The next generation CONTROL®

version 10.5

Power BI Integration

Table of Contents

Overview

This document describes the integration of CONTROL® with Microsoft's Power BI data visualization product and with Excel's pivot table function.

This integration also provides a gateway to other popular business intelligence and reporting tools, as most of these products work well with the supported data structures.

CONTROL® applications store their data in relational database tables, typically Microsoft SQL Server or Oracle. These tables are organized into a star schema structure (fact and dimension tables). This organization supports CONTROL®'s extensive dynamic computational capabilities, reporting options, and ability to handle multiple hierarchies. However, the organization can be cumbersome for other reporting tools. CONTROL®'s Power BI integration provides a bridge from the star schema structure to one more convenient for visualization tools. At the heart of the integration are CONTROL® **Power Pivot Models** that are used to manage the export of data and meta-data from CONTROL® for consumption by business intelligence and reporting tools.

The integration works in stages:

- **Stage 1** (Required): Creating relational tables or relational views of data and meta-data based on one or more CONTROL® computational or source data views.
- **Stage 2** (Optional): Creating a SQL Server Analysis Services Tabular model based on those relational tables or relational views.

The output of either stage can be used by Power BI, Excel, or other products.

Because of the completeness of the Analysis Services semantic data model, creating a tabular model will yield a richer user experience with less effort for those products that interface to it.

Power Pivot models

CONTROL® **Power Pivot** models specify the data you want exported, and how you want it exported. Several options allow flexibility over how the data will appear to your targeted product.

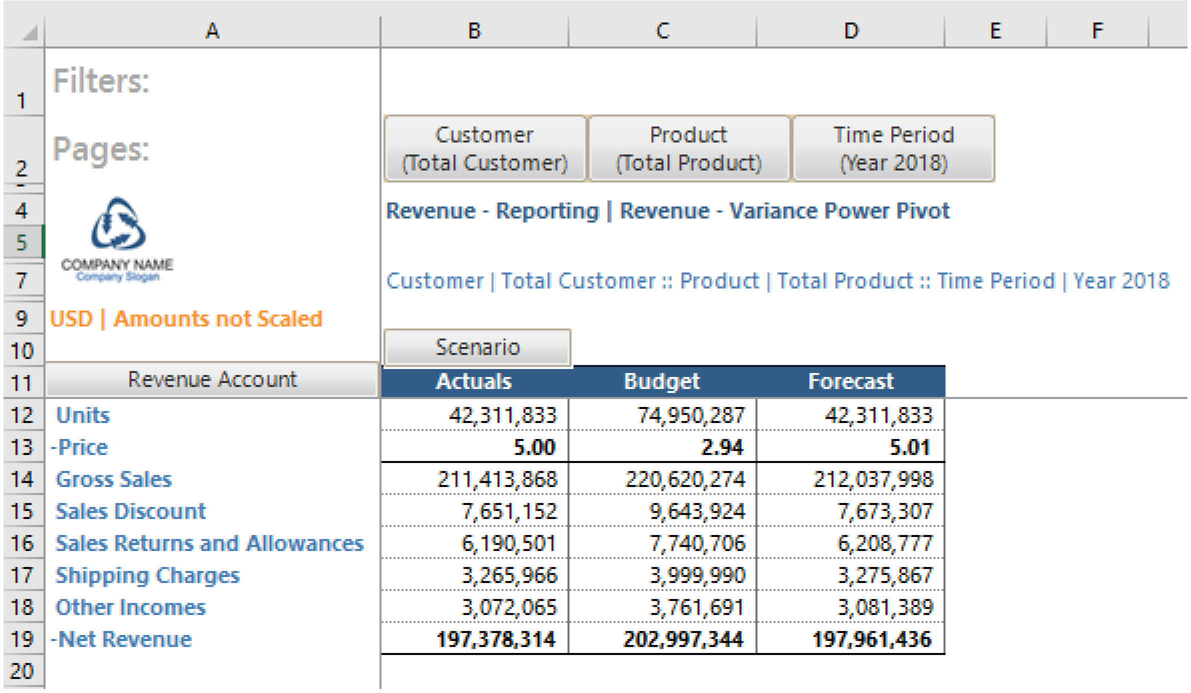
Here is how it works:

- Create a **Power Pivot** model based on a computational or source data model, and optionally on a view from that model. This specifies the data and levels of summarization that will be available.
- If you do not specify a pre-built view, you can create a view with the filters, branches, etc. that you want within the **Power Pivot** model's edit book. You can also include custom dimensions.

- Specify the export options to integrate the CONTROL® data with external systems, or use the default values, which in most cases works fine.
- Use the **Export** button on the **CONTROL® Model** ribbon to affect the transfer.
- You can also create a script to run the export as a scheduled process or to run it from your application menu system.
- Access the data in your favorite BI tool!

Create a Power Pivot model

Here is a simple example of creating a **Power Pivot** model using a view on the **Revenue** model:



Filters:

Pages: Customer (Total Customer) | Product (Total Product) | Time Period (Year 2018)

Revenue - Reporting | Revenue - Variance Power Pivot

Customer | Total Customer :: Product | Total Product :: Time Period | Year 2018

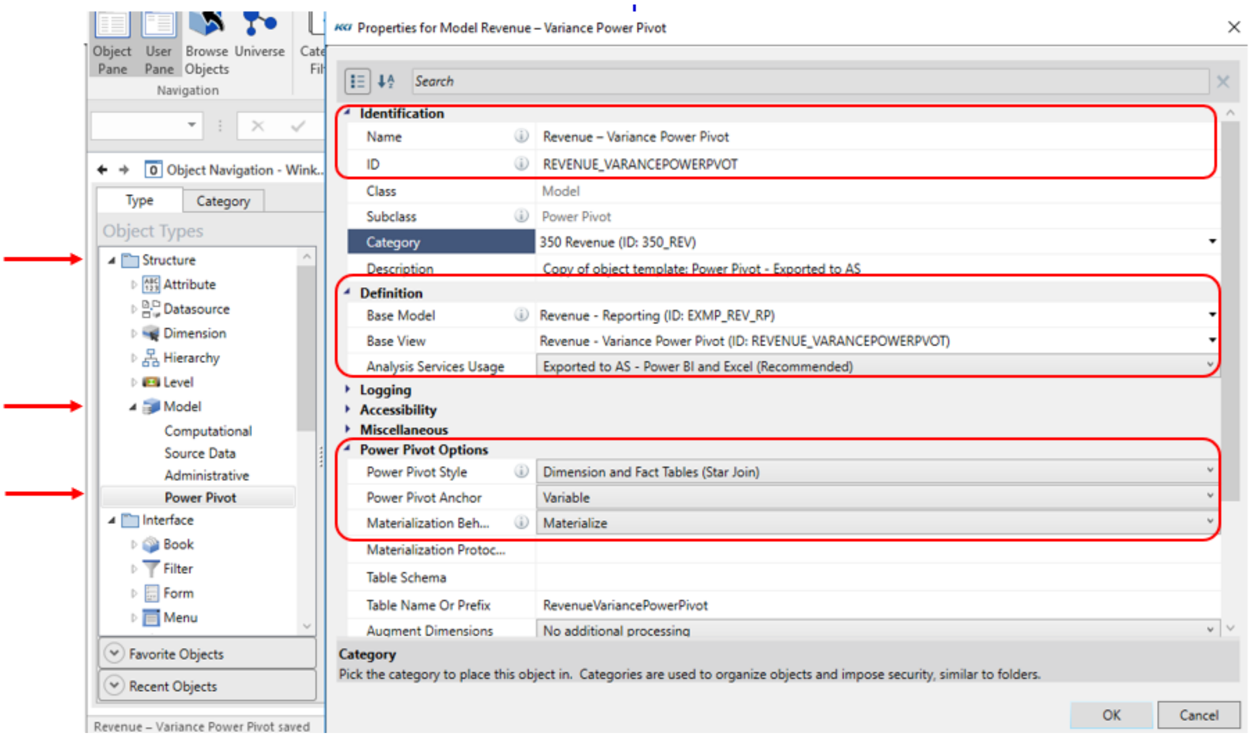
USD | Amounts not Scaled

Scenario

Revenue Account	Actuals	Budget	Forecast
Units	42,311,833	74,950,287	42,311,833
-Price	**5.00**	**2.94**	**5.01**
Gross Sales	211,413,868	220,620,274	212,037,998
Sales Discount	7,651,152	9,643,924	7,673,307
Sales Returns and Allowances	6,190,501	7,740,706	6,208,777
Shipping Charges	3,265,966	3,999,990	3,275,867
Other Incomes	3,072,065	3,761,691	3,081,389
-Net Revenue	**197,378,314**	**202,997,344**	**197,961,436**

1. In the **Object Navigation** task pane, in the **Structure** group, click **Model**, right click **Power Pivot**, and then click **New**.
2. In the **Model Templates** pane of the **New Model** dialog, choose **Power Pivot – Exported to AS**.
3. In the **Properties** pane, make the following entries and then click **OK and Save**.
   - **Name**: Revenue – Variance Power Pivot
   - **ID**: REVENUE_VARANCEPOWERPVOT
   - **Base Model**: Revenue - Reporting
   - **Base View**: Revenue – Variance Power Pivot
   - **Analysis Services Usage**: Exported to AS – Power BI and Excel (Recommended)
   - **Power Pivot Style**: Dimension and Fact Tables (Star Join)

- **Power Pivot Anchor**: Variable
- **Materialization Behavior**: Materialize



Properties for Model Revenue – Variance Power Pivot

Identification

Name	Revenue – Variance Power Pivot
ID	REVENUE_VARANCEPOWERPVOT
Class	Model
Subclass	Power Pivot
Category	350 Revenue (ID: 350_REV)
Description	Copy of object template: Power Pivot - Exported to AS

Definition

Base Model	Revenue - Reporting (ID: EXMP_REV_RP)
Base View	Revenue - Variance Power Pivot (ID: REVENUE_VARANCEPOWERPVOT)
Analysis Services Usage	Exported to AS - Power BI and Excel (Recommended)

Logging

Accessibility

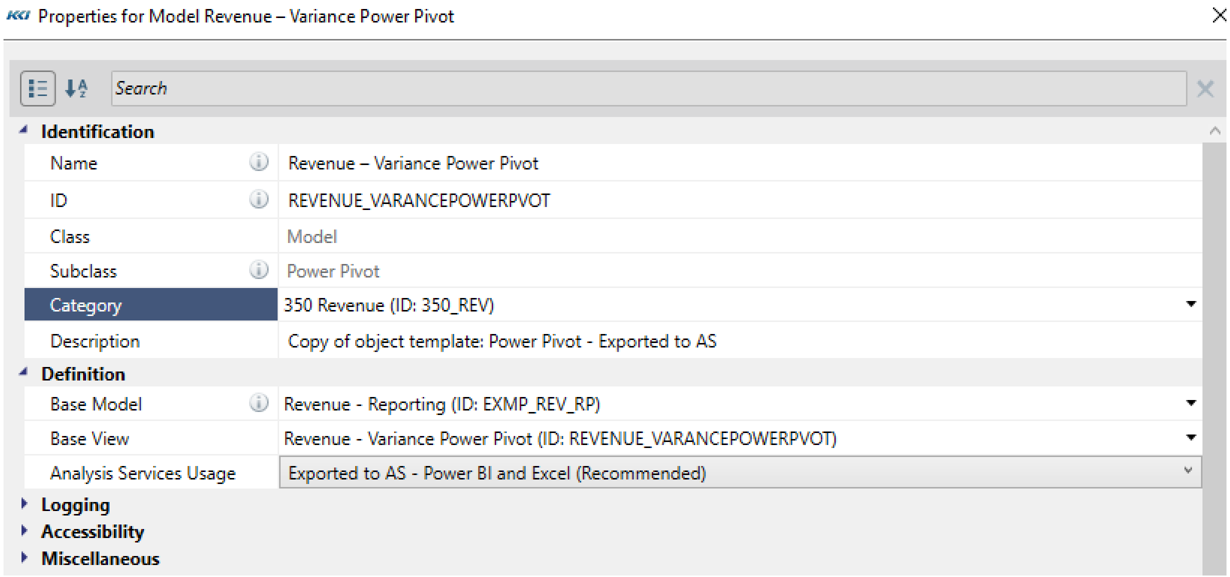
Miscellaneous

Power Pivot Options

Power Pivot Style	Dimension and Fact Tables (Star Join)
Power Pivot Anchor	Variable
Materialization Beh...	Materialize
Materialization Protoc...	
Table Schema	
Table Name Or Prefix	RevenueVariancePowerPivot
Augment Dimensions	No additional processing

Category
Pick the category to place this object in. Categories are used to organize objects and impose security, similar to folders.

Power Pivot model properties



Properties for Model Revenue – Variance Power Pivot

Identification

Name	Revenue – Variance Power Pivot
ID	REVENUE_VARANCEPOWERPVOT
Class	Model
Subclass	Power Pivot
Category	350 Revenue (ID: 350_REV)
Description	Copy of object template: Power Pivot - Exported to AS

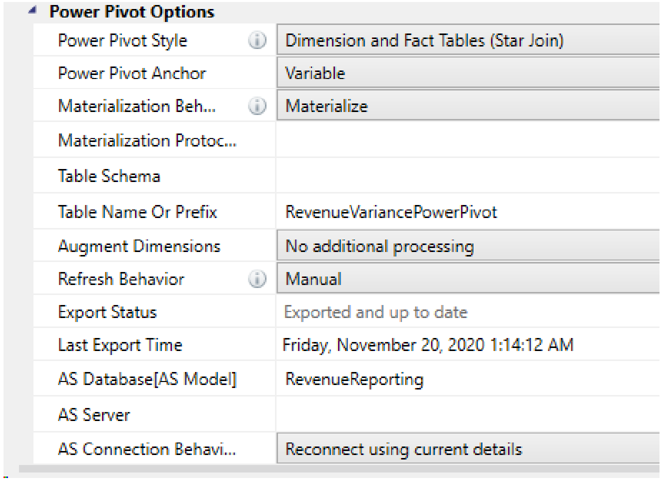
Definition

Base Model	Revenue - Reporting (ID: EXMP_REV_RP)
Base View	Revenue - Variance Power Pivot (ID: REVENUE_VARANCEPOWERPVOT)
Analysis Services Usage	Exported to AS - Power BI and Excel (Recommended)

Logging

Accessibility

Miscellaneous



Power Pivot Options

Power Pivot Style	Dimension and Fact Tables (Star Join)
Power Pivot Anchor	Variable
Materialization Beh...	Materialize
Materialization Protoc...	
Table Schema	
Table Name Or Prefix	RevenueVariancePowerPivot
Augment Dimensions	No additional processing
Refresh Behavior	Manual
Export Status	Exported and up to date
Last Export Time	Friday, November 20, 2020 1:14:12 AM
AS Database[AS Model]	RevenueReporting
AS Server	
AS Connection Behavi...	Reconnect using current details

The table below lists the most important properties:

Property	Definition
Name	The name of the **Power Pivot** model. (If exported to AS, this will be the name of the model in AS, unless you supply an override)
ID	The ID of the **Power Pivot** model. (If exported to AS, this will be the name of the database in AS, unless you supply an override)
Base Model	The name of the computational or source data model used as the source of the data.
Base View	The name of the view that filters the specific data from the base model.
Analysis Services Usage	Defines whether to export the model to an Analysis Services (AS) Tabular model, or if it is a reference to an externally created and maintained AS model. **Not Exported to AS** option creates relational tables or relational views of data and meta-data based on the **Power Pivot** model. For **Power Pivot** models exported to Analysis Services this property is set to **Exported to AS – Power BI and Excel (Recommended)** by default**.** The export creates measures for each data column in the fact table and hence the AS model works with both Excel and Power BI. The **Export to AS – Power BI** creates data columns in the fact table but no measures. The AS model works fine with Power BI. However, pivot

table in Excel does not produce right results with this model as it expects a measure for data columns. This option is available only to support existing models and should not be used for **Power Pivot** models going forward.

Managed Externally in AS indicates that the **Power Pivot** model is a reference to an externally created and maintained AS model.

Note: Using this property is recommended only if you have the required AS infrastructure and knowledge; however, it is not required for use with Microsoft's Power BI products, or for other BI applications.

Power Pivot Style

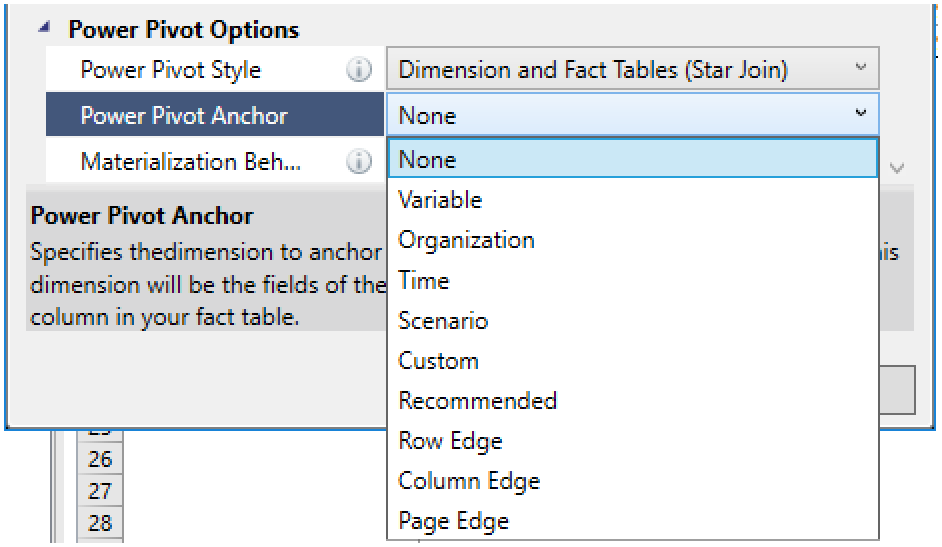
Specify whether you want to see all the data and meta-data in a single table or organized into dimension and fact tables.

A single table will be simpler to navigate for an inexperienced user and results in a small number of data fields.

Dimension and fact tables are a more efficient and more logically organized structure, and this style is recommended, particularly for use in AS and Power BI.

Power Pivot Anchor

Specify the dimension to anchor the **Power Pivot** model on – the members of this dimension will be the fields of the fact table.



Power Pivot Options

Power Pivot Style: Dimension and Fact Tables (Star Join)

Power Pivot Anchor: None

Materialization Beh...

Power Pivot Anchor

Specifies thedimension to anchor ... dimension will be the fields of the ... column in your fact table.

Specify **None** (default) to have a single data column in your fact table.

Specify **Variable** to have members of the Variable dimension be the fields and measures in your fact table.

Specify **Organization** to have members of the Organization dimension(s) be the fields and measures in your fact table. If your view has more than one Organization dimension; then the fields in the fact table will be the combination of the organization dimension members.

Specify **Time** to have members of the Time dimension be the fields and measures in your fact table.

Specify **Scenario** to have members of the Scenario dimension be the fields and measures in your fact table.

Specify **Custom** to have members of the Custom dimension be the fields and measures in your fact table. This is applicable where your view(s) are based on a Custom dimension.

Specify **Recommended** when you are not sure about how to choose the dimension to anchor for your **Power Pivot** model. CONTROL® will choose an anchor dimension in the following order:

a. Custom
b. Scenario
c. Variable

Specify **Row Edge** to have members of the dimension(s) on the Row Edge be the fields and measures in your fact table. If your view has more than one dimension on the Row Edge; then the fields in the fact table will be the combination of members of the dimensions on the row edge.

Specify **Column Edge** to have members of the dimension(s) on the Column Edge be the fields and measures in your fact table. If your view has more than one dimension on the Column Edge; then the fields in the fact table will be the combination of members of the dimensions on the column edge.

Specify **Page Edge** to have members of the dimension(s) on the Page Edge be the fields and measures in your fact table. If your view has more than one dimension on the Page Edge; then the fields in the fact table will be the combination of members of the dimensions on the page edge.

Materialization Behavior

Specify whether to create relational tables, relational views, or virtual tables:

- **Automatic** will decide whether the view(s) of the **Power Pivot** model is simple enough so that relational views on the underlying CONTROL® tables can be used, so that the data and meta-data is always current without re-executing the relational export. There are a lot of conditions to satisfy, so this is rarely the case. If the computational view(s) has any calculations, currency conversions, custom dimensions, merged hierarchies, or a non-anchored data table - the view will be materialized.

- **Materialize** will create relational tables when you export the CONTROL® information. This option yields the greatest degree of fidelity with the data and meta-data you see in the CONTROL® view, accommodating dynamic calculations, currency conversions, multiple hierarchies of the same dimension, and custom dimensions.

- **Do not Materialize** will create relational views of the hierarchy tables and data table in your model, not restricted by the filters and branches in the view. The style must be **Dimension and Fact** and the fact table will be anchored on the same dimension as the model’s data table. This option should only be used when you want the export to be as close to the ”raw” data in the underlying CONTROL® model.

- **Materialize as Virtual Tables** is functionally equivalent to **Materialize**, except that no new tables are created. The exported data is written to pre-defined utility tables, and relational views on those tables are created. This option is useful if table creation is restricted or you need to quarantine the exported data in a single location.

- **Create views on base tables** creates relational views on the underlying hierarchy and data tables in your model, restricted by the filters and branches in your view. This option should only be selected when there is no dynamically calculated data (other than time or organization rollups), no currency conversion, and one hierarchy for each dimension in the view.

- **Materialize without Foreign Keys** will create relational tables without the foreign keys on the Fact table when you export the

CONTROL® information. This option helps debug export of multiple views.

The advantage of the options that do not materialize data is you do not need to rerun the export when the CONTROL® data or meta-data changes.

You should only use those options when you have a complete understanding of their limitations.

Materialization Protocol

Specify how to iterate on parts of the model/view for very large data sets. You only need to use this option if the view contains a very large amount of data or if the memory of the computer running the export is limited.

The syntax for specifying the data transfer protocol is the same as used in mappings and transforms. For more details, you can refer to the **Specifying a Data Transfer Protocol** topic in the CONTROL® Administrators Reference Manual.

Table Schema

Specify the relational schema where you want the exported table(s) created. If left blank (default), then the resolved value in the **&KCI_RDBMS_Schema** keyword defines the schema.

Note: Processes external to CONTROL® will access these tables, so you should be sensitive to your company's security policies.

Table Name or Prefix

Specify prefix for the name of the relational table(s) that are created. For a star schema, this will be the prefix of the table name, for example, ***prefix*_Fact**, ***prefix*_Dimension1**, etc.

If left blank, then the model and dimension names define the relational table names.

Augment Dimensions

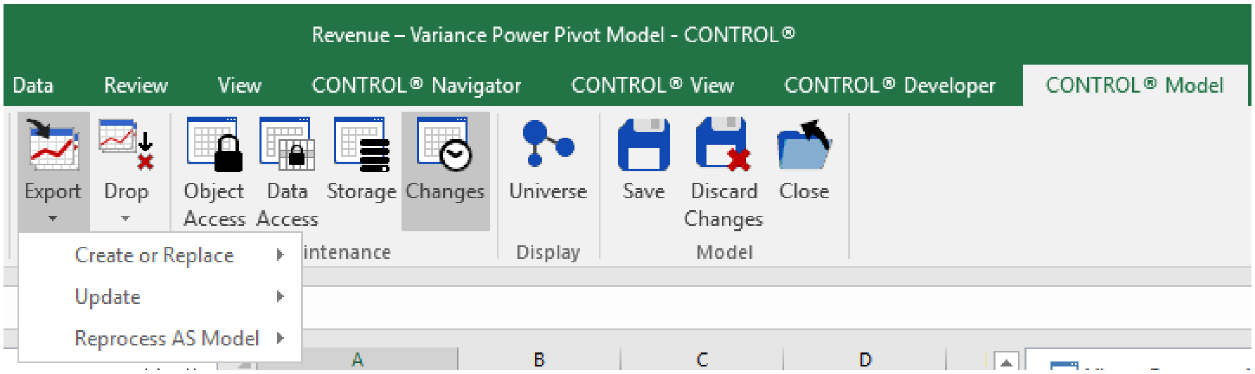
This option determines whether additional columns will be added to dimension tables, and only applies if the **Power Pivot Style** is **Dimension and Fact Tables**.

- **No additional processing** (Default) indicates that only the columns indicated by the views' selected branches and implied by the **Materialization Behavior** will be exported.

	• **Create a date/time field on the time dimension** will create a column named **FullDate**, with a data type of **datetime**, and the values will correspond to the implied date of the lowest level members of the time dimension in the view. This column is required if you want to make use of the Power BI time intelligence features. • **Add all fields to all dimensions** will add all the levels and attributes reported to by any levels in a dimension branch, as well as adding the **FullDate** column to the time dimension. If the model is exported to AS, hierarchies will be defined for any additional branches.
AS Database[AS Model]	Specify the Analysis Services database and model in the format: **databaseID[modelName]** For a CONTROL-Managed AS **Power Pivot** model: • If you leave this field blank, the database will have the same ID and name as the CONTROL® **Power Pivot** model. • If you specify only the **databaseID**, then the AS model will have the same name as the CONTROL® model. For an externally managed AS database and model: • You must specify the **databaseID**. • If you omit the model name, CONTROL® will automatically use the name of the AS model in that database.
AS Server	Specify the name of the server where the Analysis Services tabular instance for this model resides. For a CONTROL-Managed AS model, if you leave this server name blank, then the replacement value of the **&KCI_ASServer** keyword defines the server.
Impersonation Mode	Specify how SQL Server Analysis Services will query the supplied CONTROL® data.
AS Connection Behavior	You can connect to the Analysis Services server with the current connection information saved in the CONTROL® database or use the information saved in the workbook that you can modify.

Export a Power Pivot model

You can export a **Power Pivot** model manually or create a script to export it on a schedule. To export it manually edit the **Power Pivot** model. On the **CONTROL® Model** ribbon, in the **Construction** group, click the arrow on the bottom of the **Export** button.



The table below defines the multiple export commands you can select:

Export command	Used to
Create or Replace > Relational Objects	Create new relational tables/views based on the CONTROL® **Power Pivot** model or if the tables/views already exist, then rebuilds the relational tables removing any customizations done.
Create or Replace > All	Create new relational tables/views based on the CONTROL® **Power Pivot** model and creates the AS model using the exported relational tables/views. If the tables/views and AS model already exist, then rebuilds the relational tables removing any customizations done and then rebuilds the AS model.
Update > Relational Objects	Repopulates or recreates relational tables/views based on the CONTROL® **Power Pivot** model retaining any customizations (additional views or exported tables, measures/KPIs, Alias, relationship overrides) done after the initial export.
Update > All	Repopulates or recreates relational tables/views based on the CONTROL® **Power Pivot** model retaining any customizations (additional views or exported tables, measures/KPIs, Alias, relationship overrides) done after the initial export; then rebuilds and repopulates the AS model.
Reprocess AS model > Recreate	Rebuilds and repopulates the AS model from the (possibly customized) relational objects. This option only applies to models

	exported to AS. The relational tables/views are not recreated or repopulated.
Reprocess AS model > Process Only	Reprocess the AS model only from the current values in the relational tables. This option only applies to models exported to AS. The relational tables/views are not recreated or repopulated. Use this option if you have changed the AS model with an external tool (e.g., Tabular Editor).

Note: Important change from 10.4 – **Create or Replace All** will NO LONGER preserve customizations, like adding measures.

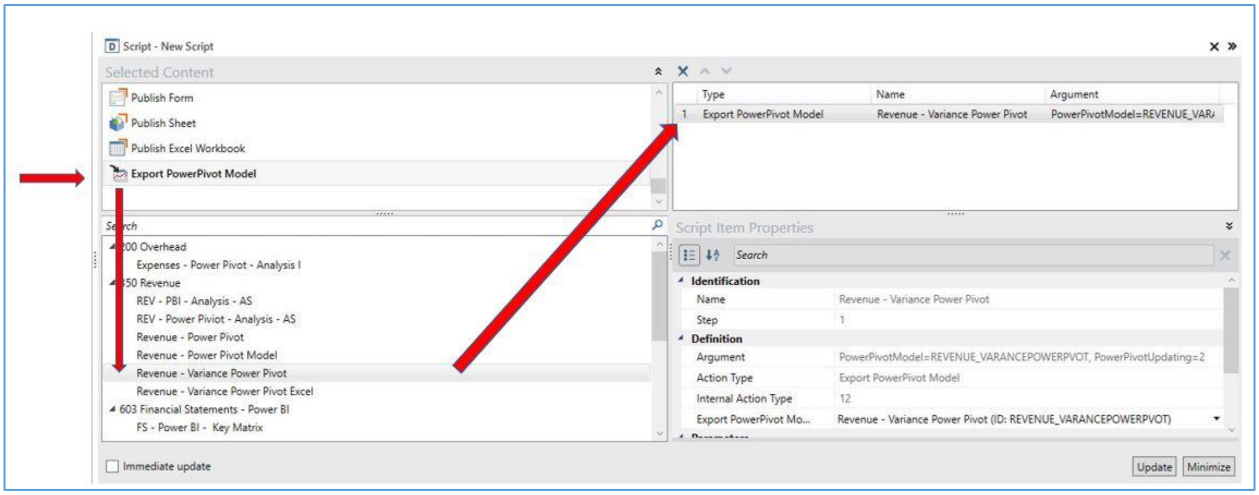
Below are the processes that run if you choose the **Create or Replace** command:

- The base model and view are validated for the user, based on CONTROL®'s object and data access privileges, as are all additional views and additional tables.
- The views' meta-data and data are generated according to the view definitions and options, including currency translation.
- The meta-data and data in the views are manipulated and the dimension and fact tables are written and populated.
- CONTROL® creates SQL datasource objects, one for each created table, so you can review the relational objects using standard CONTROL® capabilities.
    - The export of variable and custom dimensions containing simple ratio calculations will automatically create the appropriate DAX, when they are the anchor dimension for the export.
- If the **Analysis Services Usage** property is set to **Exported to AS – Power BI and Excel (Recommended) or Exported to AS – Power BI**, a connection is made to the Analysis Services Server, and the AS objects are created and processed.

Note: The export to AS tabular requires SQL Server 2016 or a higher version.

Script the export of Power Pivot model

To facilitate the periodic refresh of the relational tables and AS models, **Export PowerPivot Model** is available as a script action step.



