Flip Dimension field, and enter the selected UDA within the specified dimension in the Sign Flip UDA field. This value must be updated directly in the Excel sheet. It cannot be updated using the Cube Designer interface.
Ignore column header	Ignore	Data in a column with the heading, IGNORE is ignored during data loads and dimension builds. This value must be updated directly in the Excel sheet. It cannot be updated using the Cube Designer interface.

Property or Field	Valid Values	Description
Data Source	A valid Data Source name.	This value is used to retrieve data from the source defined in the Data Source definition. This value must be updated directly in the application workbook. It can't be updated using the Cube Designer interface.

Data Operations

When you load data, values can replace, add to, or subtract from existing data values in the cube. You indicate which of these options to use in the **Data Load Option** field on the data worksheet.

- **Replace**: Overwrites cube values with the data source values. Replace is the default.
- **Add**: Adds data source values to the cube values. For example, if you load weekly data values, you can add them to create cumulative data values in the cube.
- **Subtract**: Subtracts data source values from the database values. For example, to track available budget by week, you can subtract weekly data expenditures from the previous week's budget values.

Rule Files

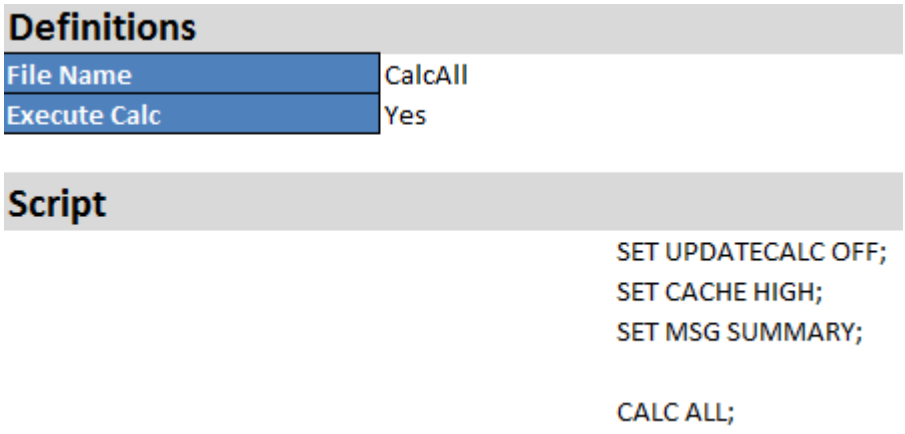
When you build a cube, data files and data load rule files are created in the Essbase web interface. Those files can then be used later if you want to load data to a cube. Data files are named with the file name specified in the definitions area of the data sheet and a `.txt` extension. For example, `cube_basic.txt`. Rule files are named with the file name specified in the definitions area of the data sheet and a `.rul` extension. For example, `cube_basic.rul`.

You can modify data worksheets in Cube Designer's Designer Panel. See Work with Data Worksheets in Cube Designer.

Understand Calculation Worksheets

The contents of the calculation worksheet are used to create a calculation script in Essbase when you build the cube from Cube Designer or the Essbase web interface. You can have one or more calculation worksheets in an application workbook.

The following image shows a calculation worksheet in a sample application workbook.



Definitions	
File Name	CalcAll
Execute Calc	Yes

Script

```
SET UPDATECALC OFF;
SET CACHE HIGH;
SET MSG SUMMARY;

CALC ALL;
```

Within the calculation worksheet, the calculation script begins in cell C6.

The name of each calculation worksheet is Calc.*scriptname*, for example, for the sample CalcAll calculation script, the calculation worksheet is called Calc.calcall.

The calculation script uses the file name specified in the definitions area of the calculation sheet and a has a `.csc` extension. For example, *`filename`*`.csc`.

You can execute the calculation script when you build the cube in Cube Designer, if you select **Run Calculation Sheets Contained within Workbook** in the Build Cube dialog box. If you do not want to execute the calculation, do not select this option.

The calculation scripts are executed in the order they appear in the application workbook.