Script - New Script

Selected Content

- Publish Form
- Publish Sheet
- Publish Excel Workbook
- Export PowerPivot Model

Search

- 200 Overhead
  - Expenses - Power Pivot - Analysis I
- 350 Revenue
  - REV - PBI - Analysis - AS
  - REV - Power Pivot - Analysis - AS
  - Revenue - Power Pivot
  - Revenue - Power Pivot Model
  - Revenue - Variance Power Pivot
  - Revenue - Variance Power Pivot Excel
- 603 Financial Statements - Power BI
  - FS - Power BI - Key Matrix

	Type	Name	Argument
1	Export PowerPivot Model	Revenue - Variance Power Pivot	PowerPivotModel=REVENUE_VAR/

Script Item Properties

Identification

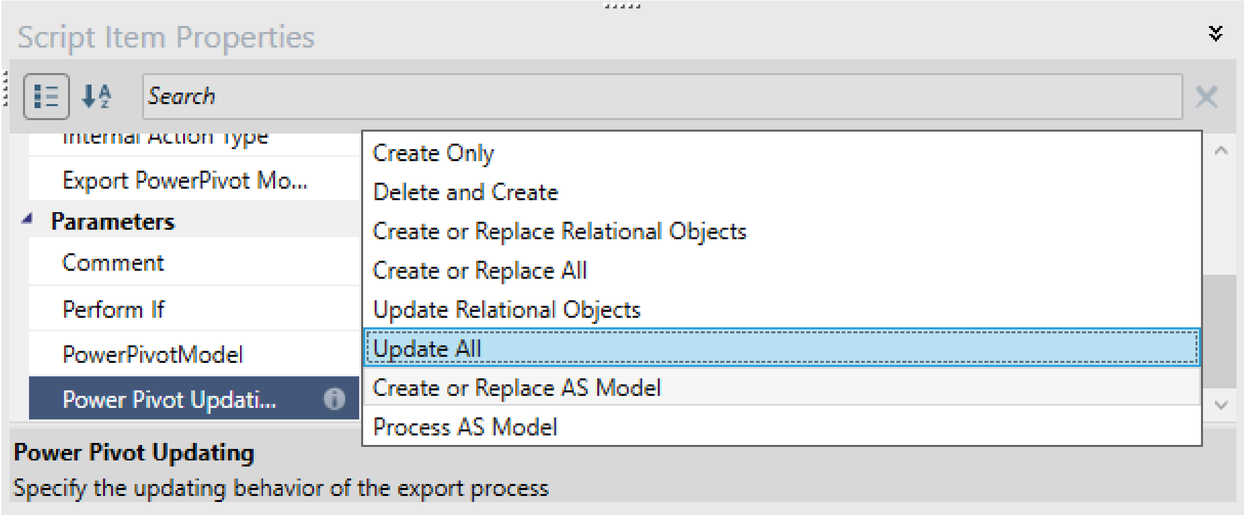
Name	Revenue - Variance Power Pivot
Step	1

Definition

Argument	PowerPivotModel=REVENUE_VARANCEPOWERPIVOT, PowerPivotUpdating=2
Action Type	Export PowerPivot Model
Internal Action Type	12
Export PowerPivot Mo...	Revenue - Variance Power Pivot (ID: REVENUE_VARANCEPOWERPVOT)

□ Immediate update

The export options available through the **Export Power Pivot** action step are more comprehensive and include all options from the previous release.



Script Item Properties

Search

- Internal Action Type
- Export PowerPivot Mo...
- Parameters
  - Comment
  - Perform If
  - PowerPivotModel
  - Power Pivot Updati...

- Create Only
- Delete and Create
- Create or Replace Relational Objects
- Create or Replace All
- Update Relational Objects
- Update All
- Create or Replace AS Model
- Process AS Model

Power Pivot Updating
Specify the updating behavior of the export process

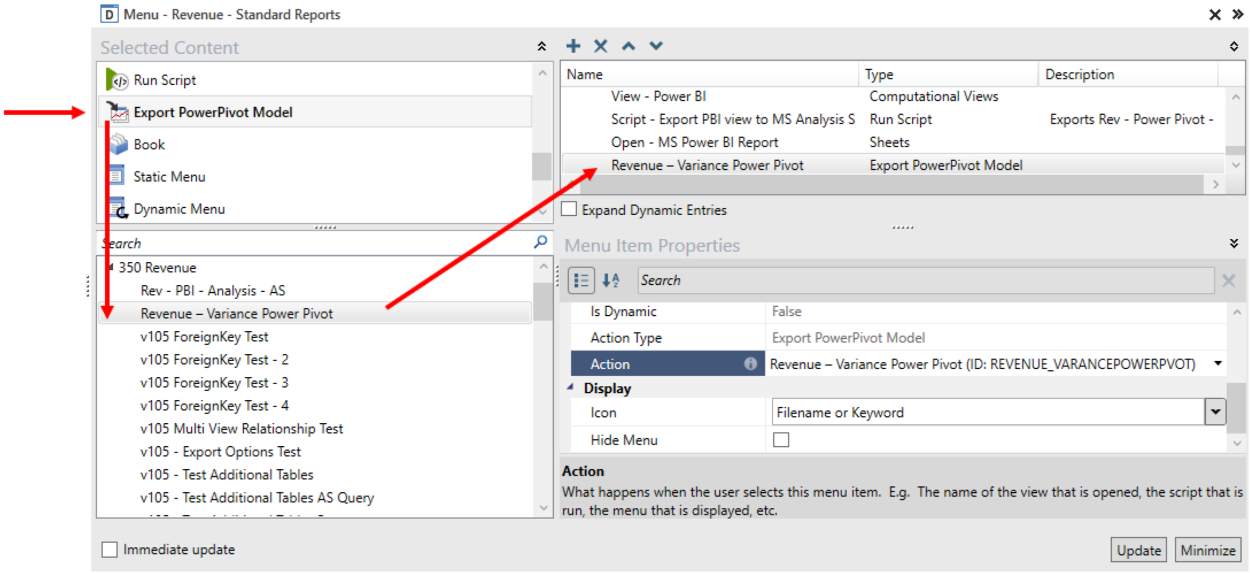
So, in addition to the export options available in the **Power Pivot** model's edit book, the following two additional options are available in the action step:

Export command	Used to
Create Only	Create new relational tables/views based on the CONTROL® **Power Pivot** model. If the relational tables/views or AS database/model exist, the process will fail. Use this option if you do not want to overwrite any existing objects.

Delete and Create	Delete the relational tables/views and re-create them based on the CONTROL® **Power Pivot** model.

Add a menu item to export a Power Pivot Model

To make exporting easily available from the application menu, **Export PowerPivot Model** is now an option for a static or dynamic menu item. The export from the menu item does the **Export > Update All**.

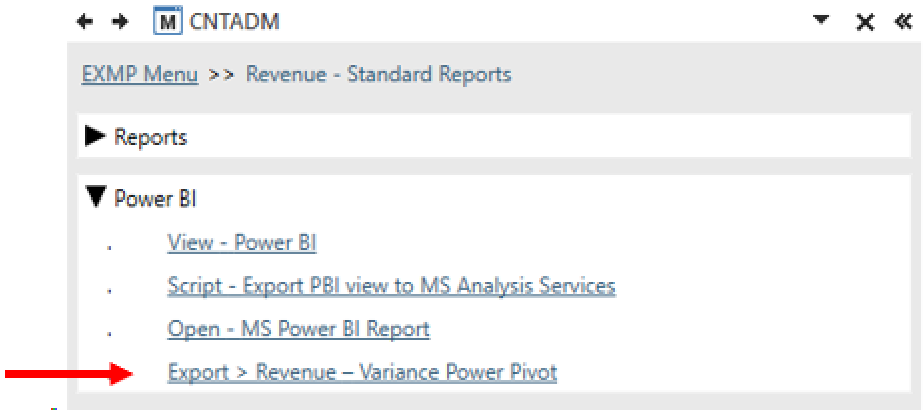


Name	Type	Description
View - Power BI	Computational Views	
Script - Export PBI view to MS Analysis S	Run Script	Exports Rev - Power Pivot -
Open - MS Power BI Report	Sheets	
Revenue – Variance Power Pivot	Export PowerPivot Model	

□ Expand Dynamic Entries

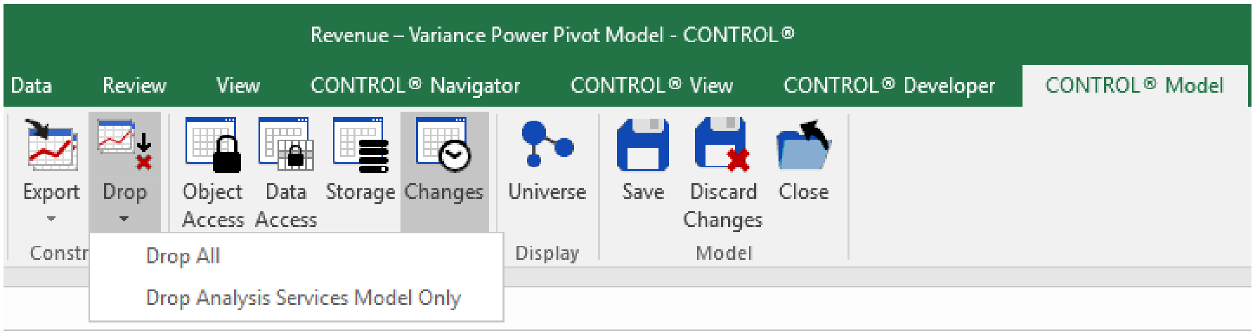
Is Dynamic	False
Action Type	Export PowerPivot Model
Action	Revenue – Variance Power Pivot (ID: REVENUE_VARANCEPOWERPVOT)
Display	
Icon	Filename or Keyword
Hide Menu	□

Action
What happens when the user selects this menu item. E.g. The name of the view that is opened, the script that is run, the menu that is displayed, etc.


CNTADM
EXMP Menu >> Revenue - Standard Reports
Reports
Power BI
View - Power BI
Script - Export PBI view to MS Analysis Services
Open - MS Power BI Report
Export > Revenue – Variance Power Pivot

Drop exported Power Pivot model objects

When you are developing **Power Pivot** models there may be occasions when you need to drop the exported objects and start afresh. You can drop the exported relational objects and/or AS model manually. To drop the objects, edit the **Power Pivot** model. On the **CONTROL® Model** ribbon, in the **Construction** group, click the arrow on the bottom of the **Drop** button.



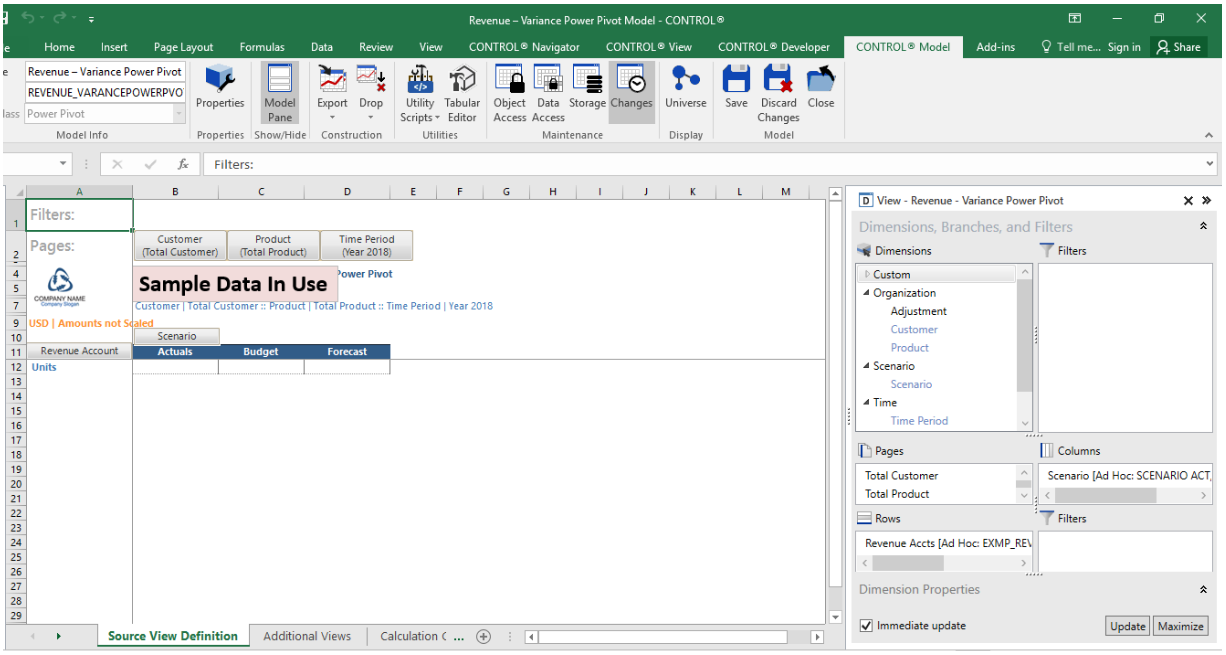
The table below explains the available Drop options:

Drop command	Used to
Drop All	Deletes the relational tables/views and the Analysis Services Model.
Drop Analysis Services Model Only	Deletes just the Analysis Services Model.

The Power Pivot model edit book

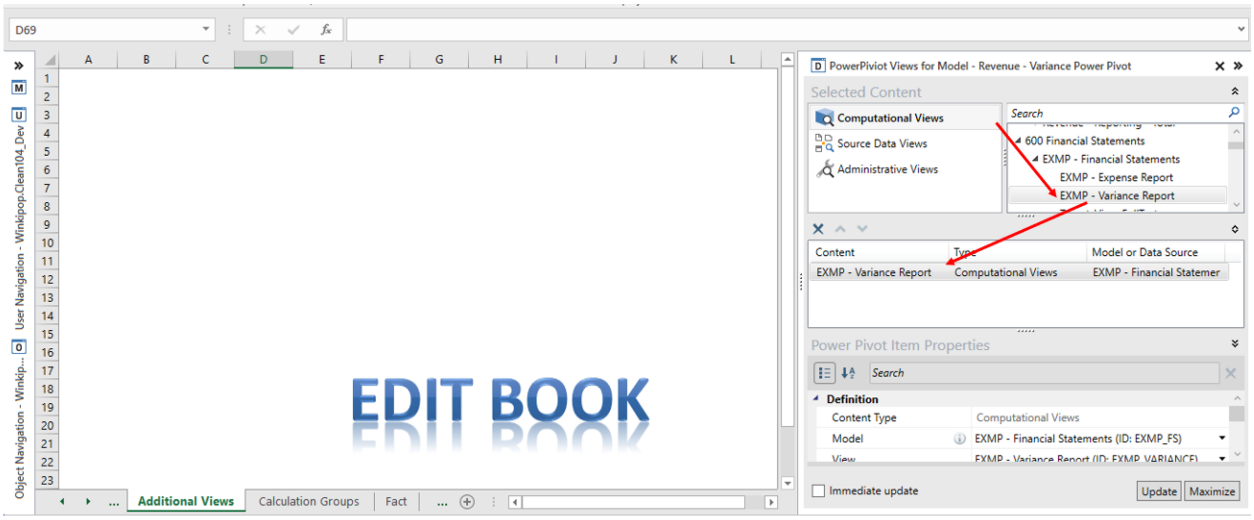
Source view definition

The first tab of the edit book allows you to review or customize the primary view of the **Power Pivot** model. If you leave the **Base View** property of the model blank, a new view is created, which is dedicated to the **Power Pivot** model, so you can modify it without affecting any existing reports or processes in your application.



Additional views

If you want data from multiple models or views included in the export, you can add them in the task pane on the **Additional Views** tab. This is particularly useful when using AS because Power BI visualizations can only contain data from one AS model.



You can override the **style**, **anchor dimension**, **materialization behavior** and **protocol properties** of each additional view.

When you export the **Power Pivot** model

1. The primary view processes first
2. The additional views process in the sequence they appear.
3. For the views following the primary view, if a previously exported dimension is present, CONTROL® evaluates whether the exported relational table/view can be reused. If the previous table contains all the selected members and levels, then no additional dimension table is exported.
4. Each view will have its own fact table. This behavior is useful when you are exporting data from multiple models/views that share one or more common dimensions.
5. When exported tables are materialized, CONTROL® creates foreign key relationships between the fact table(s) and the dimension tables, so that other tools (such as Power BI) can automatically understand how the tables are related.
6. When CONTROL® builds an AS Tabular model, it automatically creates the relationships between the tables, whether you have exported relational tables or relational views.
7. If there are multiple views with shared dimensions the relationships will be created for each fact table.
8. Additional tables specified in the Exported tables tab are included when building the AS Tabular model.
9. If the additional table(s) is an **AS Query** datasource, then the DAX expression that defines the AS Query will be used to create a calculated table in AS.
10. Any relationship overrides defined in the **Exported Tables** tab are then applied on the AS Tabular model.

Note: As of this writing, AS Tabular does not support multi-column foreign key relationships.

Calculation groups

Calculation groups are a concept introduced in SQL Server 2019 Analysis Services that allow you to define one or more sets of measures that you can use with all the other measures in an AS Tabular model.

For example, you may want to have a year-to-date or year-over-year calculation for Sales, Cost, and Profit measures. Instead of defining six new measures to accomplish this, you can add a calculation group with two generic measures – year-to-date and year-over year, and you can use them with all your financial variables.

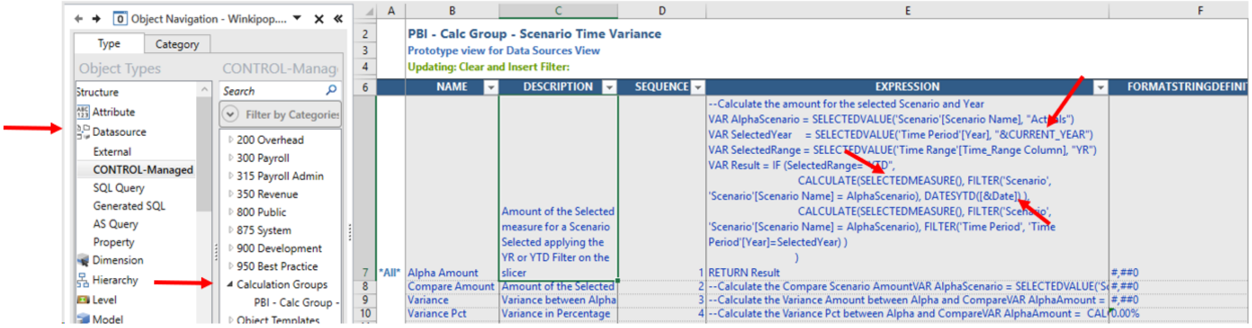
Refer to the following links for background and a detailed discussion of calculation groups:

- https://www.sqlbi.com/articles/introducing-calculation-groups/
- https://www.sqlbi.com/articles/understanding-calculation-groups/
- https://www.sqlbi.com/articles/understanding-the-application-of-calculation-items/
- https://www.sqlbi.com/articles/understanding-calculation-group-precedence/
- https://www.sqlbi.com/articles/controlling-format-strings-in-calculation-groups/

- https://www.sqlbi.com/articles/using-calculation-groups-to-selectively-replace-measures-in-dax-expressions/
- https://www.sqlbi.com/articles/using-calculation-groups-to-switch-between-dates/
- https://docs.microsoft.com/en-us/analysis-services/tabular-models/calculation-groups?view=sql-server-ver15

A calculation group appears in Power BI as another table. In CONTROL®, you define calculation groups as a **CONTROL-Managed** datasource object in the **Calculation Groups** category. Each record in the datasource table represents a calculation item, or measure.

Here is an example of a **Calculation Group** datasource that computes the Scenario and Time variance calculations:



PBI - Calc Group - Scenario Time Variance

Prototype view for Data Sources View

Updating: Clear and Insert Filter:

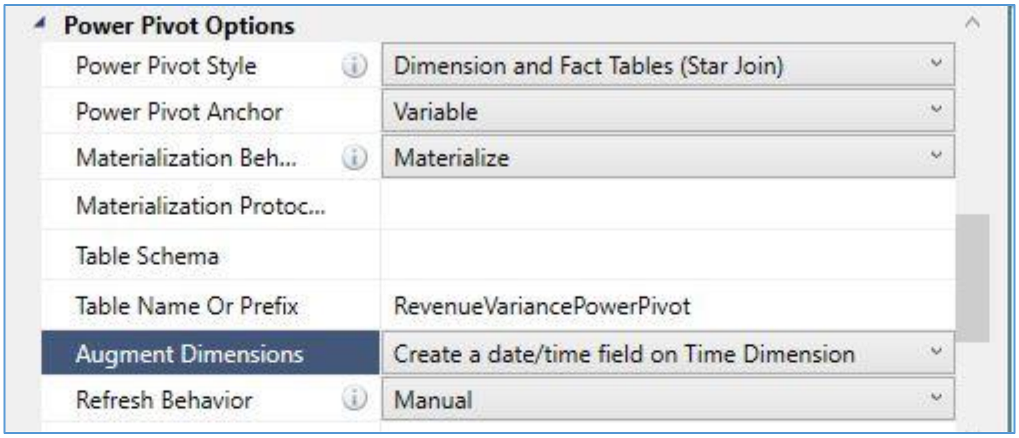
A	NAME	DESCRIPTION	SEQUENCE	EXPRESSION	FORMATSTRINGDEFINI
All	Alpha Amount	Amount of the Selected measure for a Scenario Selected applying the YR or YTD Filter on the slicer	1	--Calculate the amount for the selected Scenario and Year VAR AlphaScenario = SELECTEDVALUE('Scenario'[Scenario Name], "Actuals") VAR SelectedYear = SELECTEDVALUE('Time Period'[Year], "&CURRENT_YEAR") VAR SelectedRange = SELECTEDVALUE('Time Range'[Time_Range Column], "YR") VAR Result = IF (SelectedRange="YTD", CALCULATE(SELECTEDMEASURE(), FILTER('Scenario', 'Scenario'[Scenario Name] = AlphaScenario), DATESYTD([&Date])), CALCULATE(SELECTEDMEASURE(), FILTER('Scenario', 'Scenario'[Scenario Name] = AlphaScenario), FILTER('Time Period', 'Time Period'[Year]=SelectedYear))) RETURN Result	#,##0
	Compare Amount	Amount of the Selected	2	--Calculate the Compare Scenario AmountVAR AlphaScenario = SELECTEDVALUE('S	#,##0
	Variance	Variance between Alpha	3	--Calculate the Variance Amount between Alpha and CompareVAR AlphaAmount =	#,##0
	Variance Pct	Variance in Percentage	4	--Calculate the Variance Pct between Alpha and CompareVAR AlphaAmount = CAL	0.00%

The following are special characteristics of the **Expression**:

- The **SELECTEDMEASURE()** argument is a placeholder for any measure to which the calculation item is applied.
- The **[&Date]** argument used in time intelligence calculations will be replaced by the augmented **datetime** column of the time dimension table, for example, **'Time Period'[FullDate]**.
- Any remaining keywords will be evaluated in the model scope of the **Power Pivot** model and substituted with by their replacement values. You can see the use of the CURRENT_YEAR keyword in our formula in the screenshot above.

You should put all Calculation Group datasources in a category with its ID **CalculationGroups**. If this category exists, the **Calculation Groups** worksheet will appear in the **Power Pivot** model's edit book (after the **Additional Views** worksheet), which allows you to add calculation groups to the model.

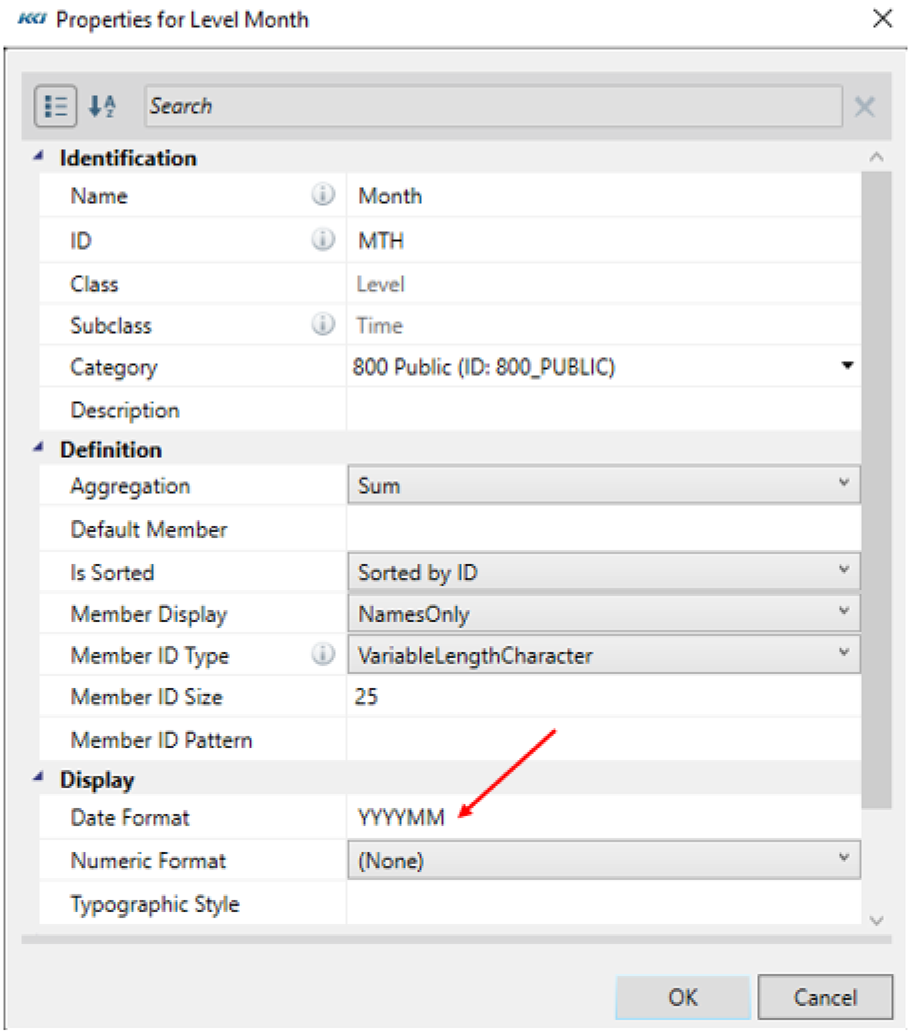
To use calculation groups with **Time Intelligence** functions in a **Power Pivot** model you need to set the **Augment Dimensions** property to **Create a date/time field on Time Dimension** or **Add all fields to all dimensions**.



Power Pivot Options

Power Pivot Style	Dimension and Fact Tables (Star Join)
Power Pivot Anchor	Variable
Materialization Beh...	Materialize
Materialization Protoc...	
Table Schema	
Table Name Or Prefix	RevenueVariancePowerPivot
Augment Dimensions	Create a date/time field on Time Dimension
Refresh Behavior	Manual

The **FullDate** column is constructed based on the **Date Format** property of the root level of the time dimension in your **Power Pivot model** and hence need to be set to use this augment dimension feature.



Properties for Level Month

Identification

Name	Month
ID	MTH
Class	Level
Subclass	Time
Category	800 Public (ID: 800_PUBLIC)
Description	

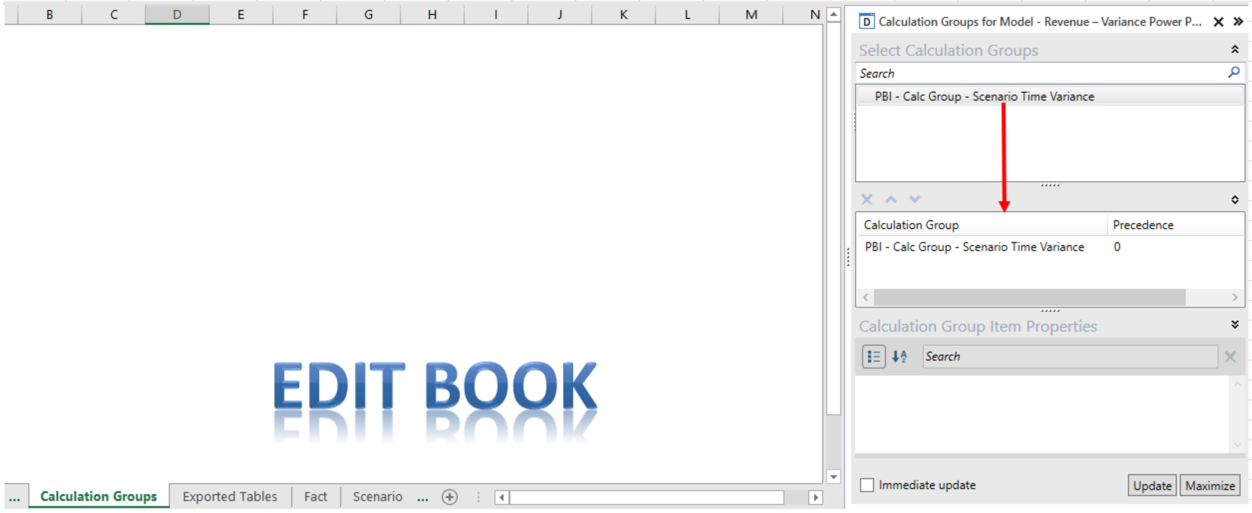
Definition

Aggregation	Sum
Default Member	
Is Sorted	Sorted by ID
Member Display	NamesOnly
Member ID Type	VariableLengthCharacter
Member ID Size	25
Member ID Pattern	

Display

Date Format	YYYYMM
Numeric Format	(None)
Typographic Style	

Now, you can add the Calculation Group to the **Power Pivot** model in the **Calculation Groups** worksheet.



Calculation Groups for Model - Revenue – Variance Power P...

Select Calculation Groups

Search

PBI - Calc Group - Scenario Time Variance

Calculation Group	Precedence
PBI - Calc Group - Scenario Time Variance	0

Calculation Group Item Properties

Search

□ Immediate update

Update | Maximize

You can have any number of Calculation Groups in your model, but if you have more than one you should take care to set the **Precedence** property correctly to determine the order of calculation in AS. Calculation Groups with a higher **precedence** value calculate before those with a lower value (descending order).

The following article provides more details about what to avoid while setting Calculation Group **precedence**.

https://www.sqlbi.com/articles/avoiding-pitfalls-in-calculation-groups-precedence/

Our Sample Calculation Group has four calculation items:

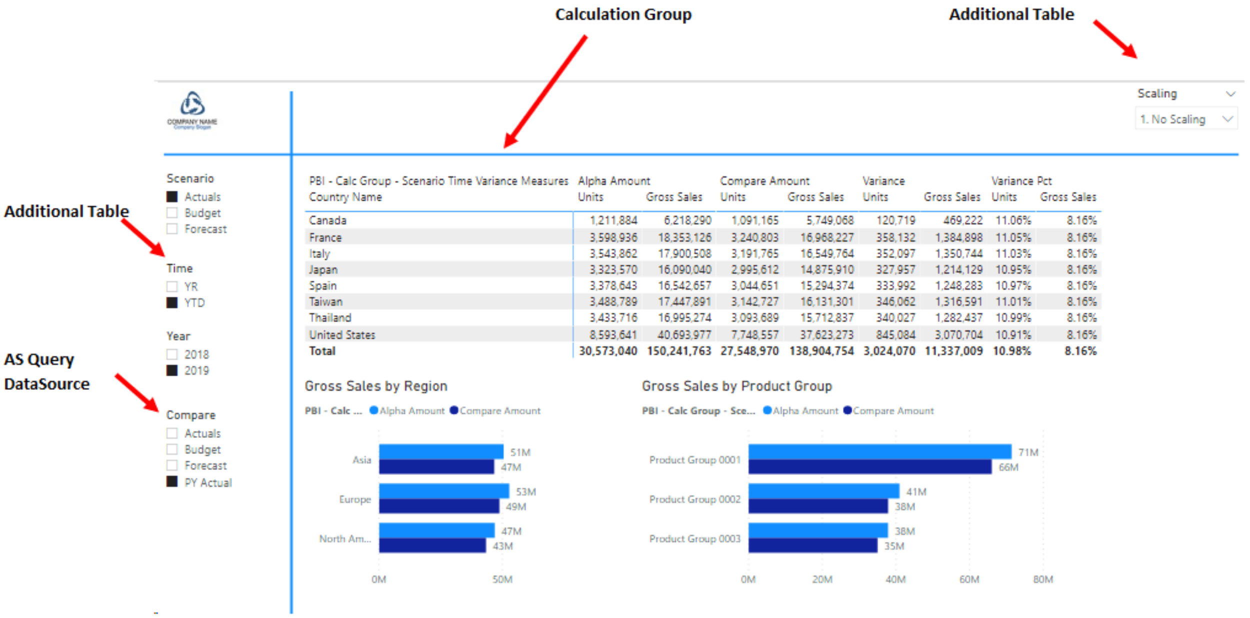
Calculation Item	DAX Expression
Alpha Amount	--Calculate the amount for the selected Scenario and Year VAR AlphaScenario = SELECTEDVALUE('Scenario'[Scenario Name], "Actuals") VAR SelectedYear = SELECTEDVALUE('Time Period'[Year], "&CURRENT_YEAR") VAR SelectedRange = SELECTEDVALUE('Time Range'[Time_Range Column], "YR") VAR Result = IF (SelectedRange="YTD", CALCULATE(SELECTEDMEASURE(), FILTER('Scenario', 'Scenario'[Scenario Name] = AlphaScenario), DATESYTD('Time Period'[FullDate])), CALCULATE(SELECTEDMEASURE(), FILTER('Scenario', 'Scenario'[Scenario Name] = AlphaScenario), FILTER('Time Period', 'Time Period'[Year]=SelectedYear))) RETURN Result
Compare Amount	--Calculate the Compare Scenario Amount VAR AlphaScenario = SELECTEDVALUE('Scenario'[Scenario Name], "Actuals") VAR ScenarioCompare = SELECTEDVALUE('Scenario Compare'[Scenario Name], "Forecast")

	VAR SelectedYear = SELECTEDVALUE('Time Period'[Year], "&CURRENT_YEAR") VAR SelectedRange = SELECTEDVALUE('Time Range'[Time_Range Column], "YR") VAR CompareAmount = IF(ScenarioCompare = "PY Actual", IF(SelectedRange = "YTD", CALCULATE(SELECTEDMEASURE(), FILTER('Scenario', 'Scenario'[Scenario Name] = AlphaScenario), DATESYTD(SAMEPERIODLASTYEAR('Time Period'[FullDate]))), CALCULATE(SELECTEDMEASURE(), FILTER('Scenario', 'Scenario'[Scenario Name] = AlphaScenario),SAMEPERIODLASTYEAR('Time Period'[FullDate]))) , IF(SelectedRange = "YTD", CALCULATE(SELECTEDMEASURE(), FILTER(All('Scenario'), 'Scenario'[Scenario Name] = ScenarioCompare), DATESYTD('Time Period'[FullDate])), CALCULATE(SELECTEDMEASURE(), FILTER(All('Scenario'), 'Scenario'[Scenario Name] = ScenarioCompare)))) RETURN CompareAmount
Variance	--Calculate the Variance Amount between Alpha and Compare VAR AlphaAmount = CALCULATE(SELECTEDMEASURE(), FILTER('Scenario Time Variance', 'Scenario Time Variance'[PBI - Calc Group - Scenario Time Variance Measures]="Alpha Amount")) VAR CompareAmount = CALCULATE(SELECTEDMEASURE(), FILTER('Scenario Time Variance', 'Scenario Time Variance'[PBI - Calc Group - Scenario Time Variance Measures]="Compare Amount")) VAR Result = AlphaAmount - CompareAmount RETURN Result
Variance Pct	--Calculate the Variance Pct between Alpha and Compare VAR AlphaAmount = CALCULATE(SELECTEDMEASURE(), FILTER('Scenario Time Variance', 'Scenario Time Variance'[PBI - Calc Group - Scenario Time Variance Measures]="Alpha Amount")) VAR CompareAmount = CALCULATE(SELECTEDMEASURE(), FILTER('Scenario Time Variance', 'Scenario Time Variance'[PBI - Calc Group - Scenario Time Variance Measures]="Compare Amount")) VAR VarianceAmount = AlphaAmount - CompareAmount VAR VarPct = DIVIDE(VarianceAmount, CompareAmount) RETURN VarPct

The Calculation Group uses a mix of relational tables from the **Power Pivot** model (Scenario and Time Period), additional tables (Scenario Compare, Time Range) and Calculation items within the Calculation Group.

The intent is to produce reports where you select a scenario and then select another scenario with which to compare your data. The option of comparison of YTD or Total year is also part of the calculations.

Here is how the Calculation Group appears in Power BI after exporting the **Power Pivot** model:



Scaling
1. No Scaling

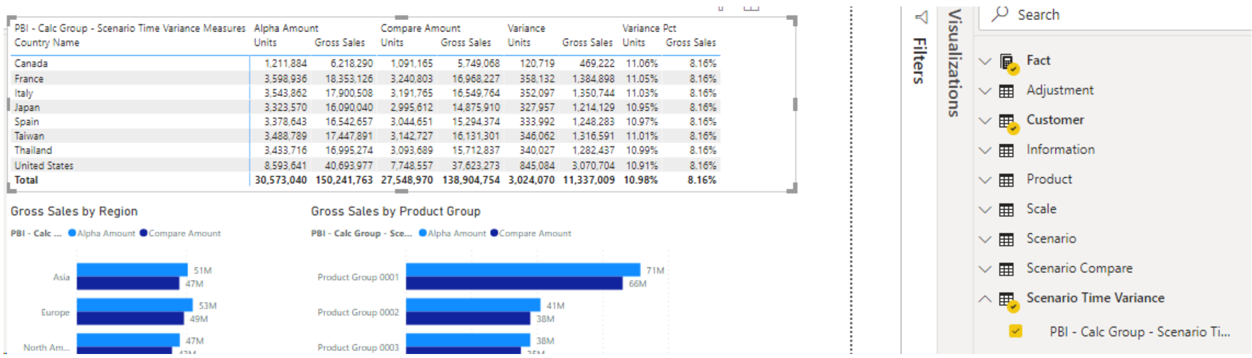
Scenario
■ Actuals
□ Budget
□ Forecast

Time
□ YR
■ YTD

Year
□ 2018
■ 2019

Compare
□ Actuals
□ Budget
□ Forecast
■ PY Actual

PBI - Calc Group - Scenario Time Variance Measures	Alpha Amount		Compare Amount		Variance		Variance Pct	
Country Name	Units	Gross Sales	Units	Gross Sales	Units	Gross Sales	Units	Gross Sales
Canada	1,211,884	6,218,290	1,091,165	5,749,068	120,719	469,222	11.06%	8.16%
France	3,598,936	18,353,126	3,240,803	16,968,227	358,132	1,384,898	11.05%	8.16%
Italy	3,543,862	17,900,508	3,191,765	16,549,764	352,097	1,350,744	11.03%	8.16%
Japan	3,323,570	16,090,040	2,995,612	14,875,910	327,957	1,214,129	10.95%	8.16%
Spain	3,378,643	16,542,657	3,044,651	15,294,374	333,992	1,248,283	10.97%	8.16%
Taiwan	3,488,789	17,447,891	3,142,727	16,131,301	346,062	1,316,591	11.01%	8.16%
Thailand	3,433,716	16,995,274	3,093,689	15,712,837	340,027	1,282,437	10.99%	8.16%
United States	8,593,641	40,693,977	7,748,557	37,623,273	845,084	3,070,704	10.91%	8.16%
Total	**30,573,040**	**150,241,763**	**27,548,970**	**138,904,754**	**3,024,070**	**11,337,009**	**10.98%**	**8.16%**



PBI - Calc Group - Scenario Time Variance Measures	Alpha Amount		Compare Amount		Variance		Variance Pct	
Country Name	Units	Gross Sales	Units	Gross Sales	Units	Gross Sales	Units	Gross Sales
Canada	1,211,884	6,218,290	1,091,165	5,749,068	120,719	469,222	11.06%	8.16%
France	3,598,936	18,353,126	3,240,803	16,968,227	358,132	1,384,898	11.05%	8.16%
Italy	3,543,862	17,900,508	3,191,765	16,549,764	352,097	1,350,744	11.03%	8.16%
Japan	3,323,570	16,090,040	2,995,612	14,875,910	327,957	1,214,129	10.95%	8.16%
Spain	3,378,643	16,542,657	3,044,651	15,294,374	333,992	1,248,283	10.97%	8.16%
Taiwan	3,488,789	17,447,891	3,142,727	16,131,301	346,062	1,316,591	11.01%	8.16%
Thailand	3,433,716	16,995,274	3,093,689	15,712,837	340,027	1,282,437	10.99%	8.16%
United States	8,593,641	40,693,977	7,748,557	37,623,273	845,084	3,070,704	10.91%	8.16%
Total	**30,573,040**	**150,241,763**	**27,548,970**	**138,904,754**	**3,024,070**	**11,337,009**	**10.98%**	**8.16%**

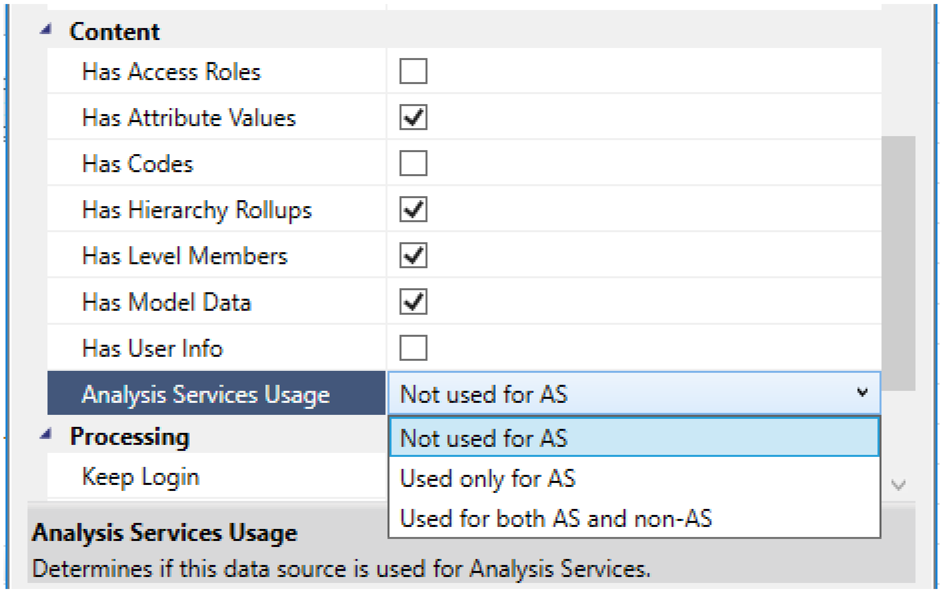
Note: Calculation Groups are only relevant for **Power Pivot** models that you export to Analysis Services, but are useable in both Power BI and Excel pivot tables.

Exported Tables

Exported Tables tab in the **Power Pivot** model edit book allows the addition and customization of the set of data sources exported to the AS model.

Data sources meeting the following criteria are included in the available objects tree, and may be added to the AS model:

- Data sources which have a relational table or view
- AS query data sources
- Data sources which have their **Analysis Services Usage** property defined as either **Used only for AS** or **Used for both AS and non-AS**

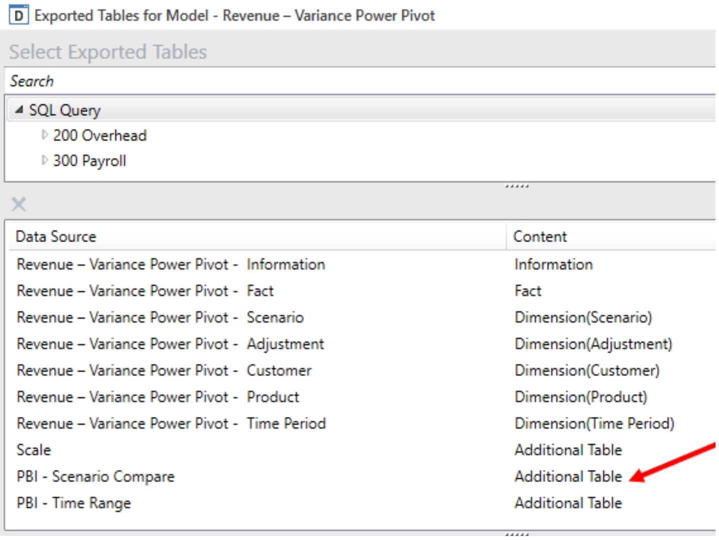


Content	
Has Access Roles	☐
Has Attribute Values	☑
Has Codes	☐
Has Hierarchy Rollups	☑
Has Level Members	☑
Has Model Data	☑
Has User Info	☐
Analysis Services Usage	Not used for AS
Processing	
Keep Login	

Analysis Services Usage

Determines if this data source is used for Analysis Services.

- The **Exported Tables** task pane contains the relational tables of the base and additional views of the **Power Pivot** model in addition to the data sources explicitly added by you.



Exported Tables for Model - Revenue – Variance Power Pivot

Select Exported Tables

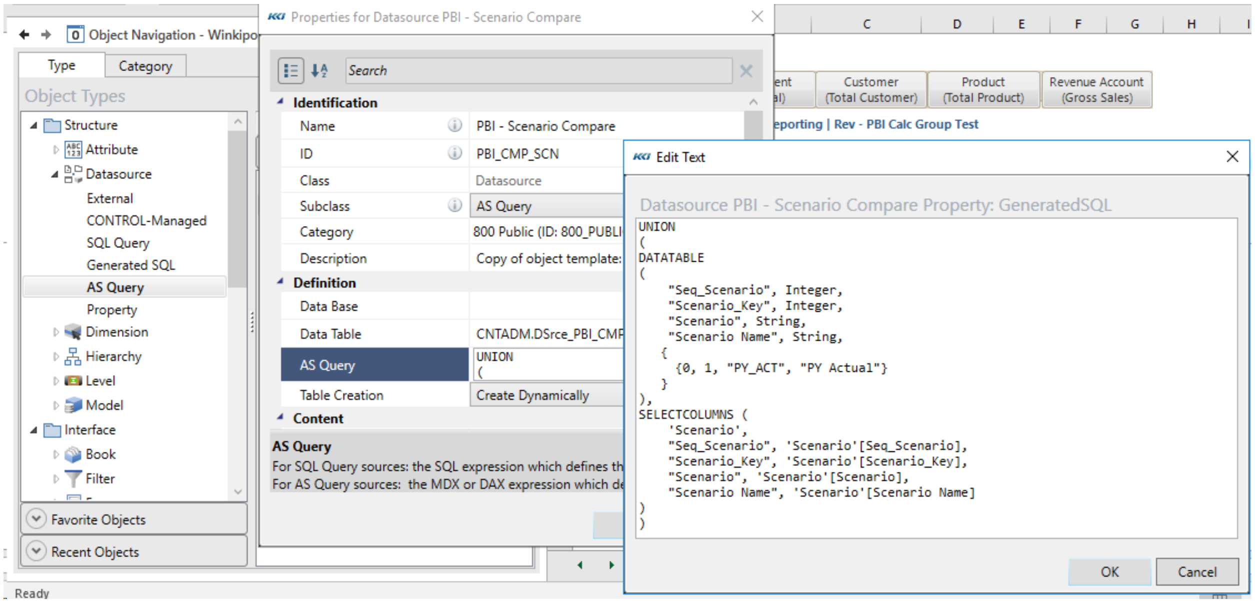
Search

- SQL Query
  - 200 Overhead
  - 300 Payroll

Data Source	Content
Revenue – Variance Power Pivot - Information	Information
Revenue – Variance Power Pivot - Fact	Fact
Revenue – Variance Power Pivot - Scenario	Dimension(Scenario)
Revenue – Variance Power Pivot - Adjustment	Dimension(Adjustment)
Revenue – Variance Power Pivot - Customer	Dimension(Customer)
Revenue – Variance Power Pivot - Product	Dimension(Product)
Revenue – Variance Power Pivot - Time Period	Dimension(Time Period)
Scale	Additional Table
PBI - Scenario Compare	Additional Table
PBI - Time Range	Additional Table

- Data sources which are defined by the **Power Pivot** models' view(s) cannot be removed.
- If an **AS Query** data source is added to the model, the DAX expression that defines the AS Query will be used to create a calculated table in AS.

  Following is the example of an **AS Query** datasource in our **Power Pivot** model.



Properties for Datasource PBI - Scenario Compare

Identification

Name	PBI - Scenario Compare
ID	PBI_CMP_SCN
Class	Datasource
Subclass	AS Query
Category	800 Public (ID: 800_PUBLI
Description	Copy of object template:

Definition

Data Base	
Data Table	CNTADM.DSrce_PBI_CMP
AS Query	UNION (
Table Creation	Create Dynamically

Content

AS Query

For SQL Query sources: the SQL expression which defines th

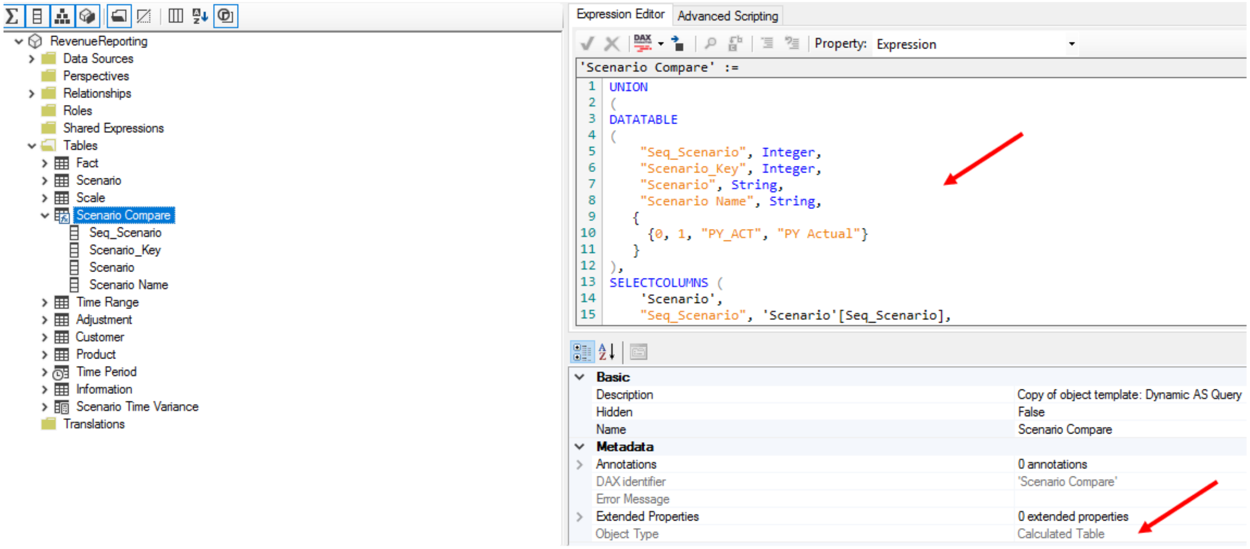
For AS Query sources: the MDX or DAX expression which de

Edit Text

Datasource PBI - Scenario Compare Property: GeneratedSQL

```
UNION
(
DATATABLE
(
    "Seq_Scenario", Integer,
    "Scenario_Key", Integer,
    "Scenario", String,
    "Scenario Name", String,
   {
     {0, 1, "PY_ACT", "PY Actual"}
   }
),
SELECTCOLUMNS (
    'Scenario',
    "Seq_Scenario", 'Scenario'[Seq_Scenario],
    "Scenario_Key", 'Scenario'[Scenario_Key],
    "Scenario", 'Scenario'[Scenario],
    "Scenario Name", 'Scenario'[Scenario Name]
)
)
```

Calculated table created in the AS model.



Property: Expression

```
'Scenario Compare' :=
UNION
(
DATATABLE
(
    "Seq_Scenario", Integer,
    "Scenario_Key", Integer,
    "Scenario", String,
    "Scenario Name", String,
   {
     {0, 1, "PY_ACT", "PY Actual"}
   }
),
SELECTCOLUMNS (
    'Scenario',
    "Seq_Scenario", 'Scenario'[Seq_Scenario],
```

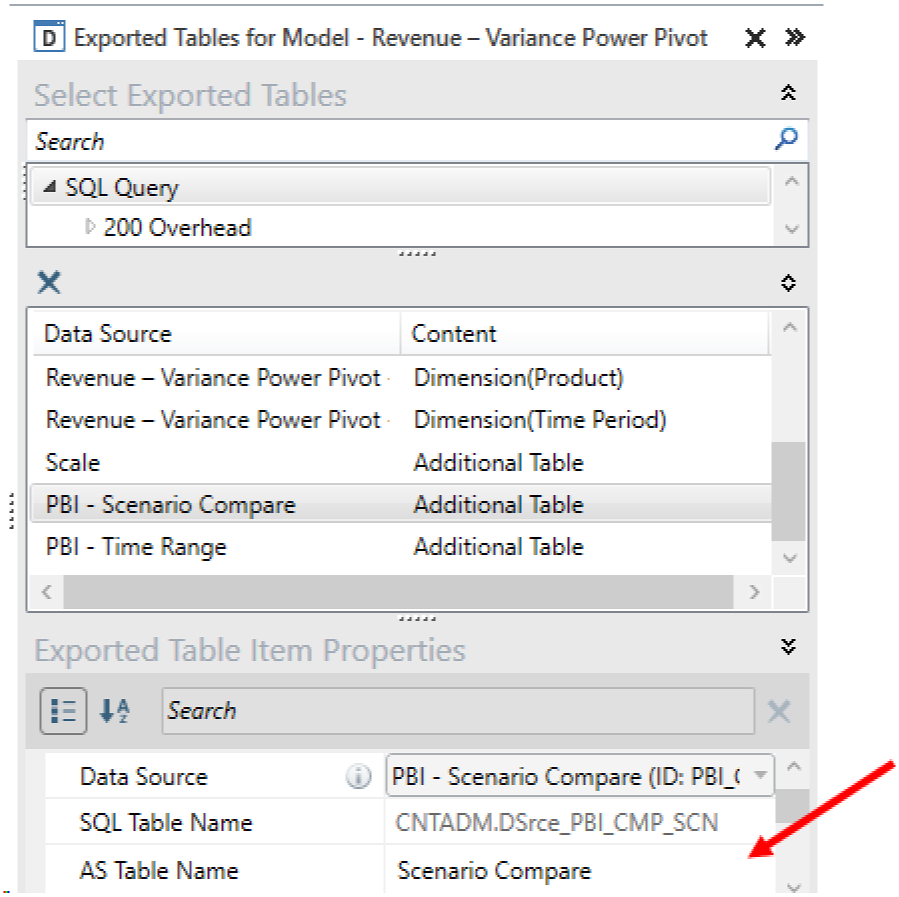
Basic

Description	Copy of object template: Dynamic AS Query
Hidden	False
Name	Scenario Compare

Metadata

Annotations	0 annotations
DAX identifier	'Scenario Compare'
Error Message	
Extended Properties	0 extended properties
Object Type	Calculated Table

- The AS Table name(s) of the data sources can be overridden more easily in this task pane:



Exported Tables for Model - Revenue – Variance Power Pivot

Select Exported Tables

Search

- SQL Query
  - 200 Overhead

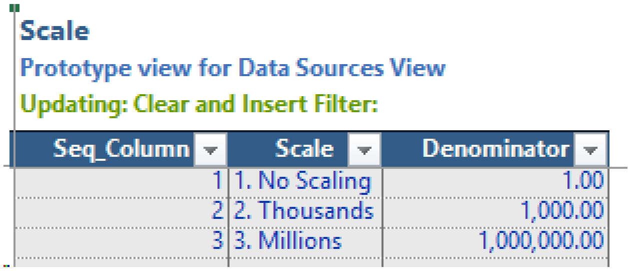
Data Source	Content
Revenue – Variance Power Pivot	Dimension(Product)
Revenue – Variance Power Pivot	Dimension(Time Period)
Scale	Additional Table
PBI - Scenario Compare	Additional Table
PBI - Time Range	Additional Table

Exported Table Item Properties

Search

Data Source	PBI - Scenario Compare (ID: PBI_(
SQL Table Name	CNTADM.DSrce_PBI_CMP_SCN
AS Table Name	Scenario Compare

In our **Revenue - Variance Power Pivot** model we added an additional datasource **Scale** with an aim to provide users the ability to see values with their choice of scaling.



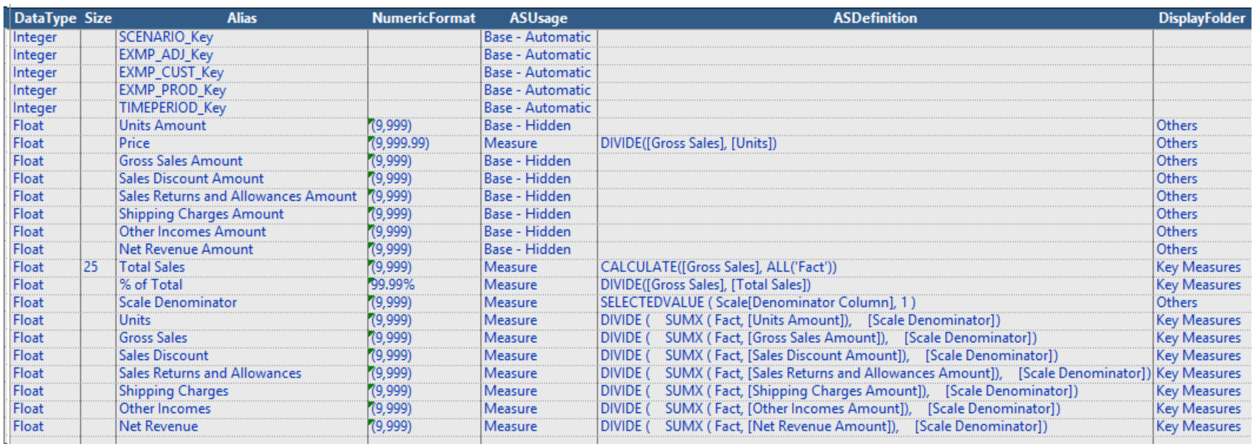
Scale

Prototype view for Data Sources View

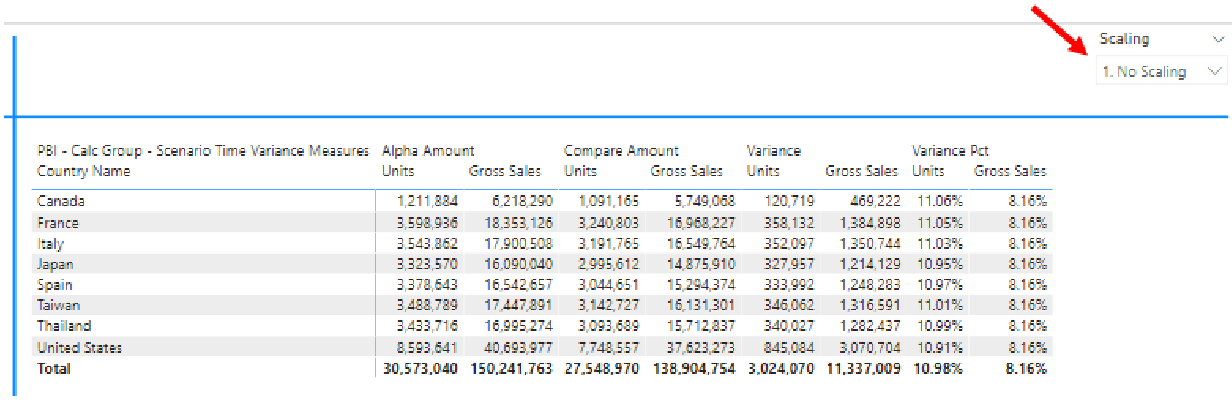
Updating: Clear and Insert Filter:

Seq_Column	Scale	Denominator
1	1. No Scaling	1.00
2	2. Thousands	1,000.00
3	3. Millions	1,000,000.00

We created DAX Measures in the **Fact** table to calculate the data to be shown dynamically based on the choice of scaling made by the user in Power BI.



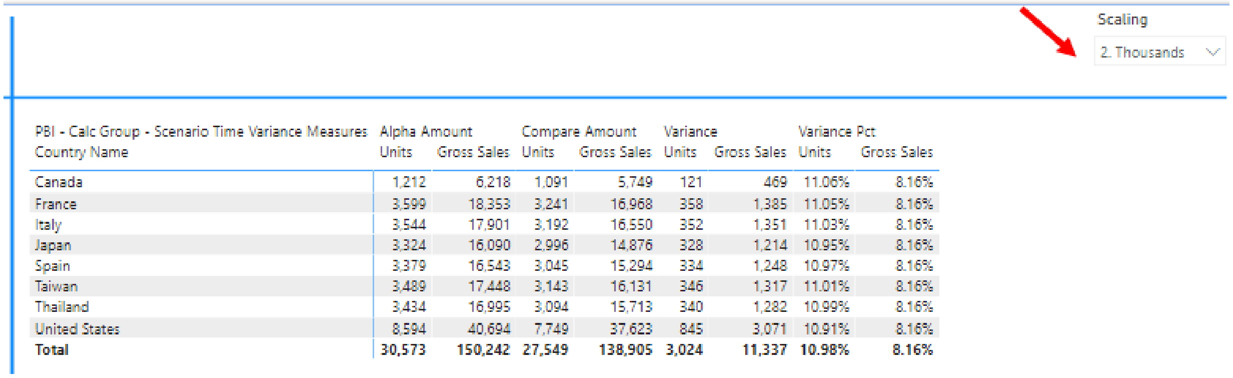
DataType	Size	Alias	NumericFormat	ASUsage	ASDefinition	DisplayFolder
Integer		SCENARIO_Key		Base - Automatic		
Integer		EXMP_ADJ_Key		Base - Automatic		
Integer		EXMP_CUST_Key		Base - Automatic		
Integer		EXMP_PROD_Key		Base - Automatic		
Integer		TIMEPERIOD_Key		Base - Automatic		
Float		Units Amount	(9,999)	Base - Hidden		Others
Float		Price	(9,999.99)	Measure	DIVIDE([Gross Sales], [Units])	Others
Float		Gross Sales Amount	(9,999)	Base - Hidden		Others
Float		Sales Discount Amount	(9,999)	Base - Hidden		Others
Float		Sales Returns and Allowances Amount	(9,999)	Base - Hidden		Others
Float		Shipping Charges Amount	(9,999)	Base - Hidden		Others
Float		Other Incomes Amount	(9,999)	Base - Hidden		Others
Float		Net Revenue Amount	(9,999)	Base - Hidden		Others
Float	25	Total Sales	(9,999)	Measure	CALCULATE([Gross Sales], ALL('Fact'))	Key Measures
Float		% of Total	99.99%	Measure	DIVIDE([Gross Sales], [Total Sales])	Key Measures
Float		Scale Denominator	(9,999)	Measure	SELECTEDVALUE (Scale[Denominator Column], 1)	Others
Float		Units	(9,999)	Measure	DIVIDE (SUMX (Fact, [Units Amount]), [Scale Denominator])	Key Measures
Float		Gross Sales	(9,999)	Measure	DIVIDE (SUMX (Fact, [Gross Sales Amount]), [Scale Denominator])	Key Measures
Float		Sales Discount	(9,999)	Measure	DIVIDE (SUMX (Fact, [Sales Discount Amount]), [Scale Denominator])	Key Measures
Float		Sales Returns and Allowances	(9,999)	Measure	DIVIDE (SUMX (Fact, [Sales Returns and Allowances Amount]), [Scale Denominator])	Key Measures
Float		Shipping Charges	(9,999)	Measure	DIVIDE (SUMX (Fact, [Shipping Charges Amount]), [Scale Denominator])	Key Measures
Float		Other Incomes	(9,999)	Measure	DIVIDE (SUMX (Fact, [Other Incomes Amount]), [Scale Denominator])	Key Measures
Float		Net Revenue	(9,999)	Measure	DIVIDE (SUMX (Fact, [Net Revenue Amount]), [Scale Denominator])	Key Measures



Scaling

1. No Scaling

PBI - Calc Group - Scenario Time Variance Measures	Alpha Amount		Compare Amount		Variance		Variance Pct	
Country Name	Units	Gross Sales	Units	Gross Sales	Units	Gross Sales	Units	Gross Sales
Canada	1,211,884	6,218,290	1,091,165	5,749,068	120,719	469,222	11.06%	8.16%
France	3,598,936	18,353,126	3,240,803	16,968,227	358,132	1,384,898	11.05%	8.16%
Italy	3,543,862	17,900,508	3,191,765	16,549,764	352,097	1,350,744	11.03%	8.16%
Japan	3,323,570	16,090,040	2,995,612	14,875,910	327,957	1,214,129	10.95%	8.16%
Spain	3,378,643	16,542,657	3,044,651	15,294,374	333,992	1,248,283	10.97%	8.16%
Taiwan	3,488,789	17,447,891	3,142,727	16,131,301	346,062	1,316,591	11.01%	8.16%
Thailand	3,433,716	16,995,274	3,093,689	15,712,837	340,027	1,282,437	10.99%	8.16%
United States	8,593,641	40,693,977	7,748,557	37,623,273	845,084	3,070,704	10.91%	8.16%
Total	**30,573,040**	**150,241,763**	**27,548,970**	**138,904,754**	**3,024,070**	**11,337,009**	**10.98%**	**8.16%**



Scaling

2. Thousands

PBI - Calc Group - Scenario Time Variance Measures	Alpha Amount		Compare Amount		Variance		Variance Pct	
Country Name	Units	Gross Sales	Units	Gross Sales	Units	Gross Sales	Units	Gross Sales
Canada	1,212	6,218	1,091	5,749	121	469	11.06%	8.16%
France	3,599	18,353	3,241	16,968	358	1,385	11.05%	8.16%
Italy	3,544	17,901	3,192	16,550	352	1,351	11.03%	8.16%
Japan	3,324	16,090	2,996	14,876	328	1,214	10.95%	8.16%
Spain	3,379	16,543	3,045	15,294	334	1,248	10.97%	8.16%
Taiwan	3,489	17,448	3,143	16,131	346	1,317	11.01%	8.16%
Thailand	3,434	16,995	3,094	15,713	340	1,282	10.99%	8.16%
United States	8,594	40,694	7,749	37,623	845	3,071	10.91%	8.16%
Total	**30,573**	**150,242**	**27,549**	**138,905**	**3,024**	**11,337**	**10.98%**	**8.16%**

There are a lot of creative ways you can enhance your reports in Power BI. You can refer to the following to get started.