Property or Field	Valid Values	Description
File Name	A valid calculation script file name. *`filename`*`.csc`.	The File Name defines the calculation script name. The calculation script created in Essbase when the cube is created is the File Name with a `.csc` extension.
Execute Calc	• Yes • No	If you enter Yes, then the calculation is executed at the time you build the cube. If you enter No, then the calculation is not executed right away. In either case, each calculation worksheet creates a calculation script in Essbase, using the specified file name with a `.csc` extension. That way, any of the calculations can be executed at a later time.

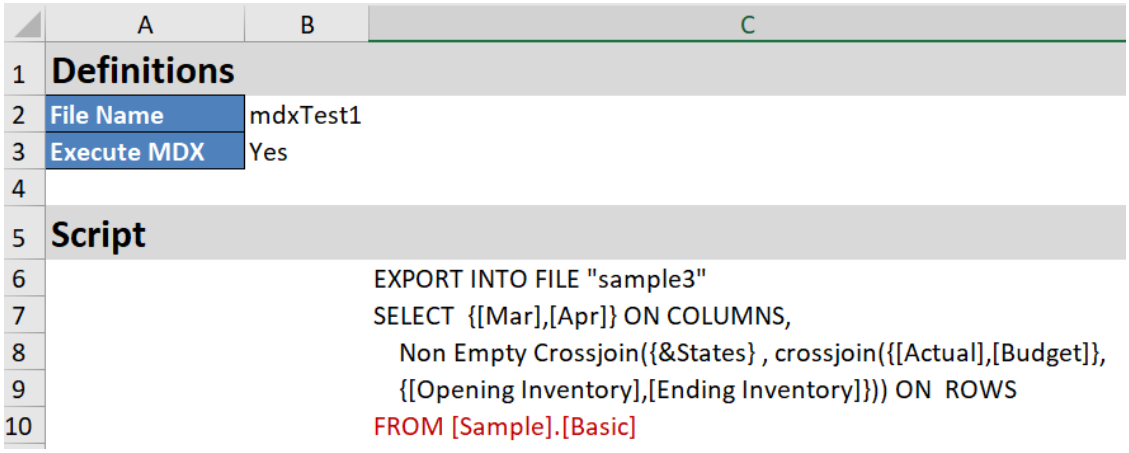
You can modify calculation worksheets in Cube Designer's Designer Panel. See Work with Calculation Worksheets in Cube Designer.

Understand MDX Worksheets

You can have one or more MDX Insert worksheets in an Essbase application workbook. With these worksheets, you can create corresponding MDX files in the cube, and you can optionally execute the MDX at the time you build the cube.

- To execute the MDX when you build the cube, indicate **Yes** in the **Execute MDX** field on the MDX worksheet in the application workbook.
- To execute the MDX after the cube is created, run the MDX script from the Essbase web interface, from **Jobs**.

The following image shows an MDX Insert worksheet in a sample application workbook.



	A	B	C
1	**Definitions**		
2	File Name	mdxTest1	
3	Execute MDX	Yes	
4			
5	**Script**		
6			EXPORT INTO FILE "sample3"
7			SELECT {[Mar],[Apr]} ON COLUMNS,
8			Non Empty Crossjoin({&States} , crossjoin({[Actual],[Budget]},
9			{[Opening Inventory],[Ending Inventory]})) ON ROWS
10			FROM [Sample].[Basic]

The name of each MDX worksheet is MDX.*scriptname*, for example, for the mdxTest1 MDX script, the MDX worksheet is called MDX.mdxTest1.

The contents of the MDX worksheet are used to create an MDX Insert script in the cube. The MDX script uses the file name specified in the definitions area of the MDX sheet and has an `.mdx` extension. For example, *filename*`.mdx`.

Property or Field	Valid Values	Description
File Name	A valid MDX script file name.	The **File Name** field defines the MDX script name. The MDX script is created in Essbase when the cube is created. The script name in Essbase is the file name with an `.mdx` extension.
Execute MDX	• Yes • No	If you enter **Yes**, then the MDX script is executed at the time you build the cube. If you enter **No**, then the MDX script is not executed right away. In either case, each MDX worksheet creates an MDX script in Essbase, using the specified file name with a `.mdx` extension. That way, any of the MDX scripts can be executed at a later time.

You can create and delete MDX worksheets in the Designer Panel. See Work with MDX Worksheets in Cube Designer.

To learn more about MDX Insert, see Insert and Export Data with MDX and MDX Insert Specification.