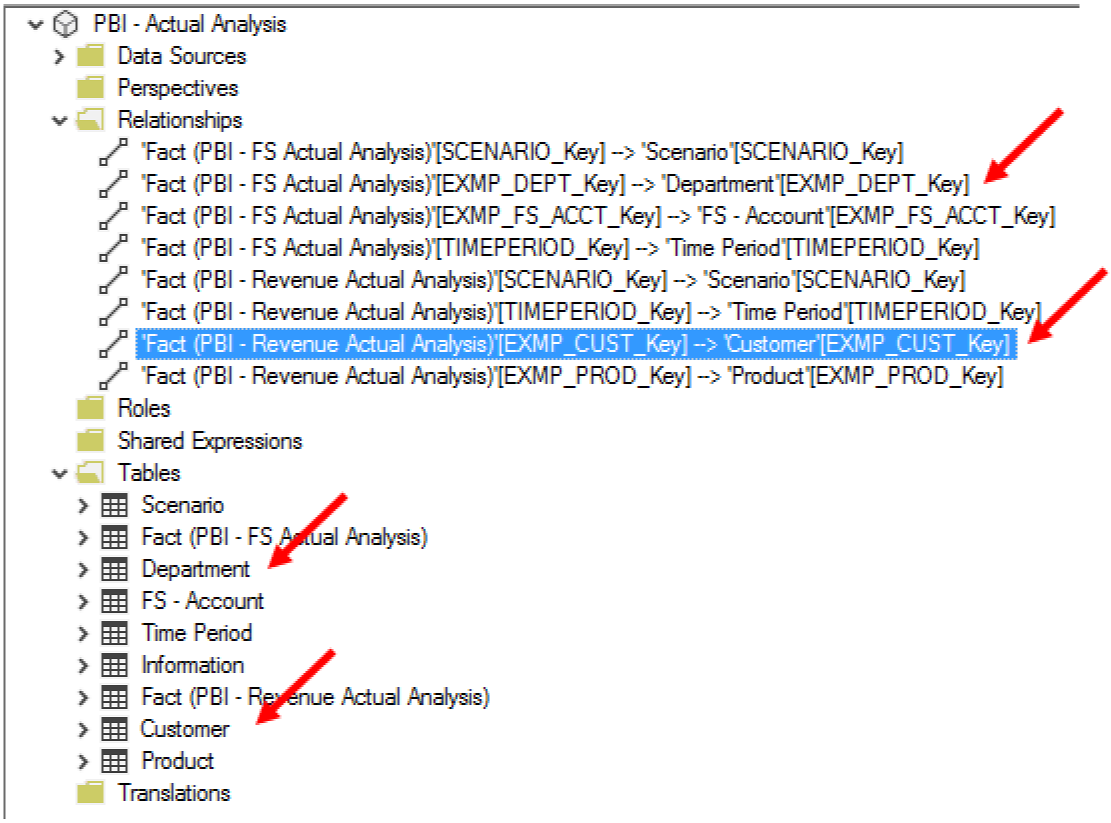
https://www.daxpatterns.com/parameter-table/
https://www.bluegranite.com/blog/disconnected-table-power-bi
https://radacad.com/change-the-column-or-measure-value-in-a-power-bi-visual-by-selection-of-the-slicer-parameter-table-pattern

Relationship Overrides

Relationships between tables in an AS Tabular model can be customized using the **Relationship Overrides** property of the exported table. The export process will automatically define the relationships between dimension and fact tables, and if any of those relationships is not desired, you can change them. The convention is noted below – note that the syntax uses AS table and column names – NOT relational table and column names.

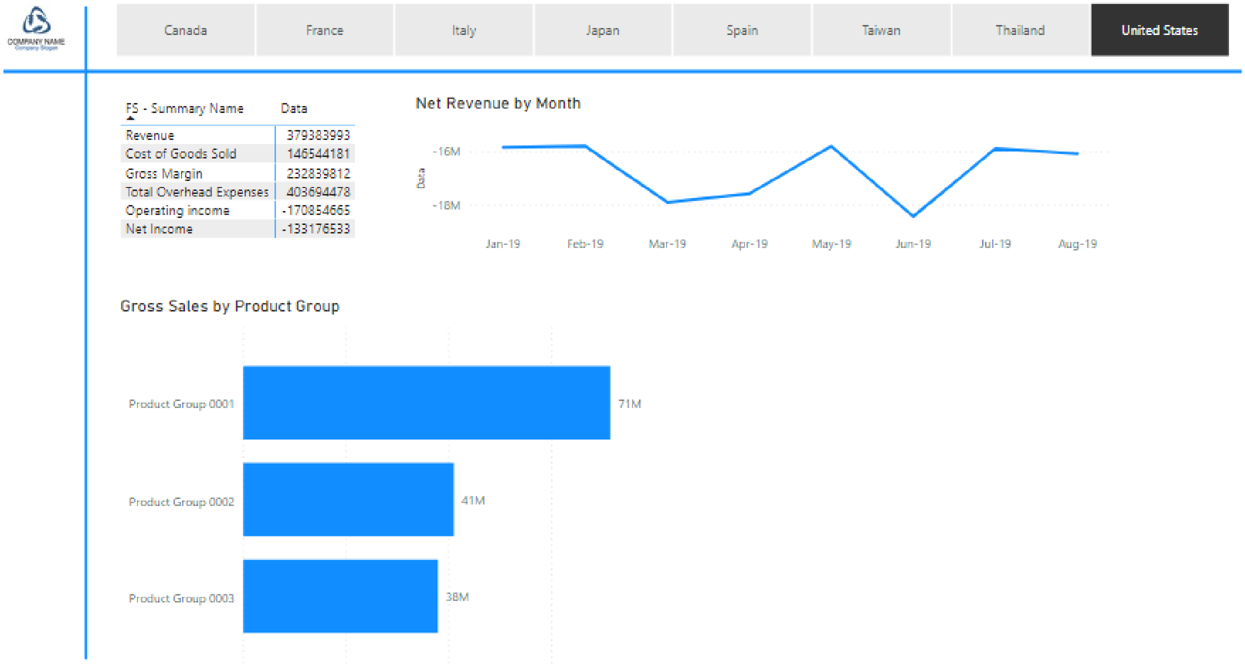
The **PBI – Actual Analysis** Power Pivot model is based on a view on the **EXMP – Financial Statements** model and has an additional view on the **Revenue – Reporting** model.

When the model is exported the relationships in AS Tabular model are defined as follows:

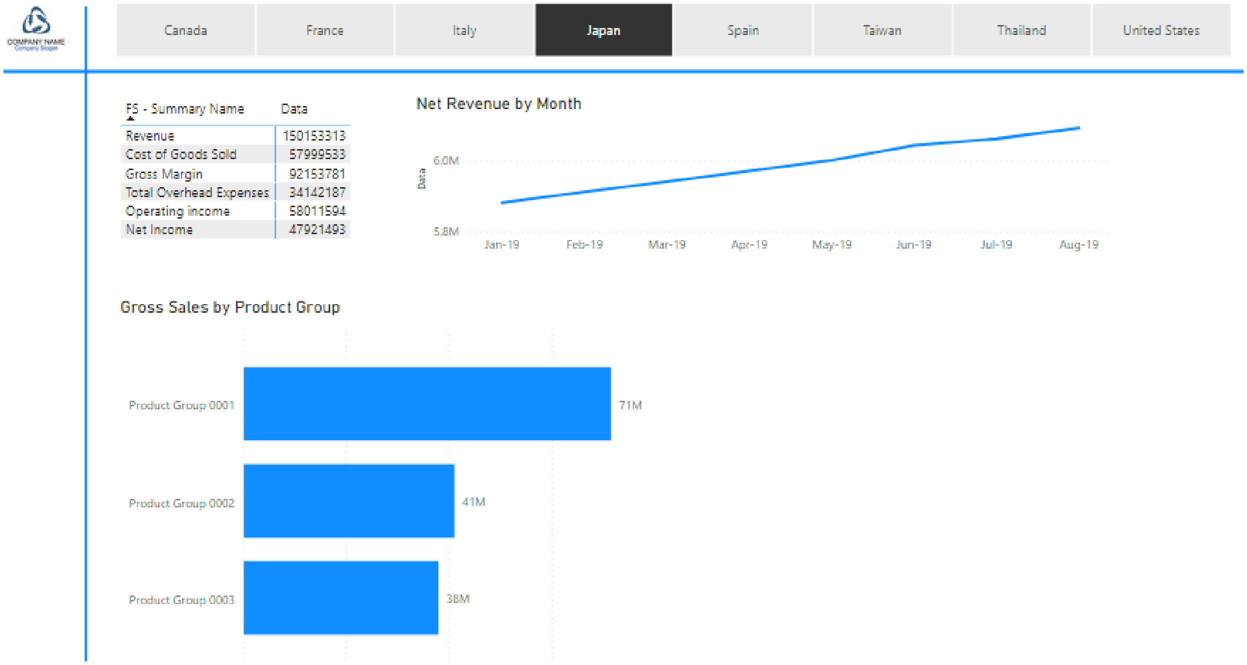


- PBI - Actual Analysis
  - Data Sources
  - Perspectives
  - Relationships
    - 'Fact (PBI - FS Actual Analysis)'[SCENARIO_Key] --> 'Scenario'[SCENARIO_Key]
    - 'Fact (PBI - FS Actual Analysis)'[EXMP_DEPT_Key] --> 'Department'[EXMP_DEPT_Key]
    - 'Fact (PBI - FS Actual Analysis)'[EXMP_FS_ACCT_Key] --> 'FS - Account'[EXMP_FS_ACCT_Key]
    - 'Fact (PBI - FS Actual Analysis)'[TIMEPERIOD_Key] --> 'Time Period'[TIMEPERIOD_Key]
    - 'Fact (PBI - Revenue Actual Analysis)'[SCENARIO_Key] --> 'Scenario'[SCENARIO_Key]
    - 'Fact (PBI - Revenue Actual Analysis)'[TIMEPERIOD_Key] --> 'Time Period'[TIMEPERIOD_Key]
    - 'Fact (PBI - Revenue Actual Analysis)'[EXMP_CUST_Key] --> 'Customer'[EXMP_CUST_Key]
    - 'Fact (PBI - Revenue Actual Analysis)'[EXMP_PROD_Key] --> 'Product'[EXMP_PROD_Key]
  - Roles
  - Shared Expressions
  - Tables
    - Scenario
    - Fact (PBI - FS Actual Analysis)
    - Department
    - FS - Account
    - Time Period
    - Information
    - Fact (PBI - Revenue Actual Analysis)
    - Customer
    - Product
  - Translations

Our Power BI report has a Slicer on **Country** field from the **Department** table. If you observe the two screenshots below changing the country in the slicer from United States to Japan has NO impact on the Gross Sales by Product Group visualization as it is based on Revenue model and there is no relationship between the **Revenue** Fact table and **Department** table.

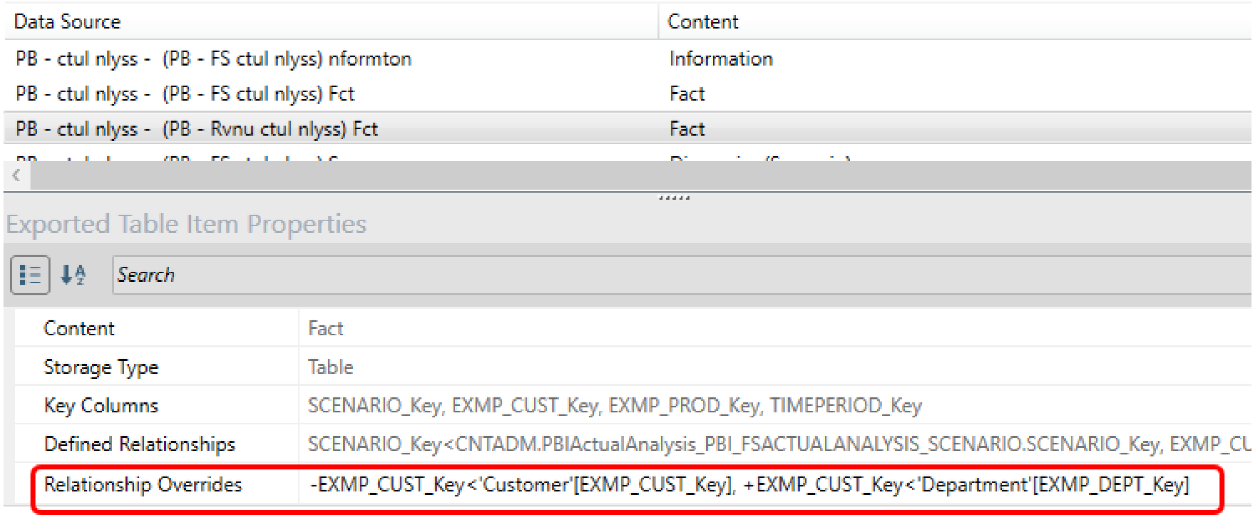


FS - Summary Name	Data
Revenue	379383993
Cost of Goods Sold	146544181
Gross Margin	232839812
Total Overhead Expenses	403694478
Operating income	-170854665
Net Income	-133176533



FS - Summary Name	Data
Revenue	150153313
Cost of Goods Sold	57999533
Gross Margin	92153781
Total Overhead Expenses	34142187
Operating income	58011594
Net Income	47921493

Within CONTROL® these two models are logically related based on the common level **Legal Entity** which is available in both Customer and Department dimensions and data transfer between these two models happens at this common level. Since the views used in the **Power Pivot** model have branches having **Legal Entity** at the lowest level, we could link the two models. Using the **Relationship Overrides** property we remove the relationship between the Revenue Fact table and Customer table and add a new relationship between the Revenue Fact table and the Department table.



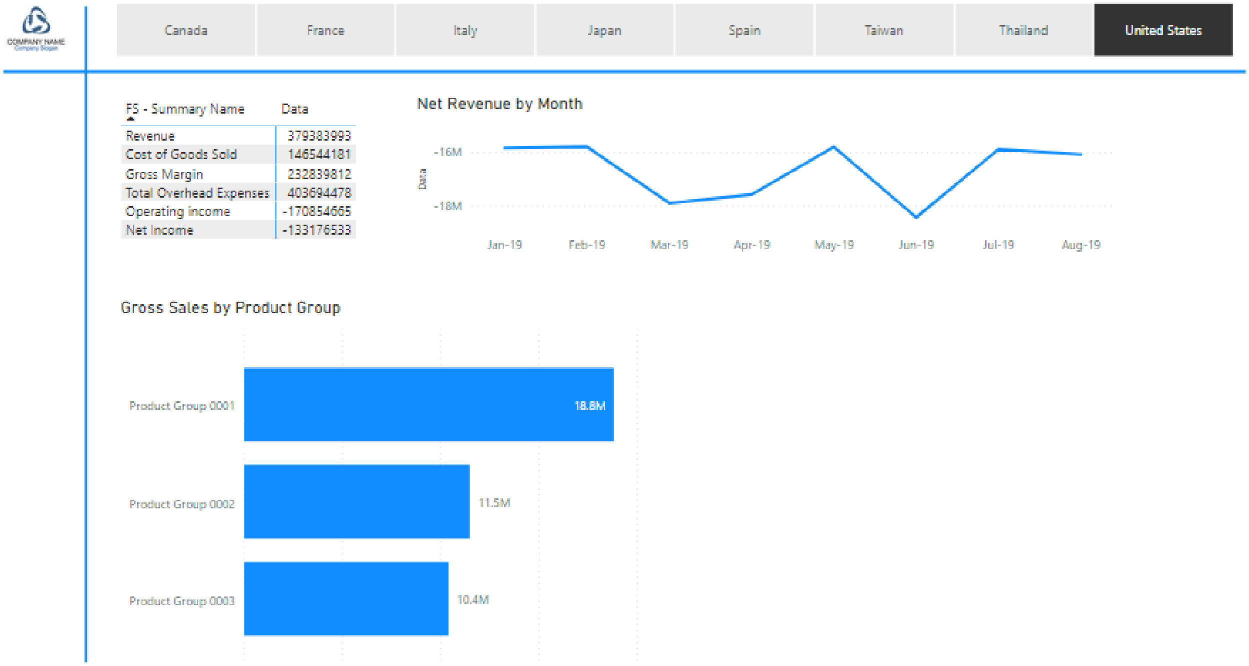
Data Source	Content
PB - ctul nlyss - (PB - FS ctul nlyss) nformton	Information
PB - ctul nlyss - (PB - FS ctul nlyss) Fct	Fact
PB - ctul nlyss - (PB - Rvnu ctul nlyss) Fct	Fact

Exported Table Item Properties

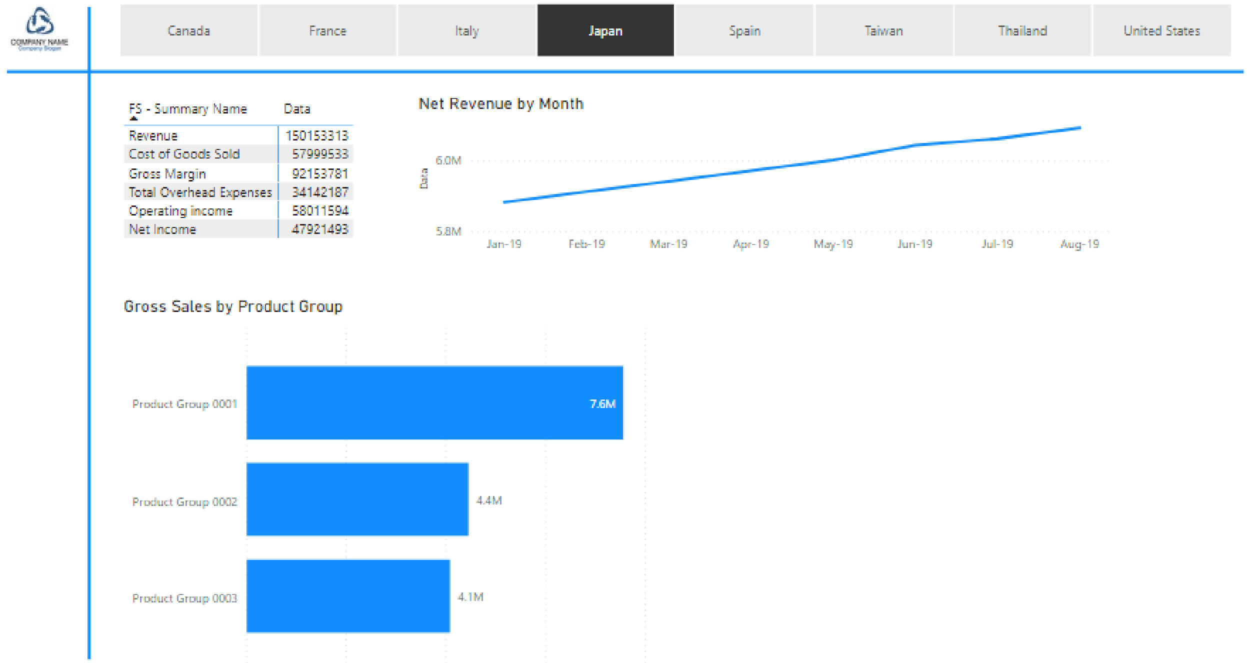
Content	Fact
Storage Type	Table
Key Columns	SCENARIO_Key, EXMP_CUST_Key, EXMP_PROD_Key, TIMEPERIOD_Key
Defined Relationships	SCENARIO_Key<CNTADM.PBIActualAnalysis_PBI_FSACTUALANALYSIS_SCENARIO.SCENARIO_Key, EXMP_CL
Relationship Overrides	-EXMP_CUST_Key<'Customer'[EXMP_CUST_Key], +EXMP_CUST_Key<'Department'[EXMP_DEPT_Key]

Since relationship overrides have an impact only on AS Tabular model you need to export the Power Pivot model using **Export > Update > All** to preserve the customization during the export**.**

Now, the Gross Sales by Product Group visualization reacts to the selection on the Slicer!



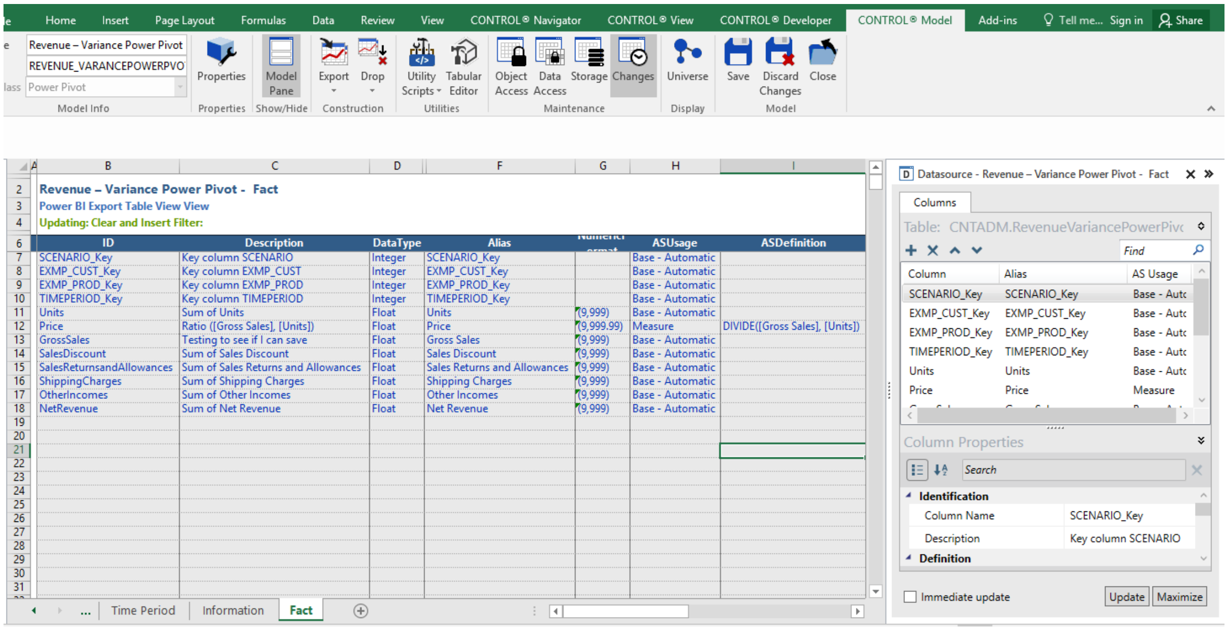
FS - Summary Name	Data
Revenue	379383993
Cost of Goods Sold	146544181
Gross Margin	232839812
Total Overhead Expenses	403694478
Operating income	-170854665
Net Income	-133176533



FS - Summary Name	Data
Revenue	150153313
Cost of Goods Sold	57999533
Gross Margin	92153781
Total Overhead Expenses	34142187
Operating income	58011594
Net Income	47921493

Datasources

Once you have run the export and reopen the edit book, you will see a worksheet in the workbook for every relational table or view that was exported. Each worksheet corresponds to a hidden, **CONTROL-Managed** datasource dedicated to the **Power Pivot** model.



Revenue – Variance Power Pivot
REVENUE_VARANCEPOWERPVO
Power Pivot

Revenue – Variance Power Pivot - Fact
Power BI Export Table View View
Updating: Clear and Insert Filter:

ID	Description	DataType	Alias	Numeric Format	ASUsage	ASDefinition
SCENARIO_Key	Key column SCENARIO	Integer	SCENARIO_Key		Base - Automatic	
EXMP_CUST_Key	Key column EXMP_CUST	Integer	EXMP_CUST_Key		Base - Automatic	
EXMP_PROD_Key	Key column EXMP_PROD	Integer	EXMP_PROD_Key		Base - Automatic	
TIMEPERIOD_Key	Key column TIMEPERIOD	Integer	TIMEPERIOD_Key		Base - Automatic	
Units	Sum of Units	Float	Units	(9,999)	Base - Automatic	
Price	Ratio ([Gross Sales], [Units])	Float	Price	(9,999.99)	Measure	DIVIDE([Gross Sales], [Units])
GrossSales	Testing to see if I can save	Float	Gross Sales	(9,999)	Base - Automatic	
SalesDiscount	Sum of Sales Discount	Float	Sales Discount	(9,999)	Base - Automatic	
SalesReturnsandAllowances	Sum of Sales Returns and Allowances	Float	Sales Returns and Allowances	(9,999)	Base - Automatic	
ShippingCharges	Sum of Shipping Charges	Float	Shipping Charges	(9,999)	Base - Automatic	
OtherIncomes	Sum of Other Incomes	Float	Other Incomes	(9,999)	Base - Automatic	
NetRevenue	Sum of Net Revenue	Float	Net Revenue	(9,999)	Base - Automatic	

Datasource - Revenue – Variance Power Pivot - Fact

Columns

Table: CNTADM.RevenueVariancePowerPivc

Column	Alias	AS Usage
SCENARIO_Key	SCENARIO_Key	Base - Autc
EXMP_CUST_Key	EXMP_CUST_Key	Base - Autc
EXMP_PROD_Key	EXMP_PROD_Key	Base - Autc
TIMEPERIOD_Key	TIMEPERIOD_Key	Base - Autc
Units	Units	Base - Autc
Price	Price	Measure

Column Properties

Identification

Column Name	SCENARIO_Key
Description	Key column SCENARIO

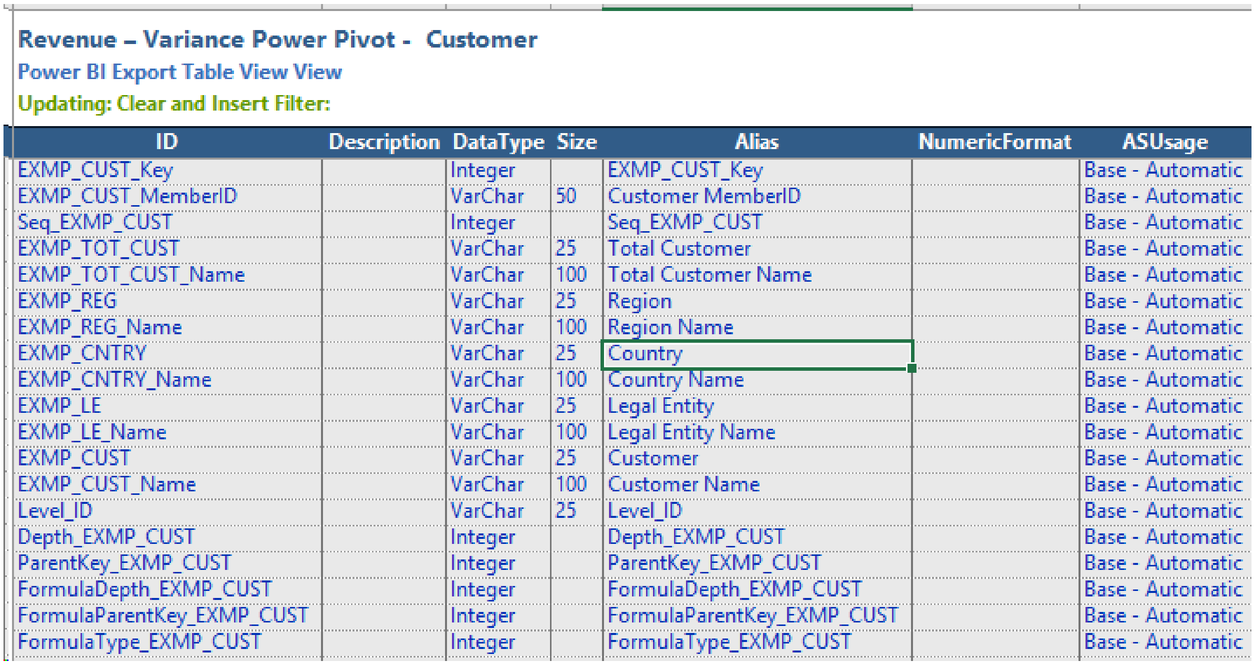
Definition

☐ Immediate update   Update   Maximize

Typically, you do not need to modify the properties of these tables or their columns. However, if you are exporting to Analysis Services, you may want to review and customize the following:

- **Description** defines the tool tip for the column in Power BI. It is the description defined in the level object's **Description** property.
- **Alias** is the name of the column or measure that will appear in Power BI. By default, CONTROL® uses level and dimension names as **Alias** for columns in the dimension tables during the export to Analysis Services. However, if you wish you can override the names by updating the **Alias** field. CONTROL® will automatically remove or replace any special characters not supported by AS.

For our sample **Power Pivot** model, the Customer dimensions' **Alias** are populated by default as follows during the export.



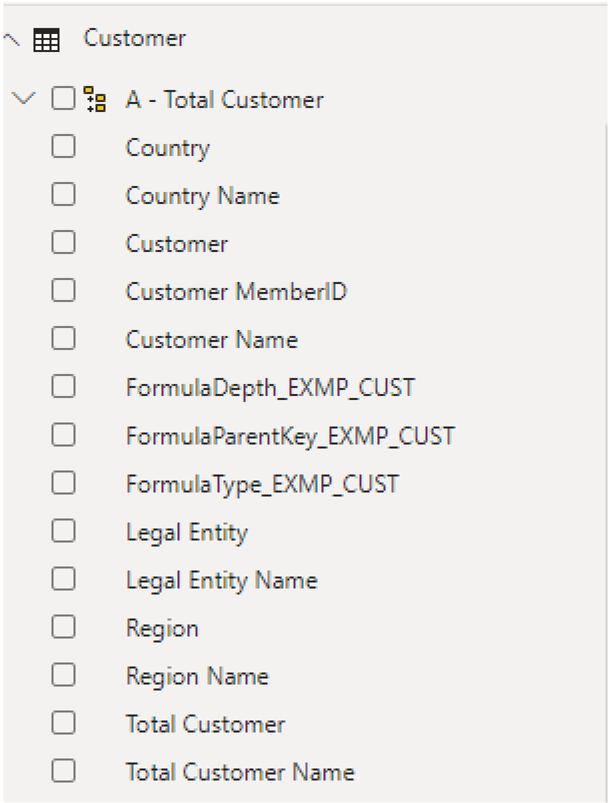
Revenue – Variance Power Pivot - Customer

Power BI Export Table View View

Updating: Clear and Insert Filter:

ID	Description	DataType	Size	Alias	NumericFormat	ASUsage
EXMP_CUST_Key		Integer		EXMP_CUST_Key		Base - Automatic
EXMP_CUST_MemberID		VarChar	50	Customer MemberID		Base - Automatic
Seq_EXMP_CUST		Integer		Seq_EXMP_CUST		Base - Automatic
EXMP_TOT_CUST		VarChar	25	Total Customer		Base - Automatic
EXMP_TOT_CUST_Name		VarChar	100	Total Customer Name		Base - Automatic
EXMP_REG		VarChar	25	Region		Base - Automatic
EXMP_REG_Name		VarChar	100	Region Name		Base - Automatic
EXMP_CNTRY		VarChar	25	Country		Base - Automatic
EXMP_CNTRY_Name		VarChar	100	Country Name		Base - Automatic
EXMP_LE		VarChar	25	Legal Entity		Base - Automatic
EXMP_LE_Name		VarChar	100	Legal Entity Name		Base - Automatic
EXMP_CUST		VarChar	25	Customer		Base - Automatic
EXMP_CUST_Name		VarChar	100	Customer Name		Base - Automatic
Level_ID		VarChar	25	Level_ID		Base - Automatic
Depth_EXMP_CUST		Integer		Depth_EXMP_CUST		Base - Automatic
ParentKey_EXMP_CUST		Integer		ParentKey_EXMP_CUST		Base - Automatic
FormulaDepth_EXMP_CUST		Integer		FormulaDepth_EXMP_CUST		Base - Automatic
FormulaParentKey_EXMP_CUST		Integer		FormulaParentKey_EXMP_CUST		Base - Automatic
FormulaType_EXMP_CUST		Integer		FormulaType_EXMP_CUST		Base - Automatic

and they appear in Power BI as follows:



Customer

□ A - Total Customer

□ Country

□ Country Name

□ Customer

□ Customer MemberID

□ Customer Name

□ FormulaDepth_EXMP_CUST

□ FormulaParentKey_EXMP_CUST

□ FormulaType_EXMP_CUST

□ Legal Entity

□ Legal Entity Name

□ Region

□ Region Name

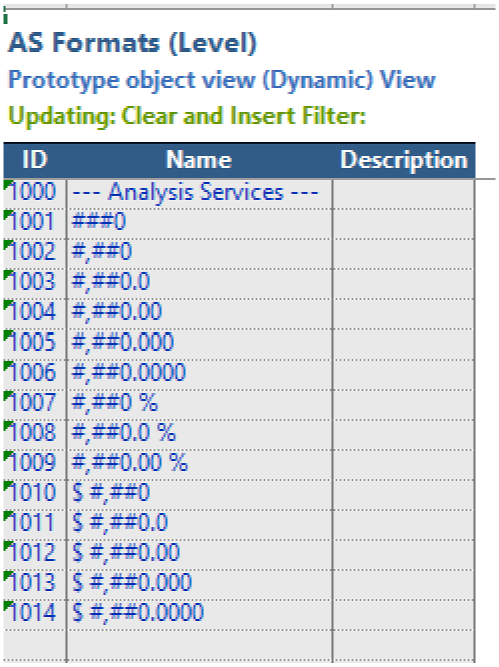
□ Total Customer

□ Total Customer Name

In rare cases where you override the **Alias** you need to use **Export > Update > All** to export your data and meta-data while keeping the customizations.

- **Numeric Format** is the numeric style used in tabular displays in Power BI. If you anchor the fact table on a variable or custom dimension, then the member's numeric format from the dimension defines the **Numeric Format**. If the fact table is unanchored or anchored on a different dimension, a default format is applied.

A new optional level **AS Formats (ID = ASFORMATS)** can be defined to define explicit formats that are usable directly in AS/Power BI. The **AS Formats** level must have numeric ID's and the ID's should have a minimum value of 1000, so as not to conflict with existing formats (which must still be available to support variables, custom dimensions, and existing data sources). Care should be taken that the two lists (**KCI_BrowserFormats** and **ASFormats**) don't intersect.

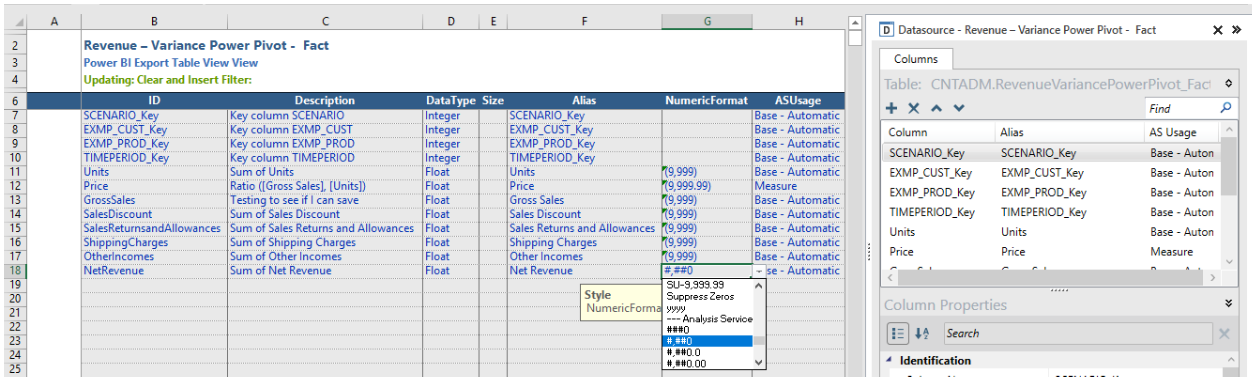


AS Formats (Level)

Prototype object view (Dynamic) View

Updating: Clear and Insert Filter:

ID	Name	Description
1000	--- Analysis Services ---	
1001	###0	
1002	#,##0	
1003	#,##0.0	
1004	#,##0.00	
1005	#,##0.000	
1006	#,##0.0000	
1007	#,##0 %	
1008	#,##0.0 %	
1009	#,##0.00 %	
1010	$ #,##0	
1011	$ #,##0.0	
1012	$ #,##0.00	
1013	$ #,##0.000	
1014	$ #,##0.0000	



Revenue – Variance Power Pivot - Fact

Power BI Export Table View View

Updating: Clear and Insert Filter:

ID	Description	DataType	Size	Alias	NumericFormat	ASUsage
SCENARIO_Key	Key column SCENARIO	Integer		SCENARIO_Key		Base - Automatic
EXMP_CUST_Key	Key column EXMP_CUST	Integer		EXMP_CUST_Key		Base - Automatic
EXMP_PROD_Key	Key column EXMP_PROD	Integer		EXMP_PROD_Key		Base - Automatic
TIMEPERIOD_Key	Key column TIMEPERIOD	Integer		TIMEPERIOD_Key		Base - Automatic
Units	Sum of Units	Float		Units	(9,999)	Base - Automatic
Price	Ratio ([Gross Sales], [Units])	Float		Price	(9,999.99)	Measure
GrossSales	Testing to see if I can save	Float		Gross Sales	(9,999)	Base - Automatic
SalesDiscount	Sum of Sales Discount	Float		Sales Discount	(9,999)	Base - Automatic
SalesReturnsandAllowances	Sum of Sales Returns and Allowances	Float		Sales Returns and Allowances	(9,999)	Base - Automatic
ShippingCharges	Sum of Shipping Charges	Float		Shipping Charges	(9,999)	Base - Automatic
OtherIncomes	Sum of Other Incomes	Float		Other Incomes	(9,999)	Base - Automatic
NetRevenue	Sum of Net Revenue	Float		Net Revenue	#,##0	se - Automatic

Datasource - Revenue – Variance Power Pivot - Fact

Table: CNTADM.RevenueVariancePowerPivot_Fact

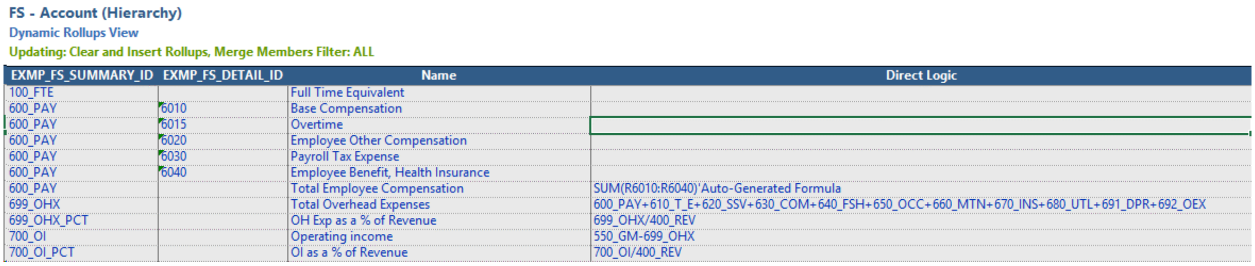
Column	Alias	AS Usage
SCENARIO_Key	SCENARIO_Key	Base - Auton
EXMP_CUST_Key	EXMP_CUST_Key	Base - Auton
EXMP_PROD_Key	EXMP_PROD_Key	Base - Auton
TIMEPERIOD_Key	TIMEPERIOD_Key	Base - Auton
Units	Units	Base - Auton
Price	Price	Measure

The dimension tables contain columns for all the levels related to the dimension branch in the CONTROL® view. Following additional columns are added to the materialized exported dimension tables to help create complex DAX expressions needed to handle parent-child hierarchies especially the ones related to the Financial Statement accounts/variables.

Column	Description
Depth	Depth of the member in the hierarchy. 1 for topmost level and incremental for each lower level.
ParentKey	Member key of the Parent to which the current member reports.

FormulaDepth	Depth of the member based on formula dependency.
FormulaParentKey	Member key of the Parent based on formula dependency.
FormulaType	0 - No formula 1 - Simple sum 2 - Addition and subtraction 3 - Simple ratio 4 – Complex
VarianceConvention	Account Convention used mainly for Variance calculation: 1 for Revenue -1 for Cost 0 for neither

Following screenshots show the Account hierarchy in CONTROL® and how the exported meta-data in the dimension relational table gets the values for the additional columns described above.

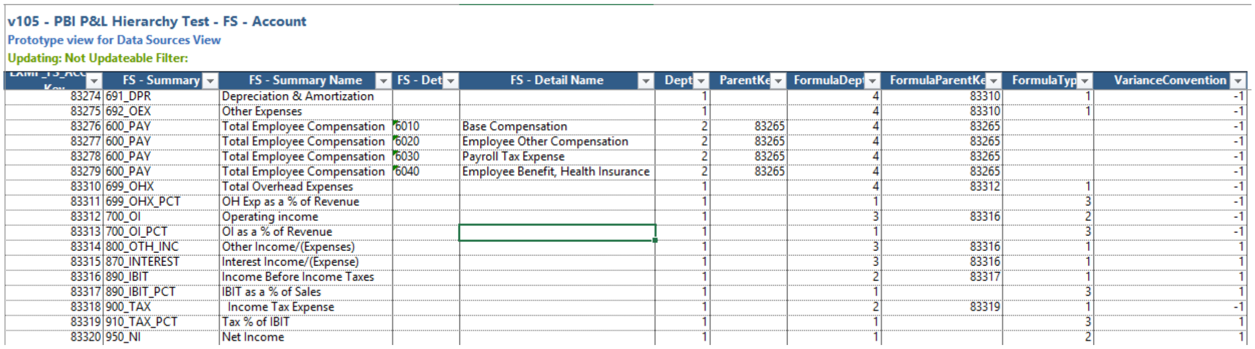


FS - Account (Hierarchy)

Dynamic Rollups View

Updating: Clear and Insert Rollups, Merge Members Filter: ALL

EXMP_FS_SUMMARY_ID	EXMP_FS_DETAIL_ID	Name	Direct Logic
100_FTE		Full Time Equivalent	
600_PAY	6010	Base Compensation	
600_PAY	6015	Overtime	
600_PAY	6020	Employee Other Compensation	
600_PAY	6030	Payroll Tax Expense	
600_PAY	6040	Employee Benefit, Health Insurance	
600_PAY		Total Employee Compensation	SUM(R6010:R6040)'Auto-Generated Formula
699_OHX		Total Overhead Expenses	600_PAY+610_T_E+620_SSV+630_COM+640_FSH+650_OCC+660_MTN+670_INS+680_UTL+691_DPR+692_OEX
699_OHX_PCT		OH Exp as a % of Revenue	699_OHX/400_REV
700_OI		Operating income	550_GM-699_OHX
700_OI_PCT		OI as a % of Revenue	700_OI/400_REV



v105 - PBI P&L Hierarchy Test - FS - Account

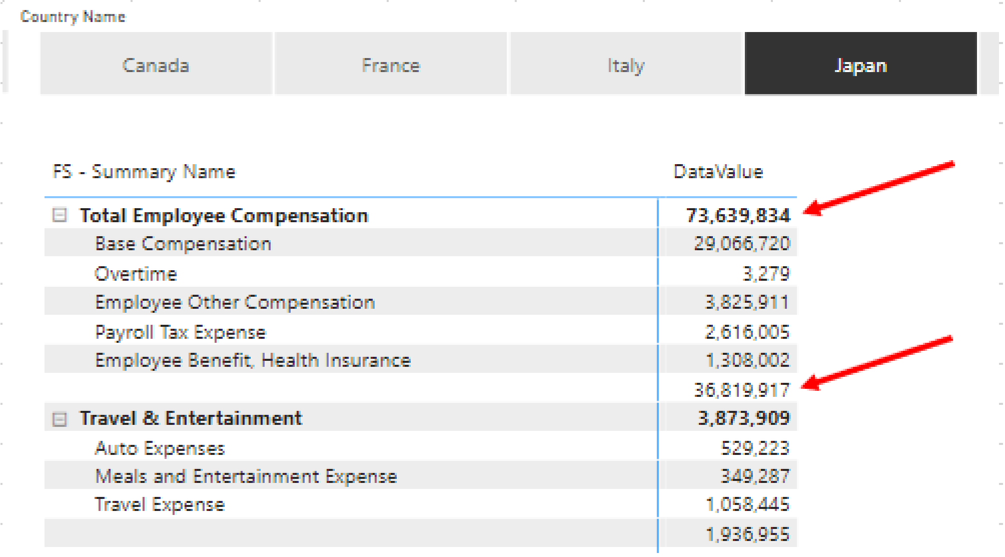
Prototype view for Data Sources View

Updating: Not Updateable Filter:

EXMP_FS_ACC Key	FS - Summary	FS - Summary Name	FS - Det	FS - Detail Name	Dept	ParentKe	FormulaDept	FormulaParentKe	FormulaTyp	VarianceConvention
83274	691_DPR	Depreciation & Amortization			1		4	83310	1	-1
83275	692_OEX	Other Expenses			1		4	83310	1	-1
83276	600_PAY	Total Employee Compensation	6010	Base Compensation	2	83265	4	83265		-1
83277	600_PAY	Total Employee Compensation	6020	Employee Other Compensation	2	83265	4	83265		-1
83278	600_PAY	Total Employee Compensation	6030	Payroll Tax Expense	2	83265	4	83265		-1
83279	600_PAY	Total Employee Compensation	6040	Employee Benefit, Health Insurance	2	83265	4	83265		-1
83310	699_OHX	Total Overhead Expenses			1		4	83312	1	-1
83311	699_OHX_PCT	OH Exp as a % of Revenue			1		1		3	-1
83312	700_OI	Operating income			1		3	83316	2	-1
83313	700_OI_PCT	OI as a % of Revenue			1		1		3	-1
83314	800_OTH_INC	Other Income/(Expenses)			1		3	83316	1	1
83315	870_INTEREST	Interest Income/(Expense)			1		3	83316	1	1
83316	890_IBIT	Income Before Income Taxes			1		2	83317	1	1
83317	890_IBIT_PCT	IBIT as a % of Sales			1		1		3	1
83318	900_TAX	Income Tax Expense			1		2	83319	1	-1
83319	910_TAX_PCT	Tax % of IBIT			1		1		3	1
83320	950_NI	Net Income			1		1		2	1

The following section describes an example where these additional columns can be used. The **Power Pivot** model **v105 - PBI P&L Hierarchy Demo** is based on a view that shows the Income Statement with the account hierarchy having both Summary and Detailed levels.

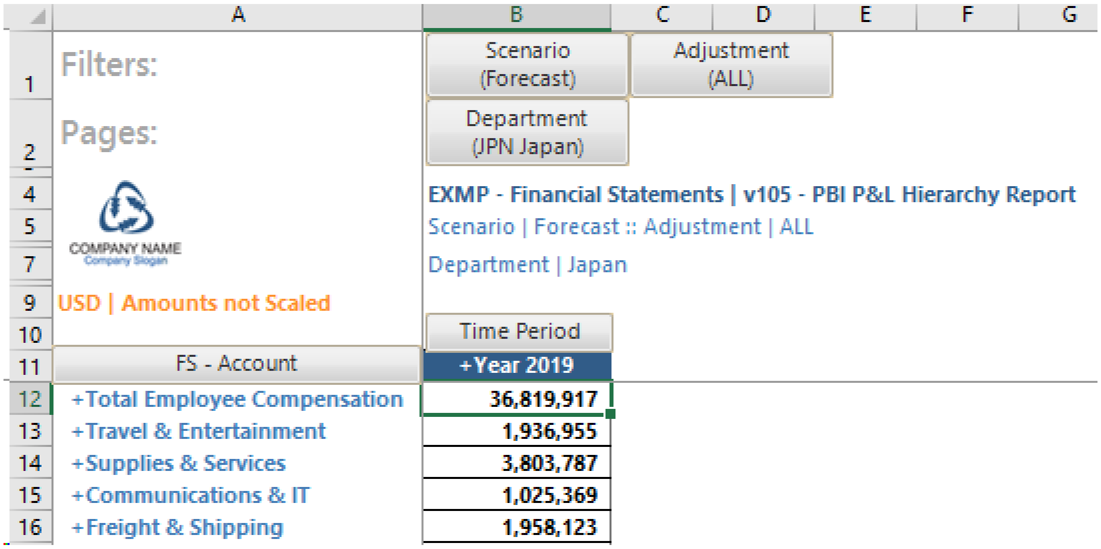
When you create a matrix visualization in Power BI you can see a problem. The **Total Employee Compensation** amount is twice when compared to CONTROL® data and you can also see a row with data but no heading.



Country Name

Canada | France | Italy | Japan

FS - Summary Name	DataValue
⊟ **Total Employee Compensation**	**73,639,834**
Base Compensation	29,066,720
Overtime	3,279
Employee Other Compensation	3,825,911
Payroll Tax Expense	2,616,005
Employee Benefit, Health Insurance	1,308,002
	36,819,917
⊟ **Travel & Entertainment**	**3,873,909**
Auto Expenses	529,223
Meals and Entertainment Expense	349,287
Travel Expense	1,058,445
	1,936,955



	A	B	C	D	E	F	G
1	Filters:	Scenario (Forecast)	Adjustment (ALL)				
2	Pages:	Department (JPN Japan)					
4		EXMP - Financial Statements | v105 - PBI P&L Hierarchy Report					
5	COMPANY NAME Company Slogan	Scenario | Forecast :: Adjustment | ALL					
7		Department | Japan					
9	USD | Amounts not Scaled						
10		Time Period					
11	FS - Account	+Year 2019					
12	+Total Employee Compensation	36,819,917					
13	+Travel & Entertainment	1,936,955					
14	+Supplies & Services	3,803,787					
15	+Communications & IT	1,025,369					
16	+Freight & Shipping	1,958,123					

We overcome these issues by creating DAX measures using the **Depth** column of the account table.

Browse Depth measure determines the level being browsed in the Matrix visualization.

```
ISINSCOPE ( 'FS - Account'[FS - Summary Name] ) + ISINSCOPE ( 'FS - Account'[FS - Detail Name] )
```

Show Row measure checks if the depth of the member being browsed is valid based on a check on the account hierarchy. We prevent the blank rows from showing data and being evaluated in computing numbers.

```
        [Browse Depth] <= Max('FS - Account'[Depth])
```

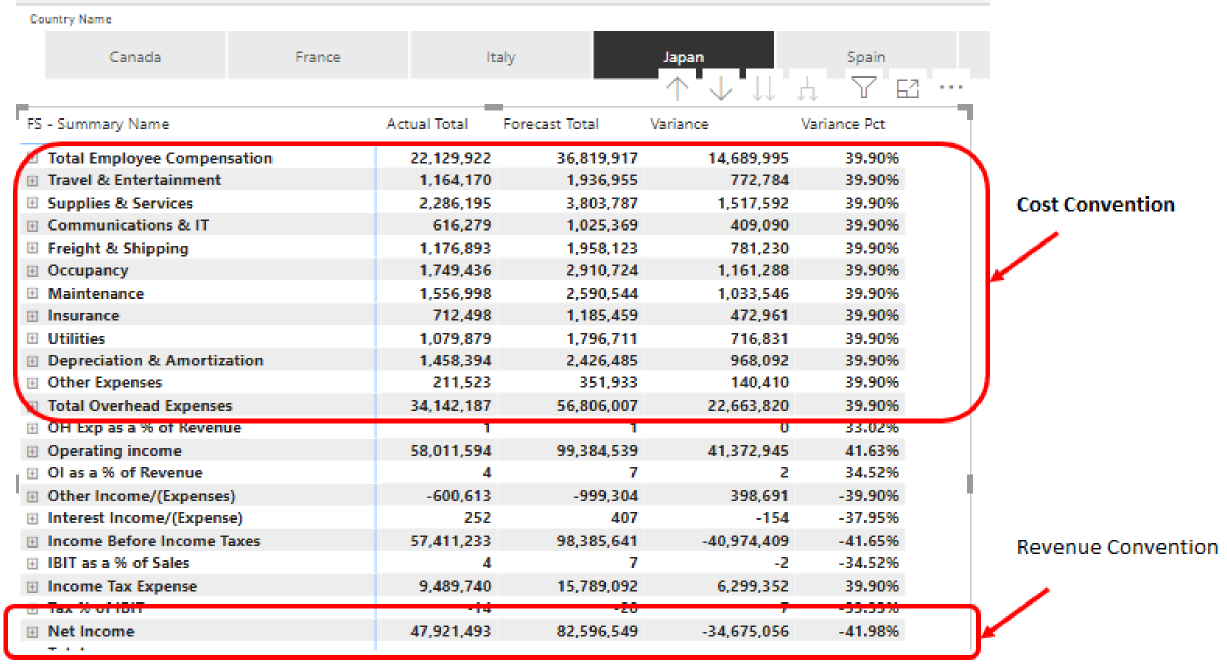
Actual Total :=

```
        VAR SummaryAccountSelected = ISINSCOPE ( 'FS - Account'[FS - Summary Name] )
        VAR SummaryAccountSum = CALCULATE ( [DataValue], 'FS - Account'[Depth] = 1,
        ALLEXCEPT ( 'FS - Account', 'FS - Account'[FS - Summary Name]),
        FILTER(ALL('Scenario'),'Scenario'[Scenario Name]="Actuals"))
        VAR DetailAccountSelected = ISINSCOPE ( 'FS - Account'[FS - Detail Name] )
        VAR DetailAccountSum = CALCULATE ( [DataValue], 'FS - Account'[Depth] = 2,
        ALLEXCEPT ( 'FS - Account', 'FS - Account'[FS - Detail Name]), FILTER(ALL('Scenario'),'Scenario'[Scenario
        Name]="Actuals"))
        RETURN
        SWITCH (
        TRUE (),
        DetailAccountSelected, IF([Show Row], DetailAccountSum),
        SummaryAccountSelected, IF([Show Row], SummaryAccountSum)
        )SummaryAccountSelected, IF([AccountShowRow], SummaryAccountSum)
        )
```

Forecast Total :=

```
VAR SummaryAccountSelected = ISINSCOPE ( 'FS - Account'[FS - Summary Name] )
VAR SummaryAccountSum = CALCULATE ( [DataValue], 'FS - Account'[Depth] = 1,
ALLEXCEPT ( 'FS - Account', 'FS - Account'[FS - Summary Name]), FILTER(ALL('Scenario'),'Scenario'[Scenario
Name]="Forecast"))
VAR DetailAccountSelected = ISINSCOPE ( 'FS - Account'[FS - Detail Name] )
VAR DetailAccountSum = CALCULATE ( [DataValue], 'FS - Account'[Depth] = 2,
ALLEXCEPT ( 'FS - Account', 'FS - Account'[FS - Detail Name]), FILTER(ALL('Scenario'),'Scenario'[Scenario
Name]="Forecast"))
RETURN
SWITCH (
TRUE (),
DetailAccountSelected, IF([Show Row], DetailAccountSum),
SummaryAccountSelected, IF([Show Row], SummaryAccountSum)
)
```

The measures above determine which values to display based on the level of detail. Now, using these DAX measures the Matrix visualization produces the right results.



FS - Summary Name	Actual Total	Forecast Total	Variance	Variance Pct
Total Employee Compensation	22,129,922	36,819,917	14,689,995	39.90%
Travel & Entertainment	1,164,170	1,936,955	772,784	39.90%
Supplies & Services	2,286,195	3,803,787	1,517,592	39.90%
Communications & IT	616,279	1,025,369	409,090	39.90%
Freight & Shipping	1,176,893	1,958,123	781,230	39.90%
Occupancy	1,749,436	2,910,724	1,161,288	39.90%
Maintenance	1,556,998	2,590,544	1,033,546	39.90%
Insurance	712,498	1,185,459	472,961	39.90%
Utilities	1,079,879	1,796,711	716,831	39.90%
Depreciation & Amortization	1,458,394	2,426,485	968,092	39.90%
Other Expenses	211,523	351,933	140,410	39.90%
Total Overhead Expenses	34,142,187	56,806,007	22,663,820	39.90%
OH Exp as a % of Revenue	1	1	0	33.02%
Operating income	58,011,594	99,384,539	41,372,945	41.63%
OI as a % of Revenue	4	7	2	34.52%
Other Income/(Expenses)	-600,613	-999,304	398,691	-39.90%
Interest Income/(Expense)	252	407	-154	-37.95%
Income Before Income Taxes	57,411,233	98,385,641	-40,974,409	-41.65%
IBIT as a % of Sales	4	7	-2	-34.52%
Income Tax Expense	9,489,740	15,789,092	6,299,352	39.90%
Tax % of IBIT	-14	-28	7	-33.33%
Net Income	47,921,493	82,596,549	-34,675,056	-41.98%

Cost Convention

Revenue Convention

For computing the Variance we have used the **Variance Convention** property to calculate the variance correctly for Cost and Revenue accounts.

Variance :=

```
IF
(
    OR (MAX ('FS - Account'[VarianceConvention]) = 1, MAX ('FS - Account'[VarianceConvention]) = 0),
    [Actual Total] -   [Forecast Total],
    [Forecast Total] - [Actual Total]
)
```

There are several articles related to Parent-Child hierarchy-based reporting in Power BI. You can refer the following articles for more details

- https://www.daxpatterns.com/parent-child-hierarchies/
- https://powerpivotpro.com/2018/01/star-schema-switch-drill-income-statement-design/
- https://powerpivotpro.com/2011/09/profit-lossthe-art-of-the-cascading-subtotal/

AS Usage and definition

AS Usage and **AS Definition** properties define the measure calculations in AS and Power BI using DAX – the calculation language for AS Tabular.

Sum is the default calculation for each data column in the fact table, which is appropriate for most CONTROL® data, but not all. For example, ratios such as **Percent of Sales** need a special calculation.

The table below defines the options for the **AS Usage** property.

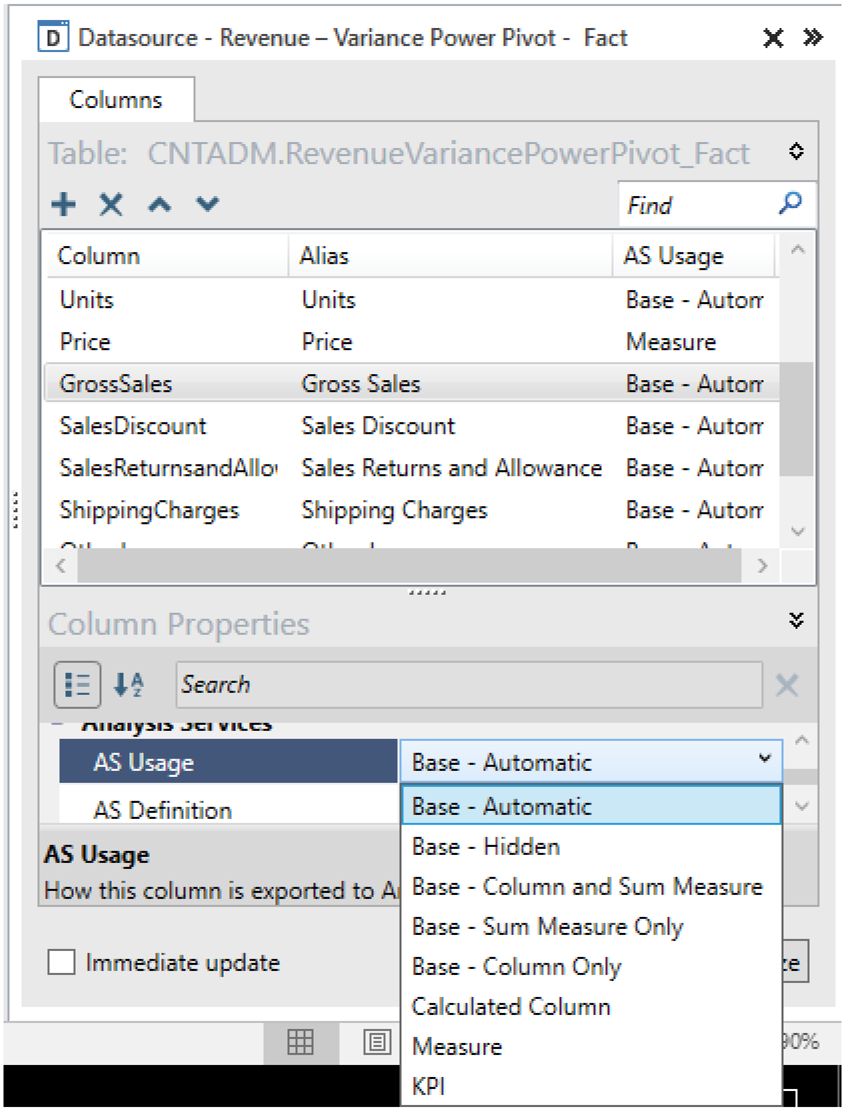


Table: CNTADM.RevenueVariancePowerPivot_Fact

Column	Alias	AS Usage
Units	Units	Base - Autom
Price	Price	Measure
GrossSales	Gross Sales	Base - Autom
SalesDiscount	Sales Discount	Base - Autom
SalesReturnsandAllo	Sales Returns and Allowance	Base - Autom
ShippingCharges	Shipping Charges	Base - Autom

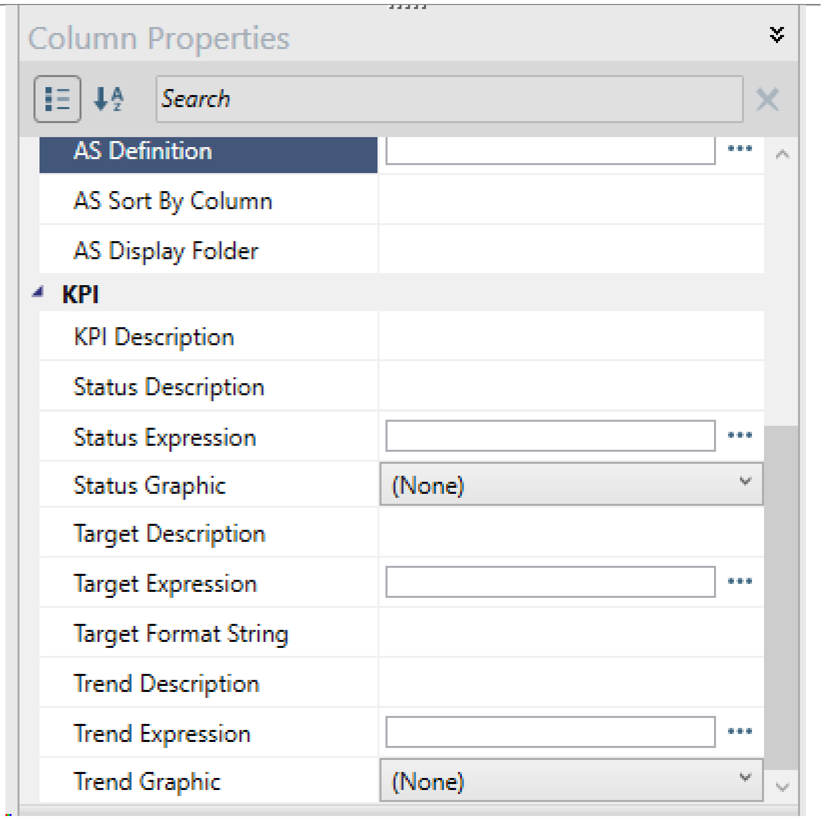
AS Usage: Base - Automatic

AS Definition

AS Usage
How this column is exported to A

AS Usage Property	Description
Base – Automatic	Automatically defines a column or measure of the base table as summation, depending on whether the Analysis Services Usage is **Export to AS Power BI and Excel (Recommended)** or **Export to AS Power BI**, and whether one or more

	calculation groups are present. This is the default option, and it lets the export process determine the most appropriate way to export the data in the column.
Base – Hidden	Hides the column from appearing in Power BI.
Base - Column and Sum Measure	Shows the column and creates a **Sum** measure.
Base - Sum Measure Only	Creates a **Sum** measure and hides the column.
Base - Column Only	Shows the column, which in Power BI, permits a **Sum** measure to be implied automatically.
Calculated Column	Creates a calculated column based on the supplied DAX expression and adds it to the table.
Measure	Creates a measure based on the supplied DAX expression and adds it to the table.
KPI	A Key Performance Indicator (KPI) is a visual cue that communicates the amount of progress made toward a measurable goal. To support the creation of KPIs for CONTROL® **Power Pivot** models' the following properties are provided:



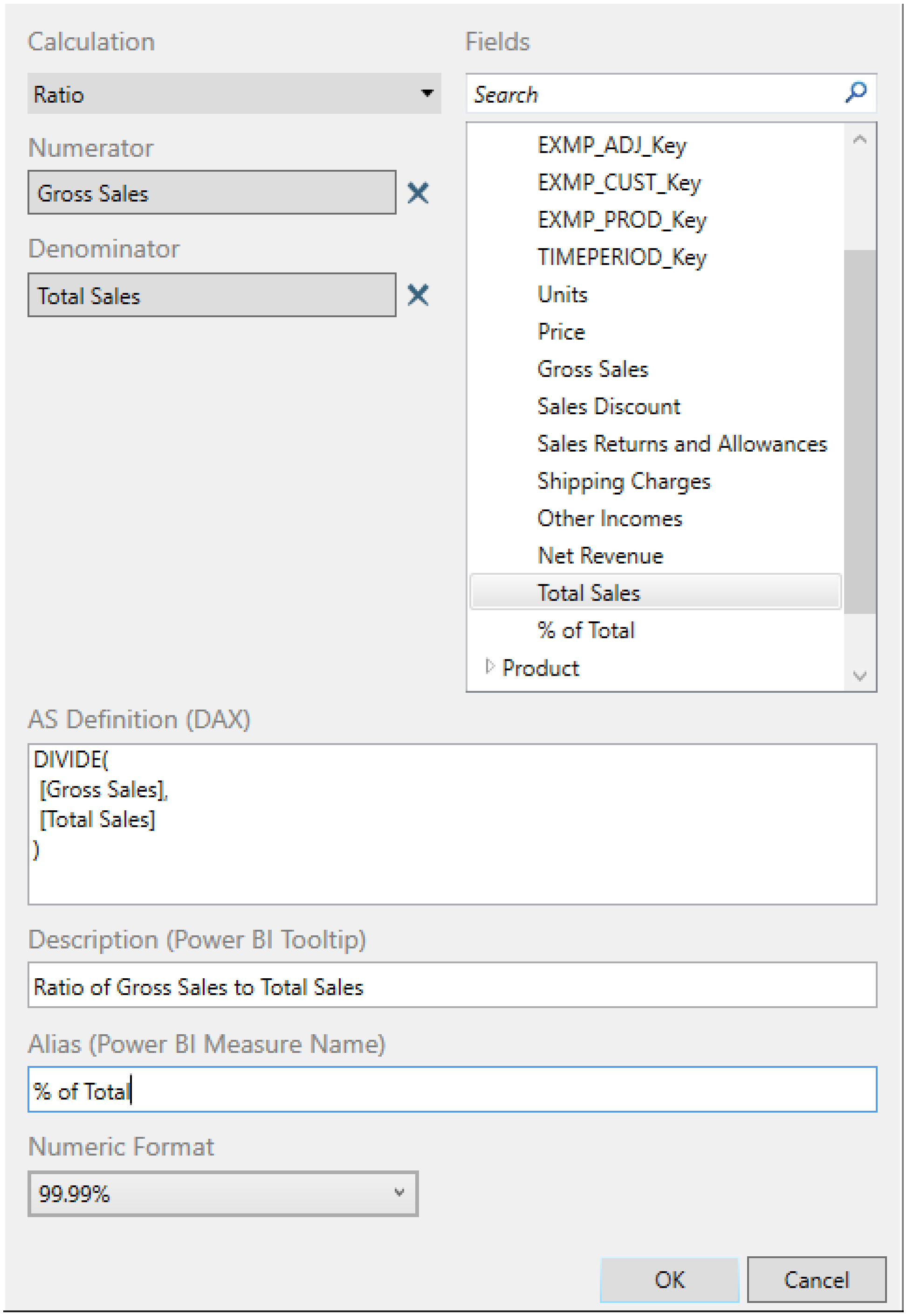
Column Properties

Search

AS Definition	
AS Sort By Column	
AS Display Folder	
KPI	
KPI Description	
Status Description	
Status Expression	
Status Graphic	(None)
Target Description	
Target Expression	
Target Format String	
Trend Description	
Trend Expression	
Trend Graphic	(None)

Only **Calculated Column**, **Measure** or **KPI** options require a DAX expression. When you select any one of these in the **AS Usage** property dropdown, and then click on the ellipsis button on the **AS Definition** field, you get a **Quick Measures** dialogue. The **Calculation** dropdown has a list of common types of formulas; most of them are from the **Quick Measures** available in Power BI. Select the required DAX formula from the dropdown and fill in the arguments by dragging and dropping the data fields from the **Power Pivot** model's fact or dimension tables available on the right on the dialogue or directly enter the arguments in the parameter fields.

Here is how the **Quick Measures** dialogue looks like in the **Power Pivot** model edit book:



Calculation: Ratio

Numerator: Gross Sales

Denominator: Total Sales

AS Definition (DAX)

```
DIVIDE(
 [Gross Sales],
 [Total Sales]
)
```

Description (Power BI Tooltip): Ratio of Gross Sales to Total Sales

Alias (Power BI Measure Name): % of Total

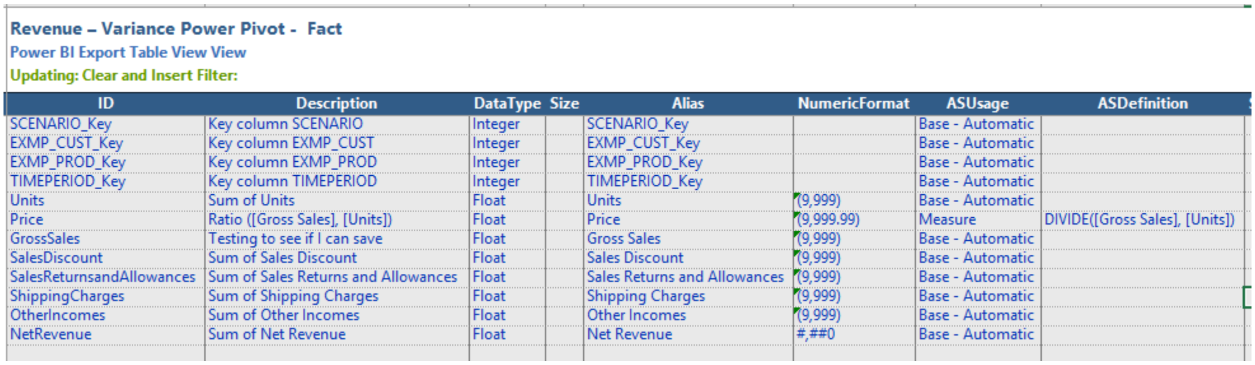
Numeric Format: 99.99%

After defining the arguments click **OK**.

Since the DAX expression executes in Analysis Services or Power BI the DAX expressions in your **Calculated Column**, **Measure** or **KPI** should reference columns by their **Alias** and not the **Column ID**. In the DAX expression above, **Gross Sales** is the **Alias** while the **Column ID** is **GrossSales**. Since the formula was defined using the Quick Measure dialogue it is already ensured as the Field list is based on the **Alias**. However, if you are entering the formula manually directly in the **AS definition** field either in the column properties task pane or the object view then you need to make sure you reference the **Alias**.

You can add new measures to the column list and hide existing columns if you do not want them to appear in the Power BI design pane.

For convenience, you can enter directly in the object view worksheet, if you need to make many customizations:



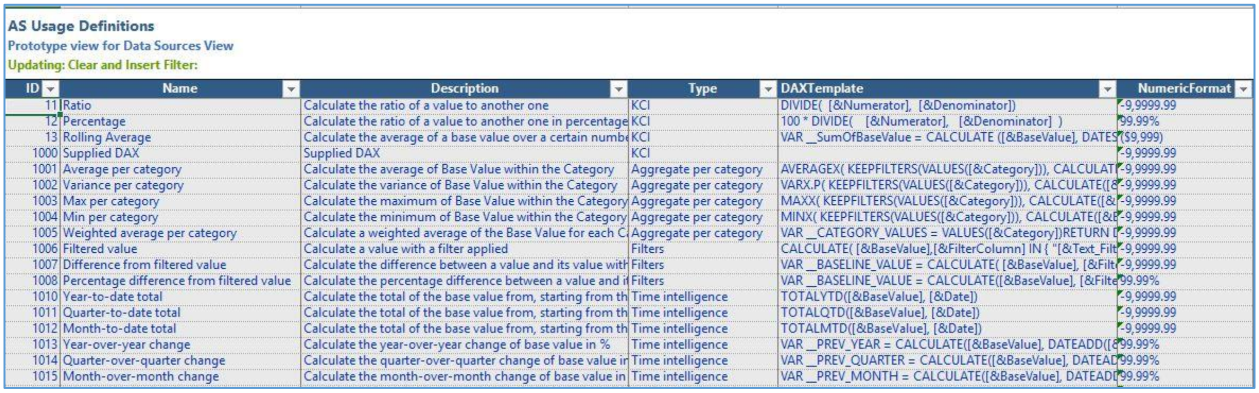
Revenue – Variance Power Pivot - Fact
Power BI Export Table View View
Updating: Clear and Insert Filter:

ID	Description	DataType	Size	Alias	NumericFormat	ASUsage	ASDefinition
SCENARIO_Key	Key column SCENARIO	Integer		SCENARIO_Key		Base - Automatic	
EXMP_CUST_Key	Key column EXMP_CUST	Integer		EXMP_CUST_Key		Base - Automatic	
EXMP_PROD_Key	Key column EXMP_PROD	Integer		EXMP_PROD_Key		Base - Automatic	
TIMEPERIOD_Key	Key column TIMEPERIOD	Integer		TIMEPERIOD_Key		Base - Automatic	
Units	Sum of Units	Float		Units	(9,999)	Base - Automatic	
Price	Ratio ([Gross Sales], [Units])	Float		Price	(9,999.99)	Measure	DIVIDE([Gross Sales], [Units])
GrossSales	Testing to see if I can save	Float		Gross Sales	(9,999)	Base - Automatic	
SalesDiscount	Sum of Sales Discount	Float		Sales Discount	(9,999)	Base - Automatic	
SalesReturnsandAllowances	Sum of Sales Returns and Allowances	Float		Sales Returns and Allowances	(9,999)	Base - Automatic	
ShippingCharges	Sum of Shipping Charges	Float		Shipping Charges	(9,999)	Base - Automatic	
OtherIncomes	Sum of Other Incomes	Float		Other Incomes	(9,999)	Base - Automatic	
NetRevenue	Sum of Net Revenue	Float		Net Revenue	#,##0	Base - Automatic	

For some people, writing a DAX formula for a measure might be easy. However, for others it can be quite a challenge.

Therefore, CONTROL® supplies a list of common types of formulas; most of them are from the **Quick Measures** available in Power BI.

The CONTROL® datasource named **AS Usage Definitions** stores all these predefined DAX formulas. You can add additional formulas to this datasource as they become available.



AS Usage Definitions
Prototype view for Data Sources View
Updating: Clear and Insert Filter:

ID	Name	Description	Type	DAXTemplate	NumericFormat
11	Ratio	Calculate the ratio of a value to another one	KCI	DIVIDE([&Numerator], [&Denominator])	-9,9999.99
12	Percentage	Calculate the ratio of a value to another one in percentage	KCI	100 * DIVIDE([&Numerator], [&Denominator])	99.99%
13	Rolling Average	Calculate the average of a base value over a certain numbe	KCI	VAR __SumOfBaseValue = CALCULATE ([&BaseValue], DATES	($9,999)
1000	Supplied DAX	Supplied DAX	KCI		-9,9999.99
1001	Average per category	Calculate the average of Base Value within the Category	Aggregate per category	AVERAGEX(KEEPFILTERS(VALUES([&Category])), CALCULAT	-9,9999.99
1002	Variance per category	Calculate the variance of Base Value within the Category	Aggregate per category	VARX.P(KEEPFILTERS(VALUES([&Category])), CALCULATE([&	-9,9999.99
1003	Max per category	Calculate the maximum of Base Value within the Category	Aggregate per category	MAXX(KEEPFILTERS(VALUES([&Category])), CALCULATE([&	-9,9999.99
1004	Min per category	Calculate the minimum of Base Value within the Category	Aggregate per category	MINX(KEEPFILTERS(VALUES([&Category])), CALCULATE([&E	-9,9999.99
1005	Weighted average per category	Calculate a weighted average of the Base Value for each C	Aggregate per category	VAR __CATEGORY_VALUES = VALUES([&Category])RETURN [	-9,9999.99
1006	Filtered value	Calculate a value with a filter applied	Filters	CALCULATE([&BaseValue],[&FilterColumn] IN { "[&Text_Filt	-9,9999.99
1007	Difference from filtered value	Calculate the difference between a value and its value with	Filters	VAR __BASELINE_VALUE = CALCULATE([&BaseValue], [&Filt	-9,9999.99
1008	Percentage difference from filtered value	Calculate the percentage difference between a value and it	Filters	VAR __BASELINE_VALUE = CALCULATE([&BaseValue], [&Filte	99.99%
1010	Year-to-date total	Calculate the total of the base value from, starting from th	Time intelligence	TOTALYTD([&BaseValue], [&Date])	-9,9999.99
1011	Quarter-to-date total	Calculate the total of the base value from, starting from th	Time intelligence	TOTALQTD([&BaseValue], [&Date])	-9,9999.99
1012	Month-to-date total	Calculate the total of the base value from, starting from th	Time intelligence	TOTALMTD([&BaseValue], [&Date])	-9,9999.99
1013	Year-over-year change	Calculate the year-over-year change of base value in %	Time intelligence	VAR __PREV_YEAR = CALCULATE([&BaseValue], DATEADD([&	99.99%
1014	Quarter-over-quarter change	Calculate the quarter-over-quarter change of base value ir	Time intelligence	VAR __PREV_QUARTER = CALCULATE([&BaseValue], DATEAD	99.99%
1015	Month-over-month change	Calculate the month-over-month change of base value in	Time intelligence	VAR __PREV_MONTH = CALCULATE([&BaseValue], DATEADD	99.99%

To avoid conflicts with future updates of CONTROL®, the CONTROL® specific formulas have IDs 0-999, and Microsoft's Quick Measures have IDs 1001-1999. Please use ID values of 2001 or greater for site-specific formulas.

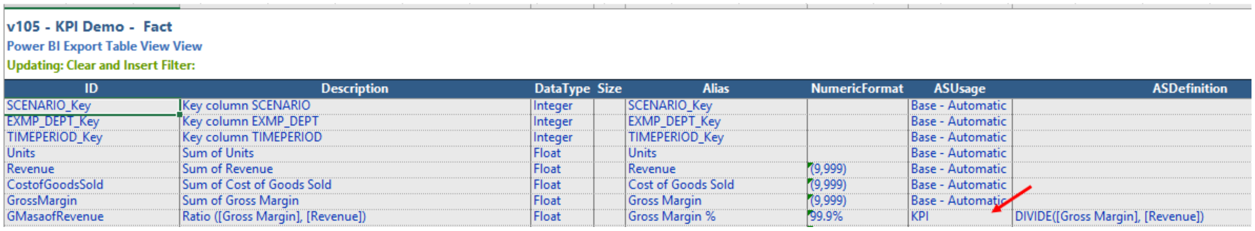
It is important to understand that you do not need to use a formula from this table if you want a custom DAX calculation for a measure. This table is for users who are not DAX experts to take advantage of calculations that others have created and shared.

Note: If the supplied expression defines the measure name (measureName = …), the name supplied in the expression will override the alias or column name as the name of the measure.

KPI

The sample model **v105 – KPI Demo** demonstrates creating a KPI in a **Power Pivot** model.

1. A KPI is always based on a measure.
2. The sample model is anchored on the Variable dimension.
3. When you create a KPI, you replace a measure with the KPI. In our **Power Pivot** model, we changed the Gross Margin % measure's **AS Usage** property from **Measure** to **KPI.**



v105 - KPI Demo - Fact
Power BI Export Table View View
Updating: Clear and Insert Filter:

ID	Description	DataType	Size	Alias	NumericFormat	ASUsage	ASDefinition
SCENARIO_Key	Key column SCENARIO	Integer		SCENARIO_Key		Base - Automatic	
EXMP_DEPT_Key	Key column EXMP_DEPT	Integer		EXMP_DEPT_Key		Base - Automatic	
TIMEPERIOD_Key	Key column TIMEPERIOD	Integer		TIMEPERIOD_Key		Base - Automatic	
Units	Sum of Units	Float		Units		Base - Automatic	
Revenue	Sum of Revenue	Float		Revenue	(9,999)	Base - Automatic	
CostofGoodsSold	Sum of Cost of Goods Sold	Float		Cost of Goods Sold	(9,999)	Base - Automatic	
GrossMargin	Sum of Gross Margin	Float		Gross Margin	(9,999)	Base - Automatic	
GMasaofRevenue	Ratio ([Gross Margin], [Revenue])	Float		Gross Margin %	99.9%	KPI	DIVIDE([Gross Margin], [Revenue])

4. There are basically three properties: Status, Target, and Trend. These are the internal Tabular names for Status, Goal, and Trend. Status and Trend let you define a description, expression, and graphic to use. The Target (which is called Goal in Power BI) has a format string and no graphic. Status and Trend are graphical items, whereas Target is a number.
5. The measure itself becomes the **value** of the KPI. In our case the **AS Definition** had already a DAX expression to calculate the Gross Margin %. The newly created measures are named _***Gross Margin % Goal***, _***Gross Margin % Trend***, and _***Gross Margin % Status***. They act as regular measures; you can call them in your DAX code and use them in reports.
6. Find below the values for the various properties for our KPI:

Status Expression

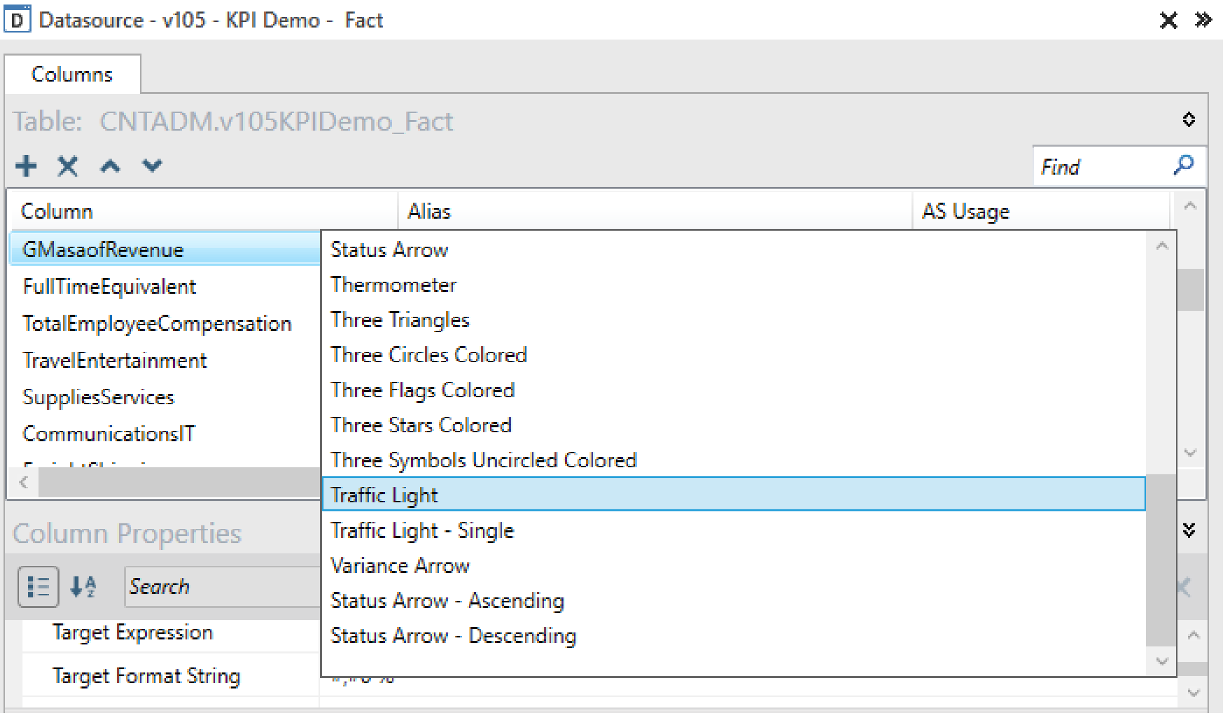
```
-- DAX code for Status Expression
VAR MarginPercentage = [Gross Margin %]
VAR MarginTolerance = 0.02
VAR MarginGoal = [_Gross Margin % Goal]
RETURN
   IF (NOT ISBLANK ( MarginPercentage ),
      SWITCH (TRUE,
            MarginPercentage < MarginGoal - MarginTolerance, -1,     -- Negative
            MarginPercentage > MarginGoal + MarginTolerance, 1,      -- Positive
            0
```

```
        )
    )
```

Observe that the DAX expression above refers the _**Margin % Goal** measure to obtain the target of the KPI. By comparing the value of the KPI against the target, we return -1 for bad, 1 for good, and 0 for average.

Status Graphic

The Status graphic and Trend graphic properties are enumerations and can be specified by selecting a value from the Drop down either in the task pane or the object view. We have set the Status Graphic to **Traffic Light**.



Target Expression

CALCULATE([Gross Margin %], FILTER(ALL('Scenario'), 'Scenario'[Scenario Name]="Forecast"))

Target Format String

Since the Target is a value, we specify a format string. In our case Gross Margin % is a Percentage and we specify the format string as **#,#0 %** which is the DAX format for percentage.

Refer the following link for more details on the format string specifications.
https://docs.microsoft.com/en-us/dax/custom-numeric-formats-for-the-format-function

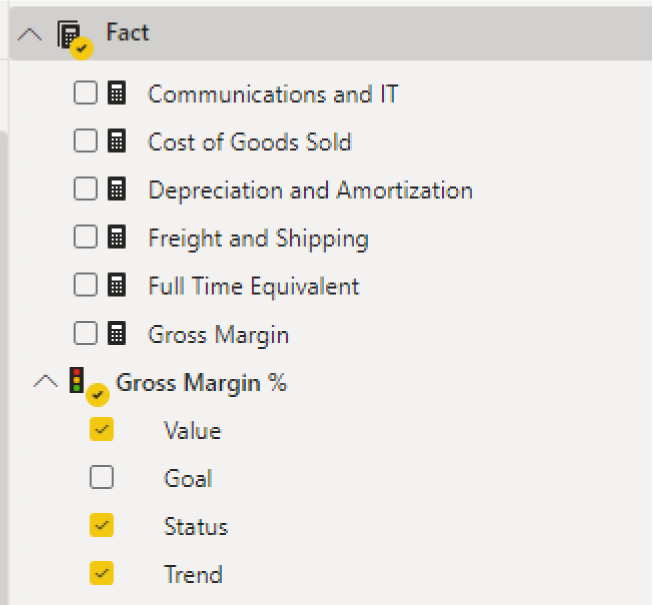
Trend Expression

```
-- DAX code for Trend Expression
VAR MarginPerc = [Gross Margin %]
VAR PrevMarginPerc = CALCULATE([Gross Margin %],SAMEPERIODLASTYEAR('Time
Period'[FullDate]))
RETURN
IF (
   NOT ISBLANK ( MarginPerc ) && NOT ISBLANK ( PrevMarginPerc ),
   SWITCH (
         TRUE,
         MarginPerc > PrevMarginPerc, 1,  -- Positive
         MarginPerc < PrevMarginPerc, -1, -- Negative
         0
        )
   )
```

Trend Graphic

We specified this as **Standard Arrow**.

7. In the Power BI user interface, the measure is shown as a KPI, no longer as a measure.



Fact

☐ Communications and IT

☐ Cost of Goods Sold

☐ Depreciation and Amortization

☐ Freight and Shipping

☐ Full Time Equivalent

☐ Gross Margin

Gross Margin %

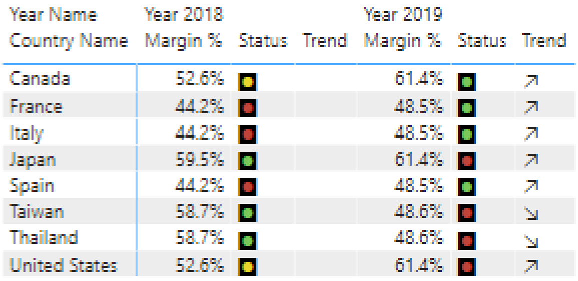
☑ Value

☐ Goal

☑ Status

☑ Trend

8. The result of our KPI in a matrix visualization is as follows:



Year Name	Year 2018			Year 2019		
Country Name	Margin %	Status	Trend	Margin %	Status	Trend
Canada	52.6%	🟨		61.4%	🟩	↗
France	44.2%	🟥		48.5%	🟩	↗
Italy	44.2%	🟥		48.5%	🟩	↗
Japan	59.5%	🟩		61.4%	🟥	↗
Spain	44.2%	🟥		48.5%	🟩	↗
Taiwan	58.7%	🟩		48.6%	🟥	↘
Thailand	58.7%	🟩		48.6%	🟥	↘
United States	52.6%	🟨		61.4%	🟥	↗

You can refer to the following articles to understand more about KPIs and their usage in Power BI and excel:

- https://www.sqlbi.com/articles/creating-kpis-in-power-bi-desktop/
- https://www.sqlbi.com/articles/kpi-in-tabular-models-for-power-bi-and-excel/

Keyword resolution

While defining DAX expressions you may use keywords. However, please be aware about the keyword resolution in Power Pivot models.

- Keywords are resolved in the scope of the Power Pivot model
- Keywords are resolved in the Description of ANY table dedicated to the Power Pivot model
- Keywords are resolved in the DAX expressions of any calculation group
- Keywords are resolved for any Additional table in the following fields:
    - AS Definition (the DAX measure)
    - KPI Status Expression
    - KPI Target Expression
    - KPI Trend Expression
- Keyword Collection that is appended to the Information table is resolved.

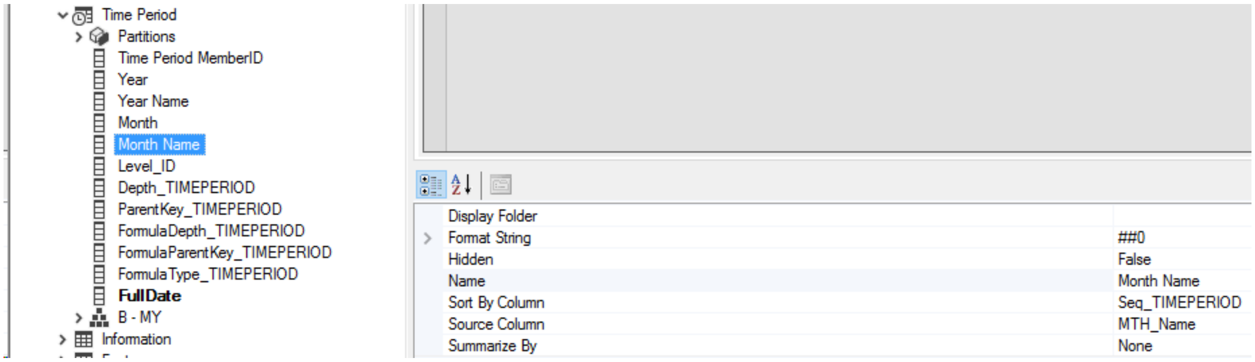
Essentially, all tables which are dedicated to Power Pivot model do NOT have keyword resolution. For clarity, define what you need explicitly in the model.

Tables which are shared among multiple Power Pivot models will have the keywords in critical expressions resolved.

AS Sort By Column

The exported relational tables of the **Power Pivot** model have a sequence column with values set according to their display order in the hierarchy in CONTROL®. The sort order for the columns is based on this column.

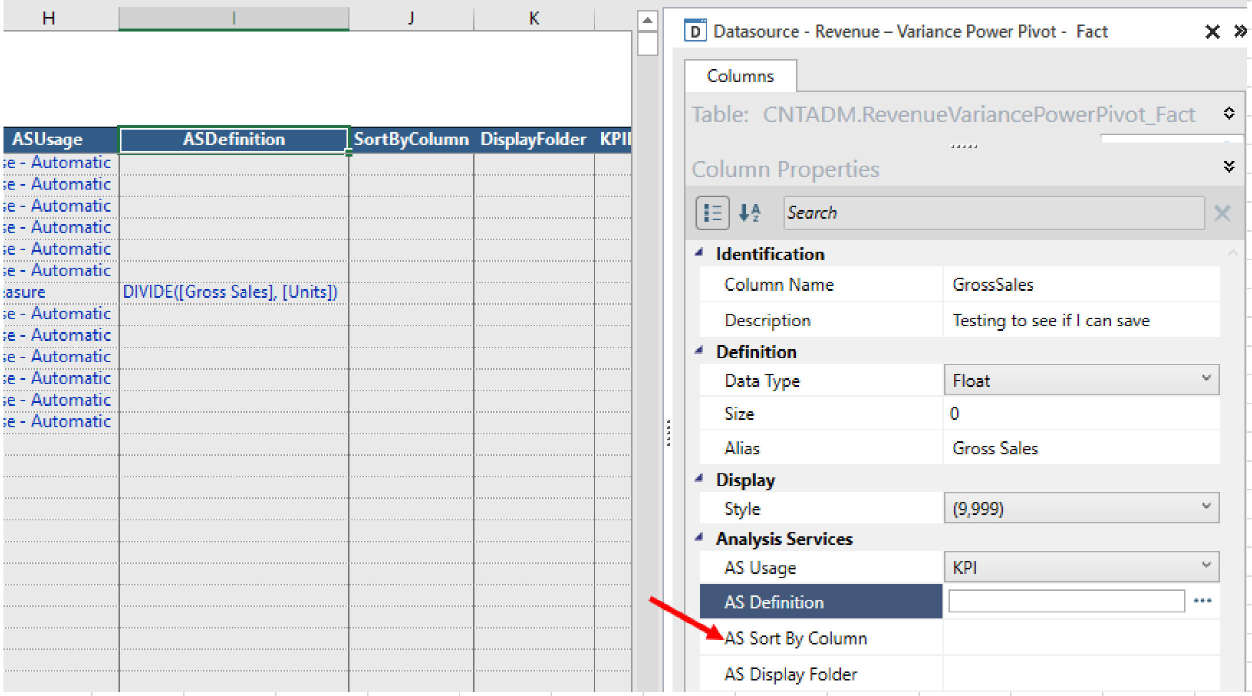
As you can see in the screenshot below the Month name column is sorted based on the sequence column Seq_TIMEPERIOD. This ensures that when you use the Month Name column in your Power BI visualizations the months are sorted based on the Month ID and not alphabetically based on the Month name.



- Time Period
  - Partitions
  - Time Period MemberID
  - Year
  - Year Name
  - Month
  - Month Name
  - Level_ID
  - Depth_TIMEPERIOD
  - ParentKey_TIMEPERIOD
  - FormulaDepth_TIMEPERIOD
  - FormulaParentKey_TIMEPERIOD
  - FormulaType_TIMEPERIOD
  - **FullDate**
  - B - MY
- Information

Display Folder	
Format String	##0
Hidden	False
Name	Month Name
Sort By Column	Seq_TIMEPERIOD
Source Column	MTH_Name
Summarize By	None

This mechanism works just fine for most cases. However, in the rare case where you want to have sorting based on a different column; like for example when you want to sort Products based on their color instead their name; you can use the **AS Sort By Column** property to specify the column that drives the sorting. When specifying this property, you need to use the **Alias** and not the **Column ID** field of your dimensions' relational table.



H	I	J	K	
ASUsage	ASDefinition	SortByColumn	DisplayFolder	KPI
e - Automatic				
e - Automatic				
e - Automatic				
e - Automatic				
e - Automatic				
e - Automatic				
asure	DIVIDE([Gross Sales], [Units])			
e - Automatic				
e - Automatic				
e - Automatic				
e - Automatic				
e - Automatic				
e - Automatic				

Datasource - Revenue – Variance Power Pivot - Fact

Columns

Table: CNTADM.RevenueVariancePowerPivot_Fact

Column Properties

Search

Identification	
Column Name	GrossSales
Description	Testing to see if I can save
Definition	
Data Type	Float
Size	0
Alias	Gross Sales
Display	
Style	(9,999)
Analysis Services	
AS Usage	KPI
AS Definition	
AS Sort By Column	
AS Display Folder	

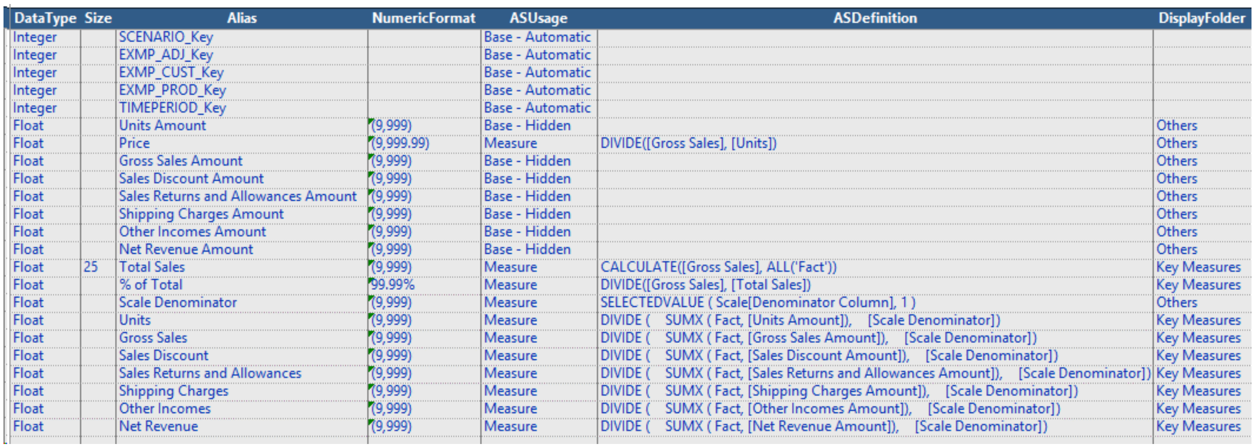
AS Display Folder

The measures of your exported **Power Pivot** model show up in the Fact table in the Power BI Fields pane. In case you have dozens of measures in the same table you might look for a better organization of the list of measures. The **AS Display Folder** column property allows grouping of measures in a folder structure.

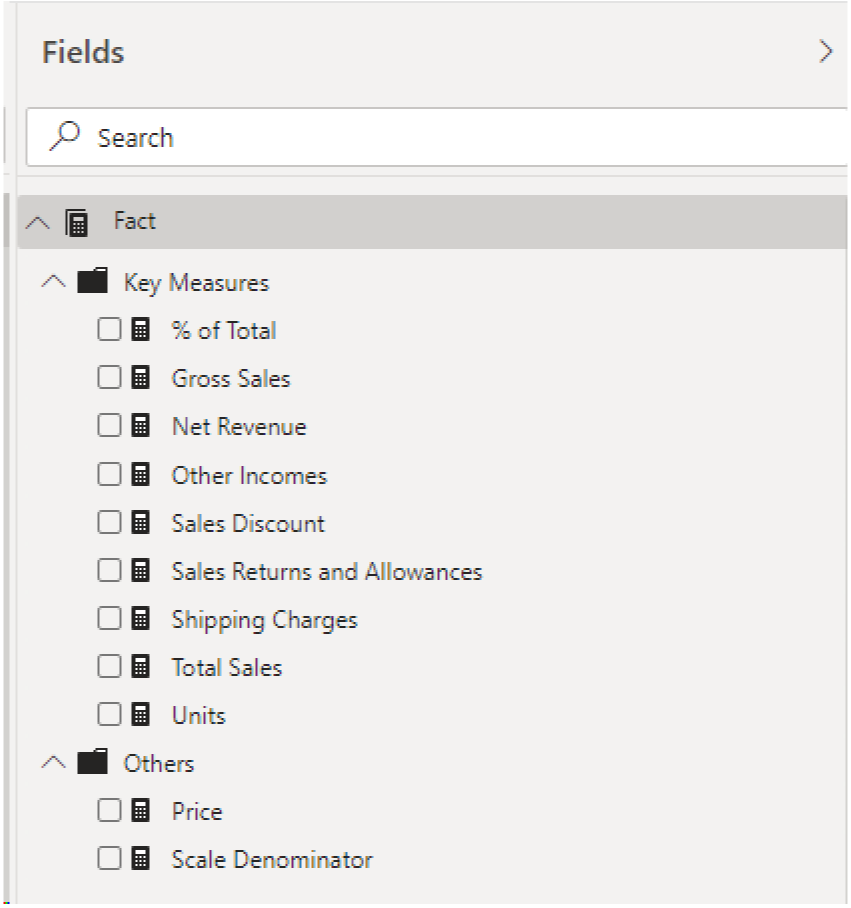
- To put a measure in a folder, specify the folder name – e.g. My Folder
- To put a measure in a nested folder grouping, use the backslash (\) character – e.g. My Folder\My Sub Folder
- To put a measure in multiple folders, use the semi-colon to separate the folder names – e.g. My Folder;My Other Folder

Note: The **DisplayFolders** are supported in Analysis Service 2016 and later versions

The screenshots below demonstrate how the **AS Display Folder** definition works in Power BI:

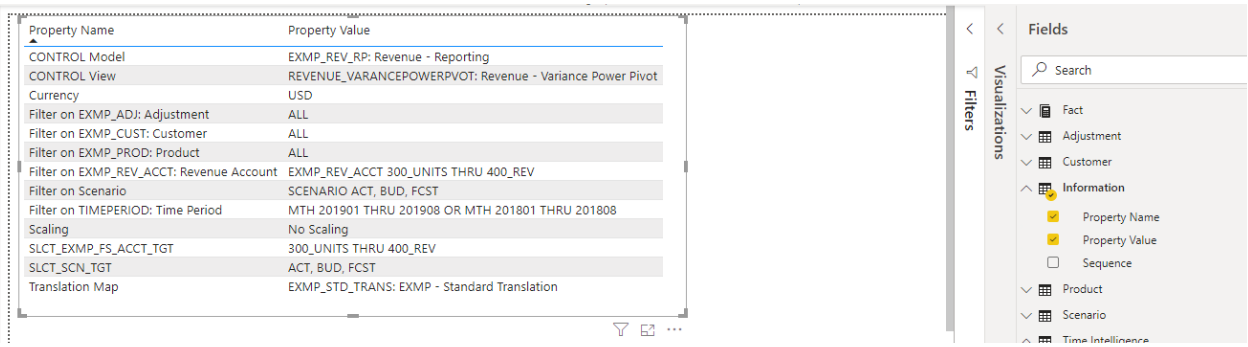


DataType	Size	Alias	NumericFormat	ASUsage	ASDefinition	DisplayFolder
Integer		SCENARIO_Key		Base - Automatic		
Integer		EXMP_ADJ_Key		Base - Automatic		
Integer		EXMP_CUST_Key		Base - Automatic		
Integer		EXMP_PROD_Key		Base - Automatic		
Integer		TIMEPERIOD_Key		Base - Automatic		
Float		Units Amount	(9,999)	Base - Hidden		Others
Float		Price	(9,999.99)	Measure	DIVIDE([Gross Sales], [Units])	Others
Float		Gross Sales Amount	(9,999)	Base - Hidden		Others
Float		Sales Discount Amount	(9,999)	Base - Hidden		Others
Float		Sales Returns and Allowances Amount	(9,999)	Base - Hidden		Others
Float		Shipping Charges Amount	(9,999)	Base - Hidden		Others
Float		Other Incomes Amount	(9,999)	Base - Hidden		Others
Float		Net Revenue Amount	(9,999)	Base - Hidden		Others
Float	25	Total Sales	(9,999)	Measure	CALCULATE([Gross Sales], ALL('Fact'))	Key Measures
Float		% of Total	99.99%	Measure	DIVIDE([Gross Sales], [Total Sales])	Key Measures
Float		Scale Denominator	(9,999)	Measure	SELECTEDVALUE (Scale[Denominator Column], 1)	Others
Float		Units	(9,999)	Measure	DIVIDE (SUMX (Fact, [Units Amount]), [Scale Denominator])	Key Measures
Float		Gross Sales	(9,999)	Measure	DIVIDE (SUMX (Fact, [Gross Sales Amount]), [Scale Denominator])	Key Measures
Float		Sales Discount	(9,999)	Measure	DIVIDE (SUMX (Fact, [Sales Discount Amount]), [Scale Denominator])	Key Measures
Float		Sales Returns and Allowances	(9,999)	Measure	DIVIDE (SUMX (Fact, [Sales Returns and Allowances Amount]), [Scale Denominator])	Key Measures
Float		Shipping Charges	(9,999)	Measure	DIVIDE (SUMX (Fact, [Shipping Charges Amount]), [Scale Denominator])	Key Measures
Float		Other Incomes	(9,999)	Measure	DIVIDE (SUMX (Fact, [Other Incomes Amount]), [Scale Denominator])	Key Measures
Float		Net Revenue	(9,999)	Measure	DIVIDE (SUMX (Fact, [Net Revenue Amount]), [Scale Denominator])	Key Measures



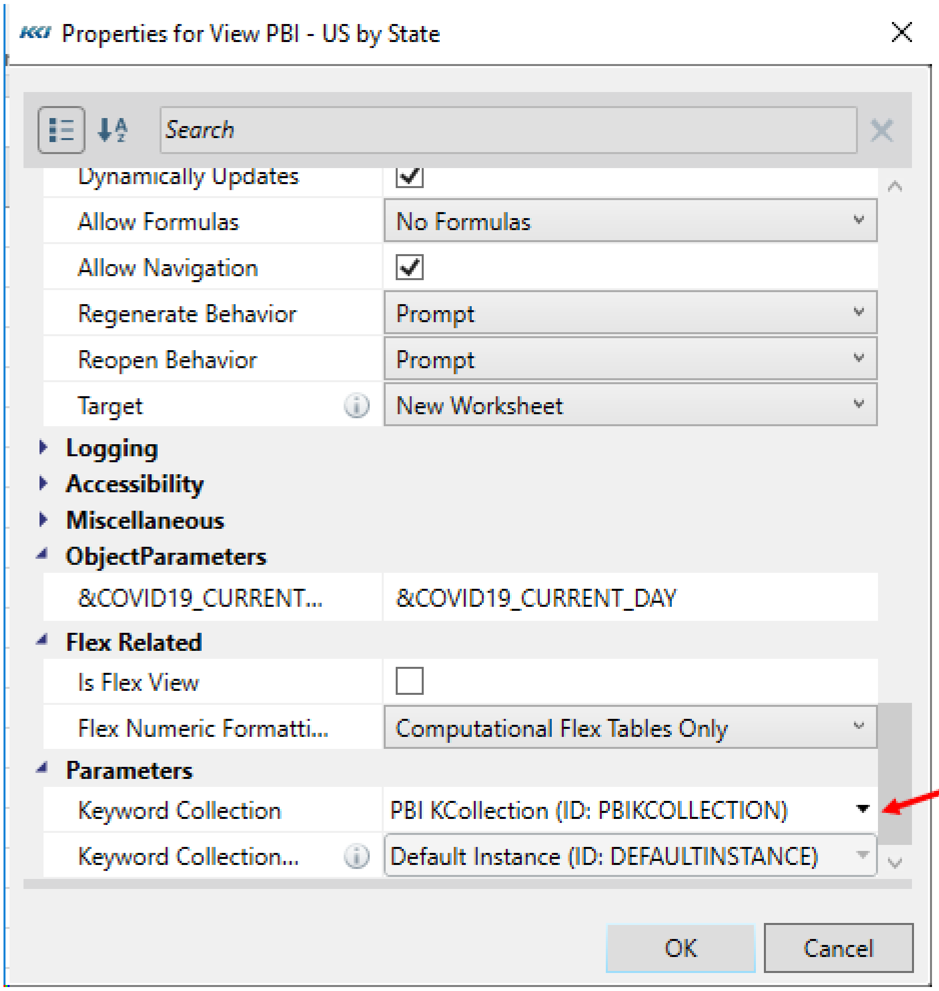
Information Table

Every export creates a table that summarizes critical information about the CONTROL® source data to assist in validation and troubleshooting. The information includes but not limited to, CONTROL® model(s) and view(s) names, view properties like scaling, currency, filters on dimensions and the Keyword collection used by the base view. You can see the information in Power BI using the Property name and value columns from the **Information** table.



Property Name	Property Value
CONTROL Model	EXMP_REV_RP: Revenue - Reporting
CONTROL View	REVENUE_VARANCEPOWERPVOT: Revenue - Variance Power Pivot
Currency	USD
Filter on EXMP_ADJ: Adjustment	ALL
Filter on EXMP_CUST: Customer	ALL
Filter on EXMP_PROD: Product	ALL
Filter on EXMP_REV_ACCT: Revenue Account	EXMP_REV_ACCT 300_UNITS THRU 400_REV
Filter on Scenario	SCENARIO ACT, BUD, FCST
Filter on TIMEPERIOD: Time Period	MTH 201901 THRU 201908 OR MTH 201801 THRU 201808
Scaling	No Scaling
SLCT_EXMP_FS_ACCT_TGT	300_UNITS THRU 400_REV
SLCT_SCN_TGT	ACT, BUD, FCST
Translation Map	EXMP_STD_TRANS: EXMP - Standard Translation

Here’s an example of a keyword collection is included in the base view of a **Power Pivot** model.



Properties for View PBI - US by State

Dynamically Updates	☑
Allow Formulas	No Formulas
Allow Navigation	☑
Regenerate Behavior	Prompt
Reopen Behavior	Prompt
Target	New Worksheet

Logging

Accessibility

Miscellaneous

ObjectParameters

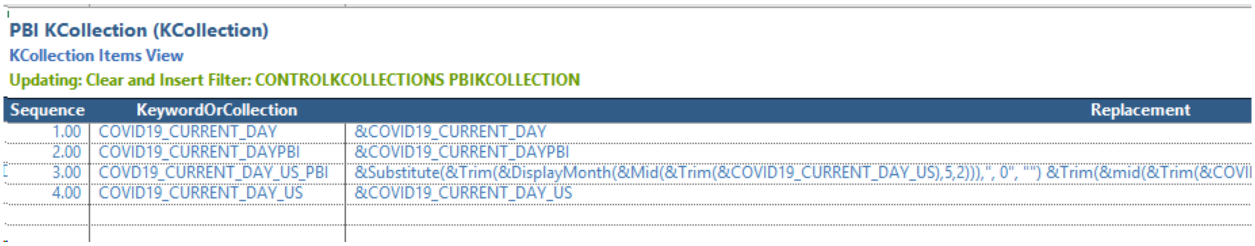
&COVID19_CURRENT...	&COVID19_CURRENT_DAY

Flex Related

Is Flex View	□
Flex Numeric Formatti...	Computational Flex Tables Only

Parameters

Keyword Collection	PBI KCollection (ID: PBIKCOLLECTION)
Keyword Collection...	Default Instance (ID: DEFAULTINSTANCE)



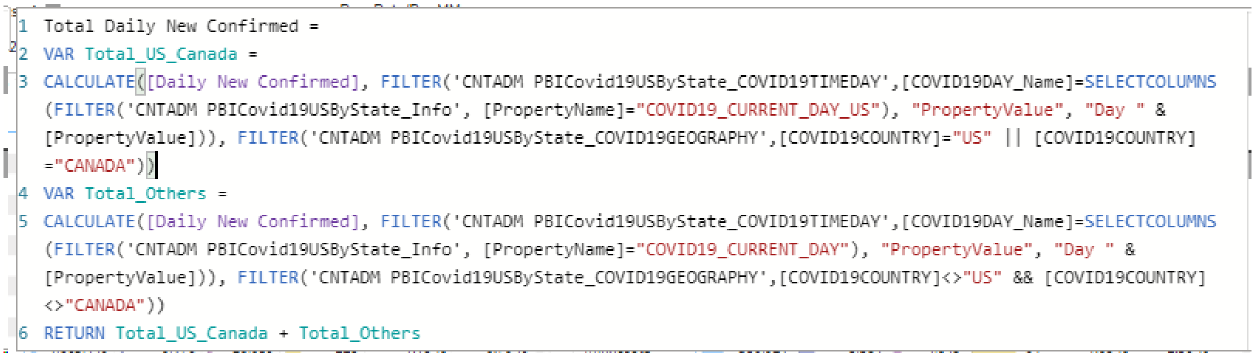
PBI KCollection (KCollection)

KCollection Items View

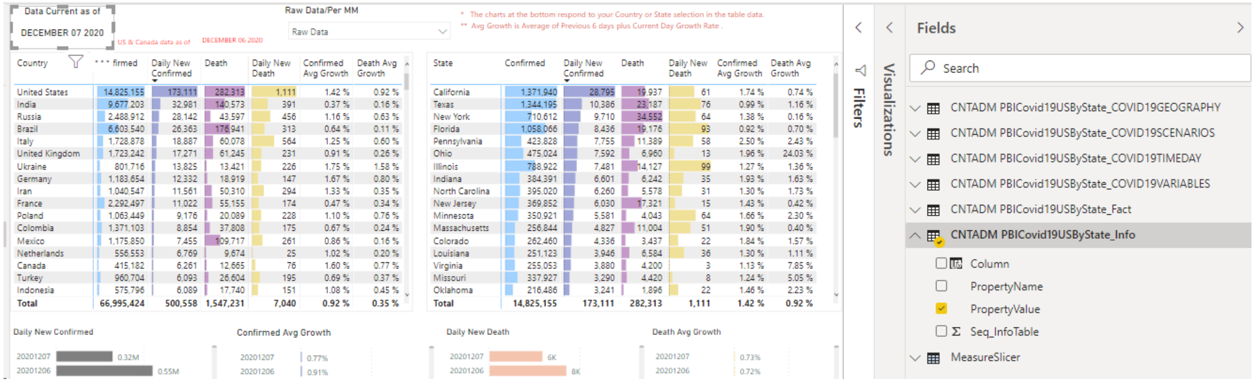
Updating: Clear and Insert Filter: CONTROLKCOLLECTIONS PBIKCOLLECTION

Sequence	KeywordOrCollection	Replacement
1.00	COVID19_CURRENT_DAY	&COVID19_CURRENT_DAY
2.00	COVID19_CURRENT_DAYPBI	&COVID19_CURRENT_DAYPBI
3.00	COVD19_CURRENT_DAY_US_PBI	&Substitute(&Trim(&DisplayMonth(&Mid(&Trim(&COVID19_CURRENT_DAY_US),5,2))),", 0", "") &Trim(&mid(&Trim(&COVII
4.00	COVID19_CURRENT_DAY_US	&COVID19_CURRENT_DAY_US

The values of the keywords from the keyword collection are used in the DAX expressions to compute values as well as for display in the Power BI report.



```
Total Daily New Confirmed =
VAR Total_US_Canada =
CALCULATE([Daily New Confirmed], FILTER('CNTADM PBICovid19USByState_COVID19TIMEDAY',[COVID19DAY_Name]=SELECTCOLUMNS
(FILTER('CNTADM PBICovid19USByState_Info', [PropertyName]="COVID19_CURRENT_DAY_US"), "PropertyValue", "Day " &
[PropertyValue])), FILTER('CNTADM PBICovid19USByState_COVID19GEOGRAPHY',[COVID19COUNTRY]="US" || [COVID19COUNTRY]
="CANADA"))
VAR Total_Others =
CALCULATE([Daily New Confirmed], FILTER('CNTADM PBICovid19USByState_COVID19TIMEDAY',[COVID19DAY_Name]=SELECTCOLUMNS
(FILTER('CNTADM PBICovid19USByState_Info', [PropertyName]="COVID19_CURRENT_DAY"), "PropertyValue", "Day " &
[PropertyValue])), FILTER('CNTADM PBICovid19USByState_COVID19GEOGRAPHY',[COVID19COUNTRY]<>"US" && [COVID19COUNTRY]
<>"CANADA"))
RETURN Total_US_Canada + Total_Others
```



Data Current as of

DECEMBER 07 2020

US & Canada data as of DECEMBER 06 2020

Raw Data/Per MM

Raw Data

* The charts at the bottom respond to your Country or State selection in the table data.

** Avg Growth is Average of Previous 6 days plus Current Day Growth Rate .

Country	˙˙˙ firmed	Daily New Confirmed	Death	Daily New Death	Confirmed Avg Growth	Death Avg Growth
United States	14,825,155	173,111	282,313	1,111	1.42 %	0.92 %
India	9,677,203	32,981	140,573	391	0.37 %	0.16 %
Russia	2,488,912	28,142	43,597	456	1.16 %	0.63 %
Brazil	6,603,540	26,363	176,941	313	0.64 %	0.11 %
Italy	1,728,878	18,887	60,078	564	1.25 %	0.60 %
United Kingdom	1,723,242	17,271	61,245	231	0.91 %	0.26 %
Ukraine	801,716	13,825	13,421	226	1.75 %	1.58 %
Germany	1,183,654	12,332	18,919	147	1.67 %	0.80 %
Iran	1,040,547	11,561	50,310	294	1.33 %	0.35 %
France	2,292,497	11,022	55,155	174	0.47 %	0.34 %
Poland	1,063,449	9,176	20,089	228	1.10 %	0.76 %
Colombia	1,371,103	8,854	37,808	175	0.67 %	0.24 %
Mexico	1,175,850	7,455	109,717	261	0.86 %	0.16 %
Netherlands	556,553	6,769	9,674	25	1.02 %	0.20 %
Canada	415,182	6,261	12,665	76	1.60 %	0.77 %
Turkey	960,704	6,093	26,604	195	0.69 %	0.37 %
Indonesia	575,796	6,089	17,740	151	1.08 %	0.45 %
Total	**66,995,424**	**500,558**	**1,547,231**	**7,040**	**0.92 %**	**0.35 %**

State	Confirmed	Daily New Confirmed	Death	Daily New Death	Confirmed Avg Growth	Death Avg Growth
California	1,371,940	28,795	19,937	61	1.74 %	0.74 %
Texas	1,344,195	10,386	23,187	76	0.99 %	1.16 %
New York	710,612	9,710	34,552	64	1.38 %	0.16 %
Florida	1,058,066	8,436	19,176	93	0.92 %	0.70 %
Pennsylvania	423,828	7,755	11,389	58	2.50 %	2.43 %
Ohio	475,024	7,592	6,960	13	1.96 %	24.03 %
Illinois	788,922	7,481	14,127	99	1.27 %	1.36 %
Indiana	384,391	6,601	6,242	35	1.93 %	1.63 %
North Carolina	395,020	6,260	5,578	31	1.30 %	1.73 %
New Jersey	369,852	6,030	17,321	15	1.43 %	0.42 %
Minnesota	350,921	5,581	4,043	64	1.66 %	2.30 %
Massachusetts	256,844	4,827	11,004	51	1.90 %	0.40 %
Colorado	262,460	4,336	3,437	22	1.84 %	1.57 %
Louisiana	251,123	3,946	6,584	36	1.30 %	1.11 %
Virginia	255,053	3,880	4,200	3	1.13 %	7.85 %
Missouri	337,927	3,290	4,420	8	1.24 %	5.05 %
Oklahoma	216,486	3,241	1,896	22	1.46 %	2.23 %
Total	**14,825,155**	**173,111**	**282,313**	**1,111**	**1.42 %**	**0.92 %**

Tabular Editor

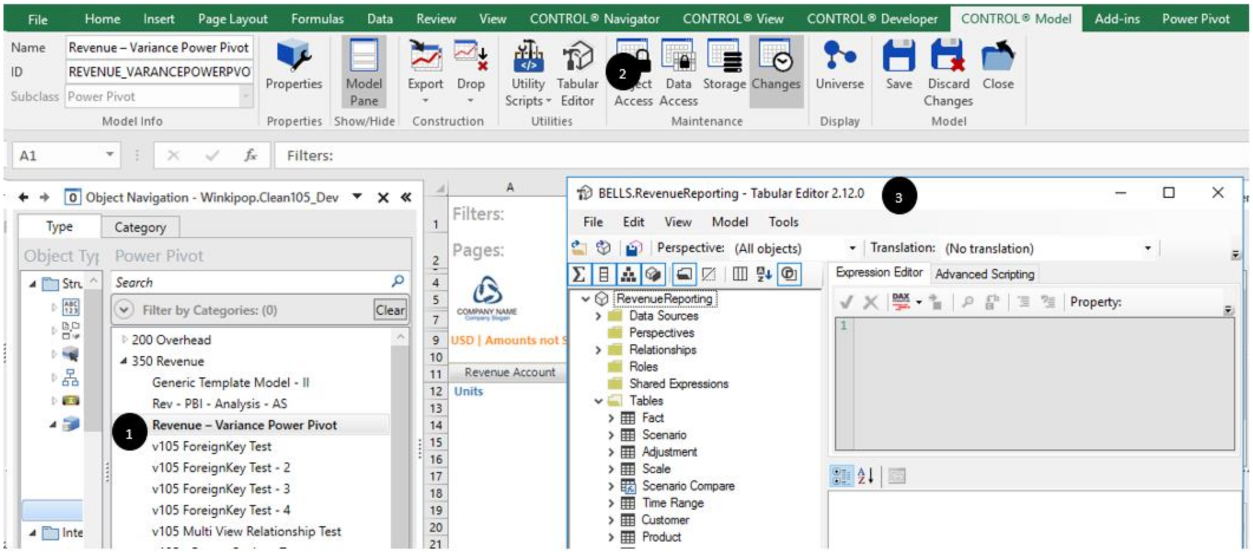
For **Power Pivot** models that have their **Analysis Services usage** property set to **Exported to AS – Power BI and Excel (Recommended)**; CONTROL® creates a AS Tabular model in Analysis Services as part of the Export process. The **CONTROL-Managed** datasources dedicated to the **Power Pivot** model allow us to review and override properties related to the AS Tabular model.

However, during development of Power Pivot models we may want to review the exported Tabular model. **Tabular Editor** is a tool that enables you to easily build, maintain and manage tabular models using an intuitive, lightweight editor. A hierarchical view shows all objects in your tabular model. They are organized by display folders, with support for multi-select property editing and DAX syntax highlighting.

For **Power Pivot** models exported from CONTROL® Tabular editor is very useful for:

1) Validating that the export has produced the expected result.
2) Dealing with properties that the export does not create or maintain.
3) Leveraging the DAX expression editor with syntax coloring and error detection.

On the **CONTROL® Model** tab, in the **Utilities** group, select **Tabular Editor**. The Tabular Editor is launched and opens the Tabular Model associated to your Power Pivot model.



Easy Power BI

To encourage our user community to create Power BI reports on their own using their CONTROL® data we have introduced a simple mechanism called Easy Power BI.

The framework of Easy Power BI is described below:

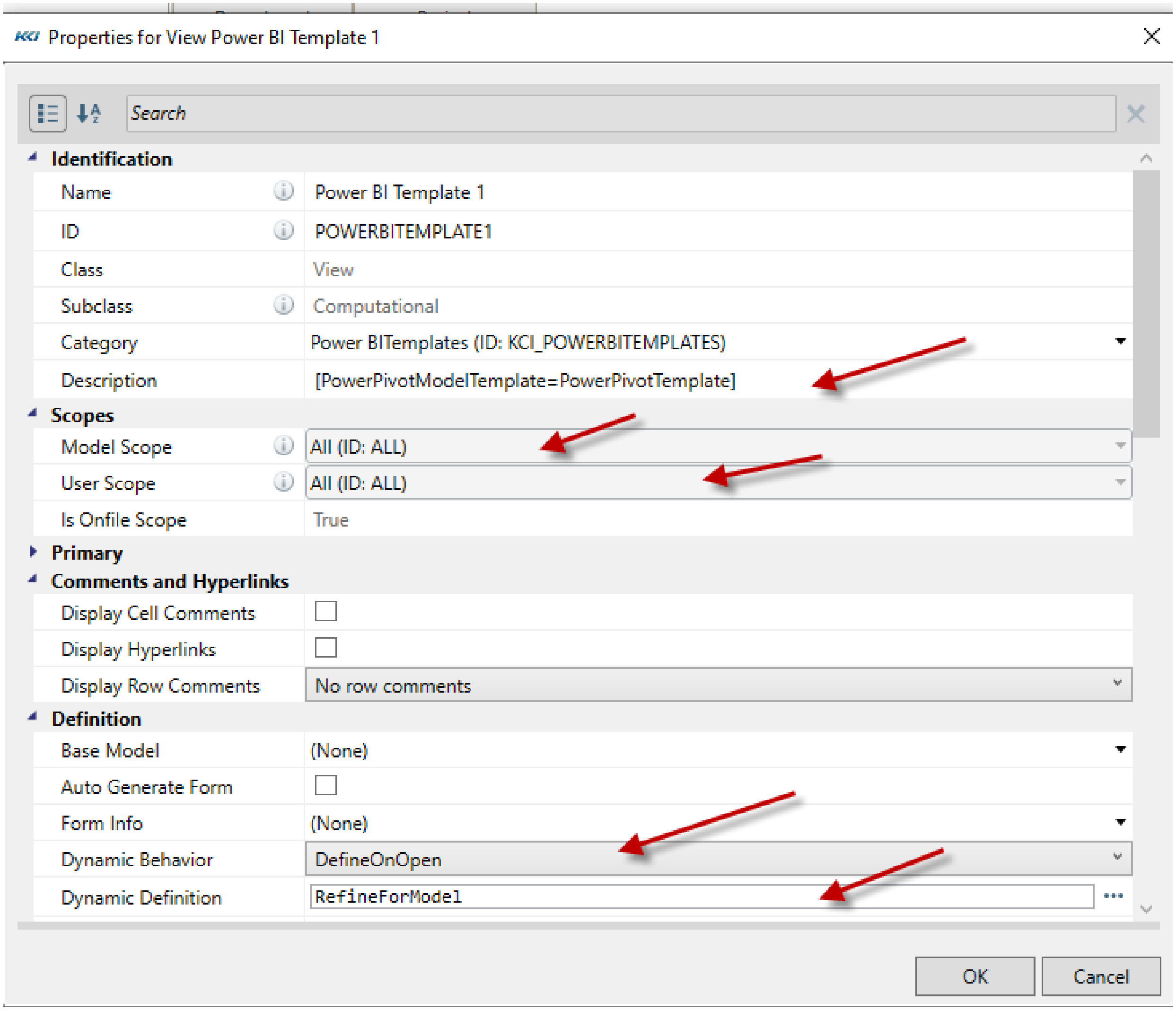
Categories

- Create an **Access category** with the ID **KCI_PowerBITemplates**. Ideally, this is where you should save your CONTROL® view templates.
- Create an **Access category** with the ID **KCI_PowerBIViews**. This is the default category for saving the user-customized views that will be exported.
- Both these categories are optional but are recommended for organizing the exported data. If you do not want to use the specified ID's, create a keyword (e.g., &KCI_PowerBITemplates) whose replacement is the category ID. The keyword may have user scopes but should not have model scopes.

Power BI Template Model and Views

- Create one or more template **Power Pivot** Models. The following three template models are made available as part of the upgrade to version 10.5:
    - PBI Model Template - Unanchored
    - PBI Model Template - Custom Dimension
    - PBI Model Template - Calculation Group

- Create one or more views to serve as templates and put them in the **KCI_PowerBITemplates** category.
- Each of these template views must contain as part of its Description [PowerPivotModelTemplate=xxxxx], where xxxxx is the ID of a power pivot model.



Properties for View Power BI Template 1

Identification

Name	Power BI Template 1
ID	POWERBITEMPLATE1
Class	View
Subclass	Computational
Category	Power BITemplates (ID: KCI_POWERBITEMPLATES)
Description	[PowerPivotModelTemplate=PowerPivotTemplate]

Scopes

Model Scope	All (ID: ALL)
User Scope	All (ID: ALL)
Is Onfile Scope	True

Primary

Comments and Hyperlinks

Display Cell Comments	□
Display Hyperlinks	□
Display Row Comments	No row comments

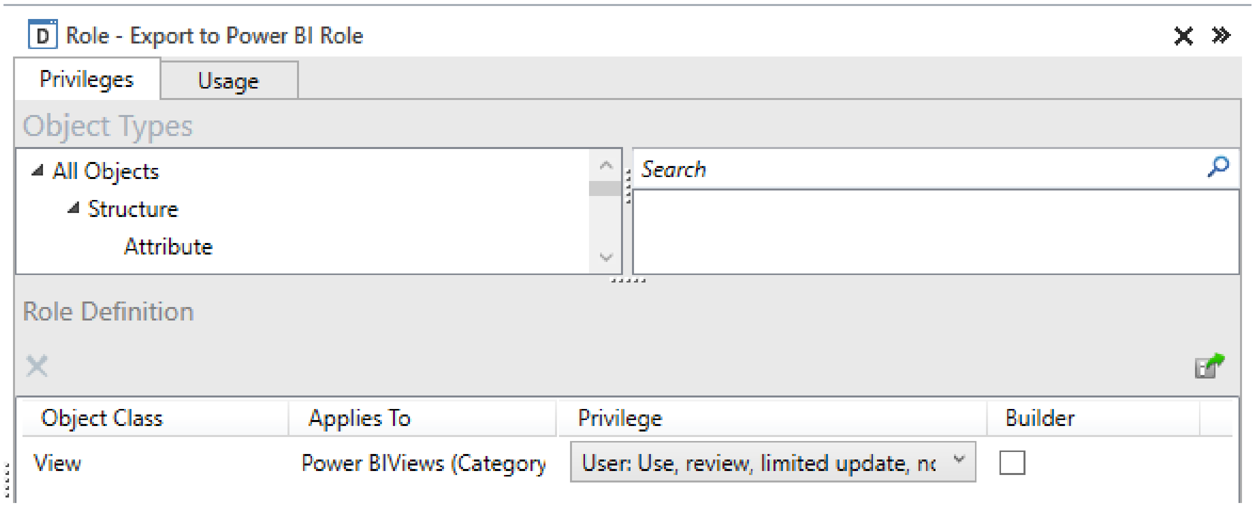
Definition

Base Model	(None)
Auto Generate Form	□
Form Info	(None)
Dynamic Behavior	DefineOnOpen
Dynamic Definition	RefineForModel

- The following three template views are made available as part of the upgrade to version 10.5:
  - PBI View Template – Unanchored
  - PBI View Template – Custom Dimension
  - PBI View Template – Calculation Group

Export Script

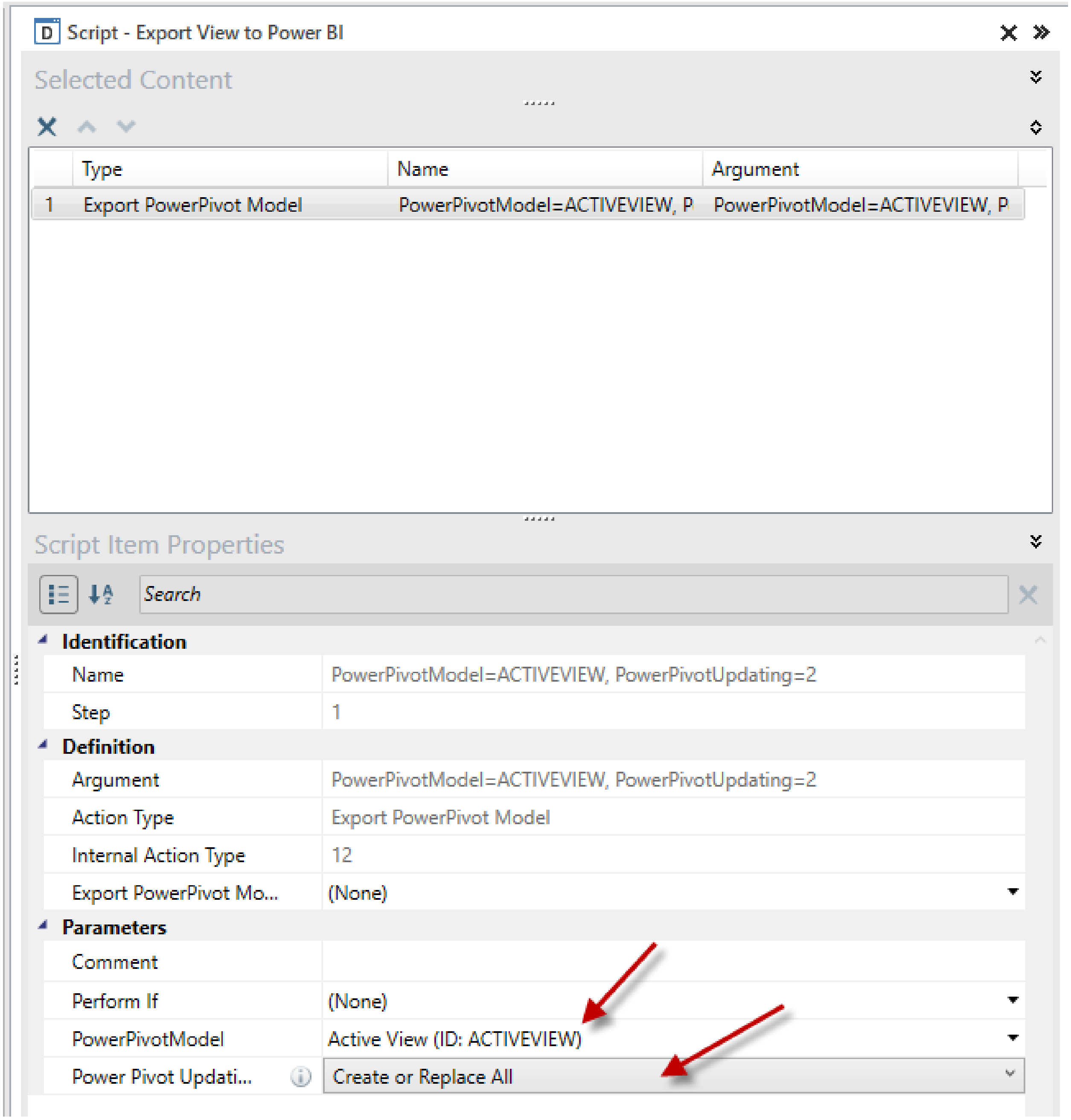
1. Create a role that specifies views in the **KCI_PowerBIViews category**



Role - Export to Power BI Role

Object Class	Applies To	Privilege	Builder
View	Power BIViews (Category	User: Use, review, limited update, nc	□

- Create a script which performs the export. The action script uses the option of PowerPivotModel=**ActiveView**, and the Power Pivot UpdatingBehavior=**Create or Replace All**



Script - Export View to Power BI

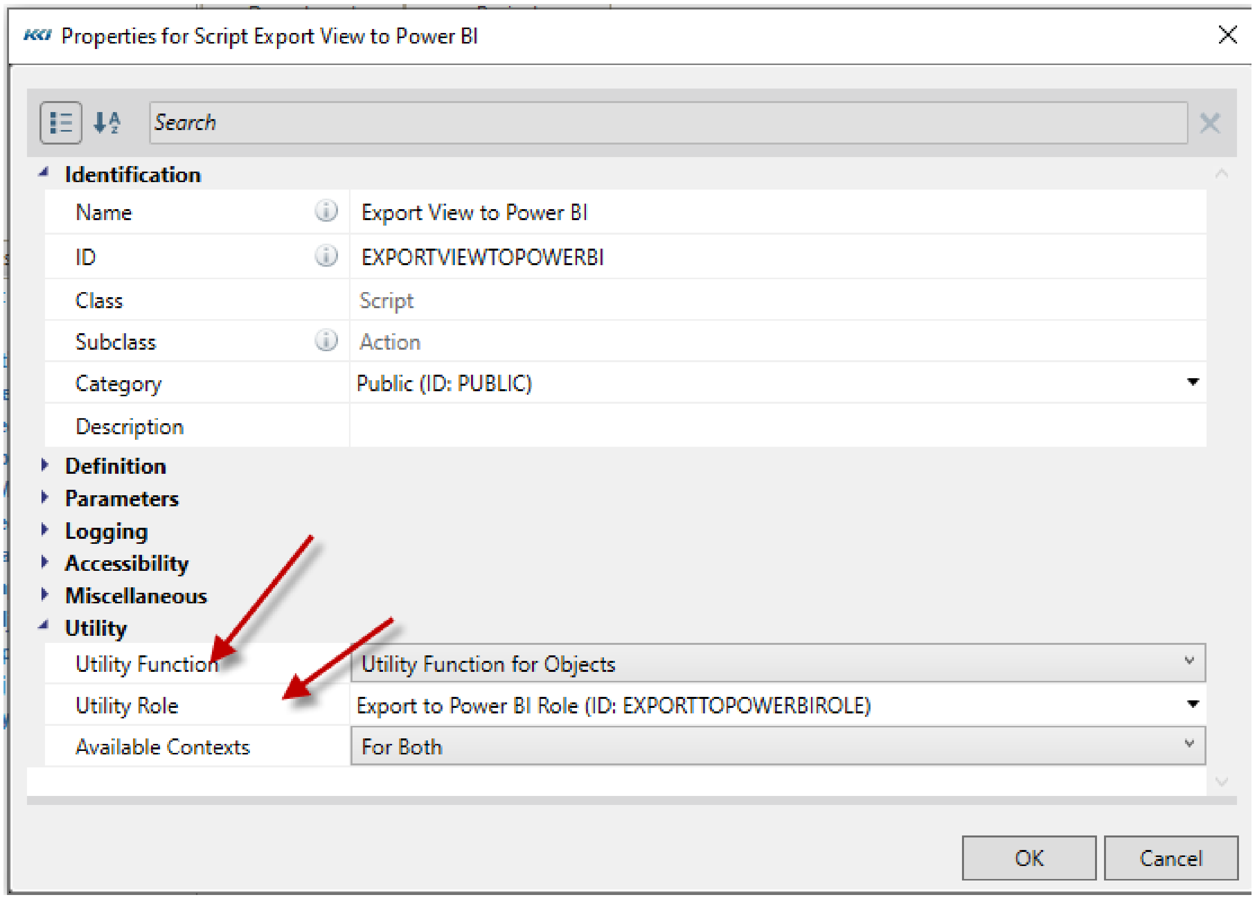
Selected Content

	Type	Name	Argument
1	Export PowerPivot Model	PowerPivotModel=ACTIVEVIEW, P	PowerPivotModel=ACTIVEVIEW, P

Script Item Properties

Identification	
Name	PowerPivotModel=ACTIVEVIEW, PowerPivotUpdating=2
Step	1
Definition	
Argument	PowerPivotModel=ACTIVEVIEW, PowerPivotUpdating=2
Action Type	Export PowerPivot Model
Internal Action Type	12
Export PowerPivot Mo...	(None)
Parameters	
Comment	
Perform If	(None)
PowerPivotModel	Active View (ID: ACTIVEVIEW)
Power Pivot Updati...	Create or Replace All

2. Make it a utility script that is available for use with views in the **KCI_PowerBIViews** category



Properties for Script Export View to Power BI

Identification

Name	Export View to Power BI
ID	EXPORTVIEWTOPOWERBI
Class	Script
Subclass	Action
Category	Public (ID: PUBLIC)
Description	

Definition

Parameters

Logging

Accessibility

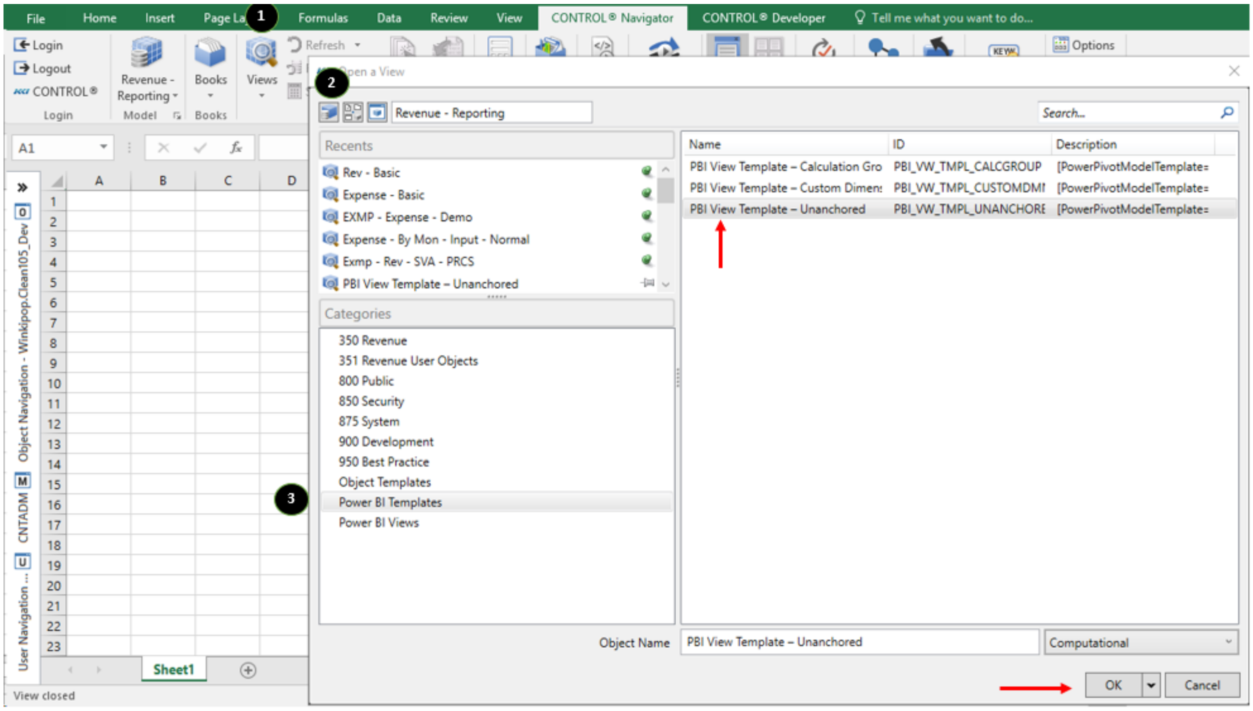
Miscellaneous

Utility

Utility Function	Utility Function for Objects
Utility Role	Export to Power BI Role (ID: EXPORTTOPOWERBIROLE)
Available Contexts	For Both

Creating Easy Power BI Views

1. From the **CONTROL® Navigator** tab, in the **Views** group, click **Views**.
2. Click the **Select Model** icon and select a *model*.
3. Click the **Power BI Templates** category, select one of the *views*, and then click **OK**.



Revenue - Reporting

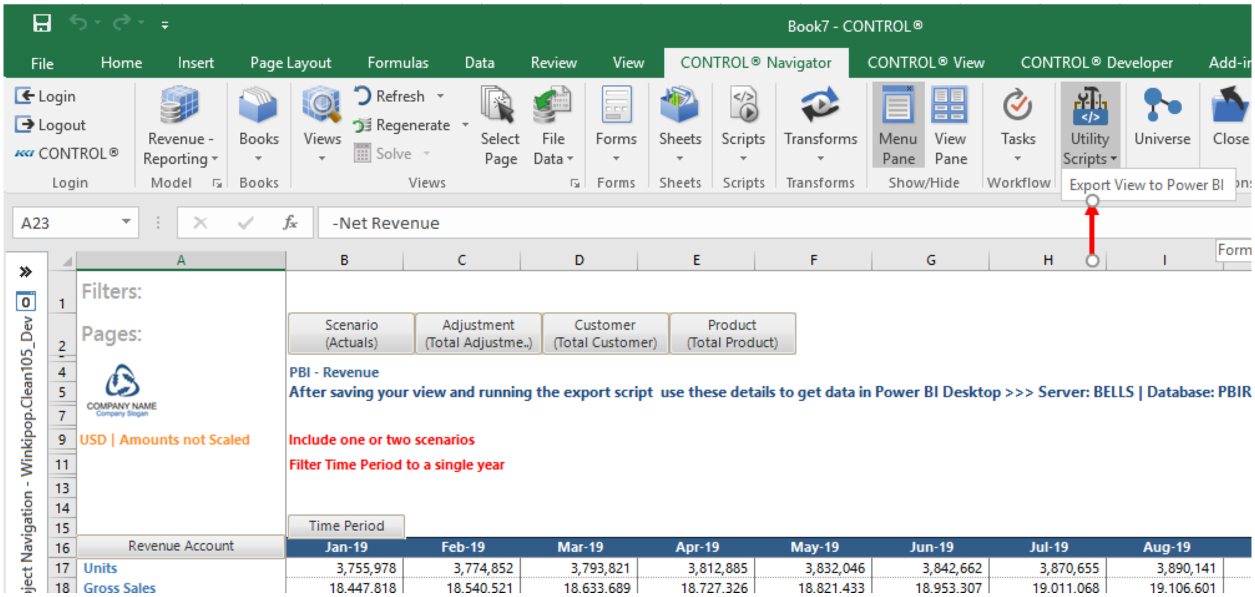
Recents
- Rev - Basic
- Expense - Basic
- EXMP - Expense - Demo
- Expense - By Mon - Input - Normal
- Exmp - Rev - SVA - PRCS
- PBI View Template – Unanchored

Categories
- 350 Revenue
- 351 Revenue User Objects
- 800 Public
- 850 Security
- 875 System
- 900 Development
- 950 Best Practice
- Object Templates
- Power BI Templates
- Power BI Views

Name	ID	Description
PBI View Template – Calculation Gro	PBI_VW_TMPL_CALCGROUP	[PowerPivotModelTemplate=
PBI View Template – Custom Dimen:	PBI_VW_TMPL_CUSTOMDMI	[PowerPivotModelTemplate=
PBI View Template – Unanchored	PBI_VW_TMPL_UNANCHORE	[PowerPivotModelTemplate=

Object Name: PBI View Template – Unanchored | Computational

4. Modify the *view's* filters and branches as needed.
5. Click **Views** > **Save As** and give it a descriptive name. The view will be saved in the **KCI_PowerBIViews** category.
6. On the **CONTROL® Navigator** ribbon, in the **Utilities** group, click the down arrow under **Utility Scripts**, and click **Export View to Power BI**.



-Net Revenue

USD | Amounts not Scaled

Scenario (Actuals) | Adjustment (Total Adjustme..) | Customer (Total Customer) | Product (Total Product)

PBI - Revenue

After saving your view and running the export script use these details to get data in Power BI Desktop >>> Server: BELLS | Database: PBIR

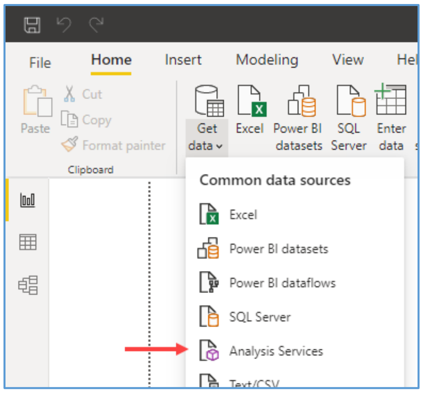
Include one or two scenarios

Filter Time Period to a single year

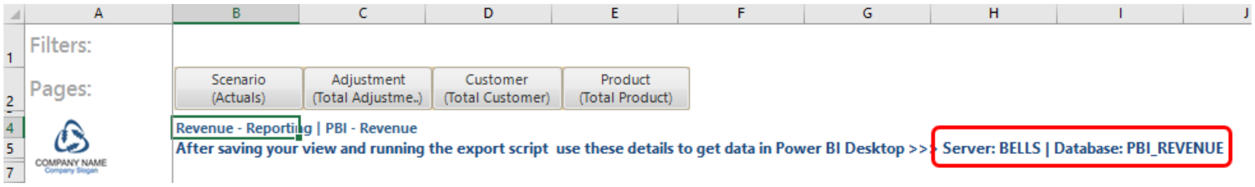
Time Period

Revenue Account	Jan-19	Feb-19	Mar-19	Apr-19	May-19	Jun-19	Jul-19	Aug-19
Units	3,755,978	3,774,852	3,793,821	3,812,885	3,832,046	3,842,662	3,870,655	3,890,141
Gross Sales	18.447.818	18.540.521	18.633.689	18.727.326	18.821.433	18.953.307	19.011.068	19.106.601

7. Launch Microsoft Power BI Desktop.
8. Once on the **Home** tab, in the **Data** group, click **Get data** > **Analysis Services**.



9. Enter your Analysis Services server name and database name. These details are available in the header section of your newly saved Easy Power BI view.



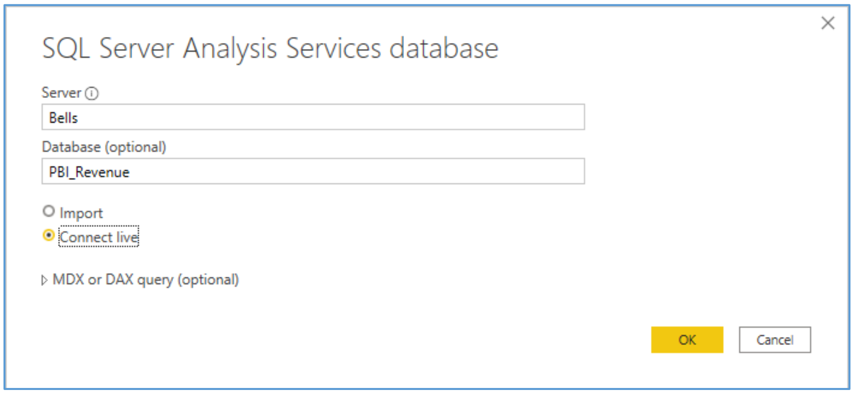
Filters:

Pages:

Scenario (Actuals)	Adjustment (Total Adjustme..)	Customer (Total Customer)	Product (Total Product)

Revenue - Reporting | PBI - Revenue

After saving your view and running the export script use these details to get data in Power BI Desktop >> Server: BELLS | Database: PBI_REVENUE



SQL Server Analysis Services database

Server ⓘ

Bells

Database (optional)

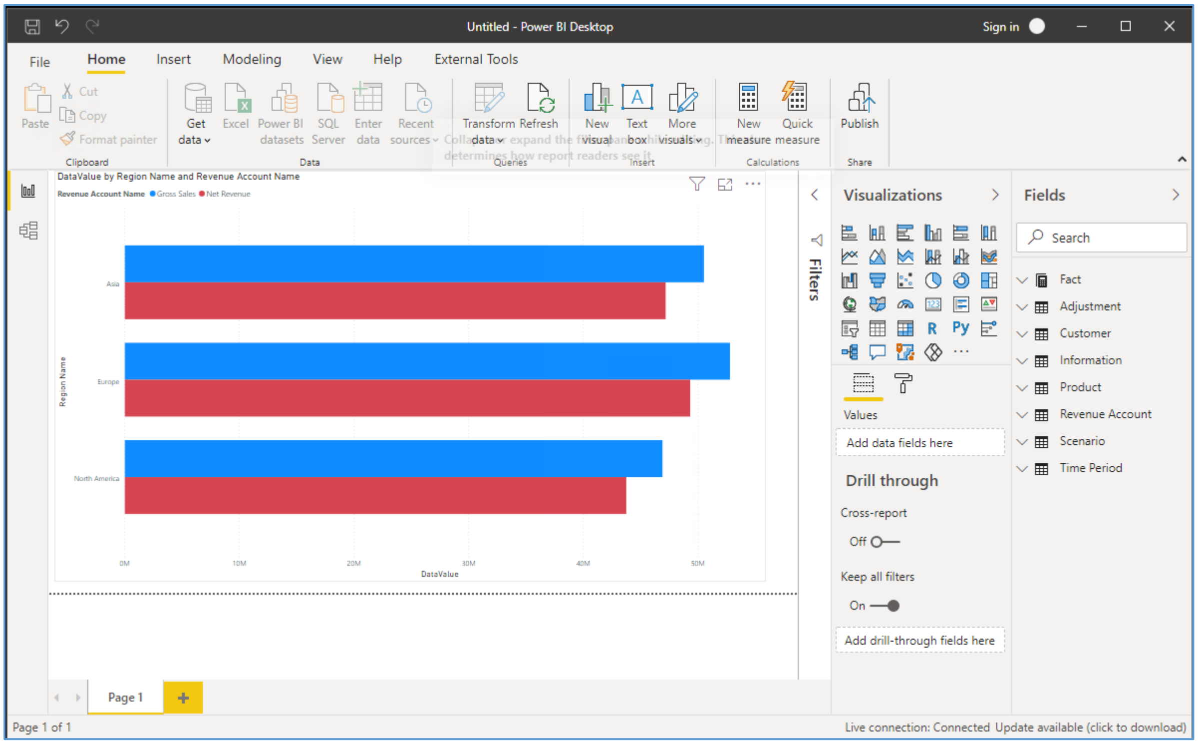
PBI_Revenue

○ Import

● Connect live

▷ MDX or DAX query (optional)

10. Start creating visualizations!



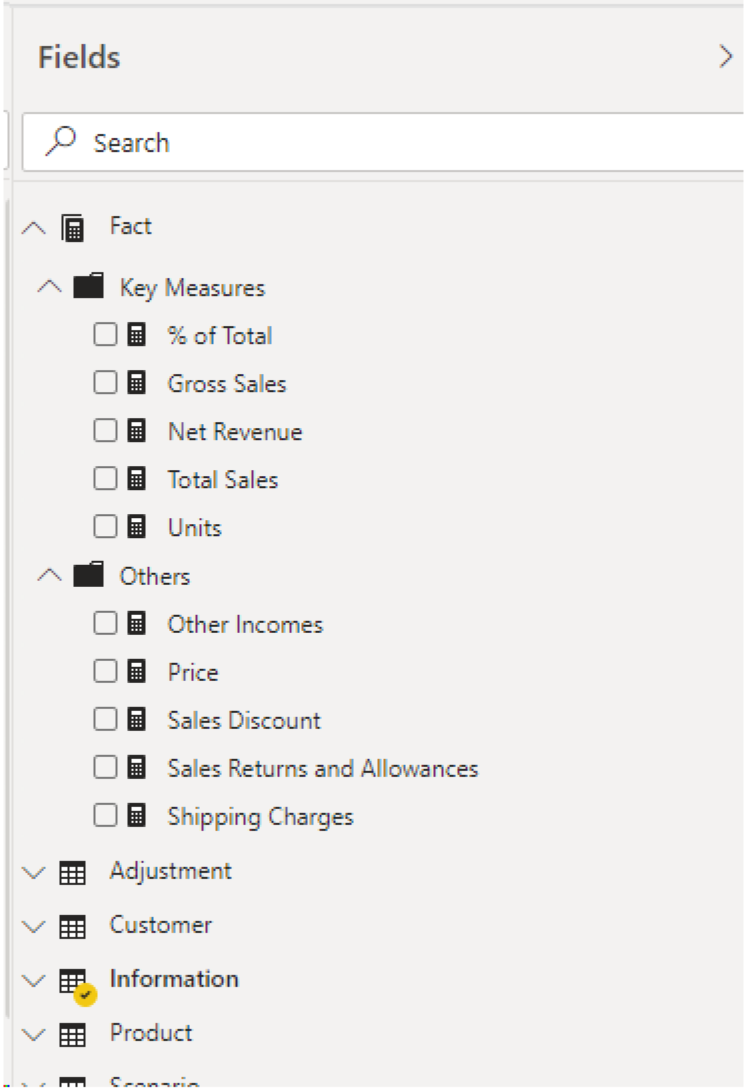
11. When your data or meta-data changes, simply reopen the *view* and re-run the script. In most cases your visualizations will just keep working.

Other administrative notes

Override AS table name

One of the primary goals of creating an AS Model from your CONTROL® data is to provide an understandable, shared source for business savvy users to easily create visualizations and derive insights.

The use of familiar, understandable names and useful tooltips is critical in supporting that goal. The use of the names and descriptions in CONTROL® is a good first step, but we recommend that you review the model in Power BI and fine tune it to suit your target audience.



Fields

Search

Fact

Key Measures

- □ % of Total
- □ Gross Sales
- □ Net Revenue
- □ Total Sales
- □ Units

Others

- □ Other Incomes
- □ Price
- □ Sales Discount
- □ Sales Returns and Allowances
- □ Shipping Charges

Adjustment

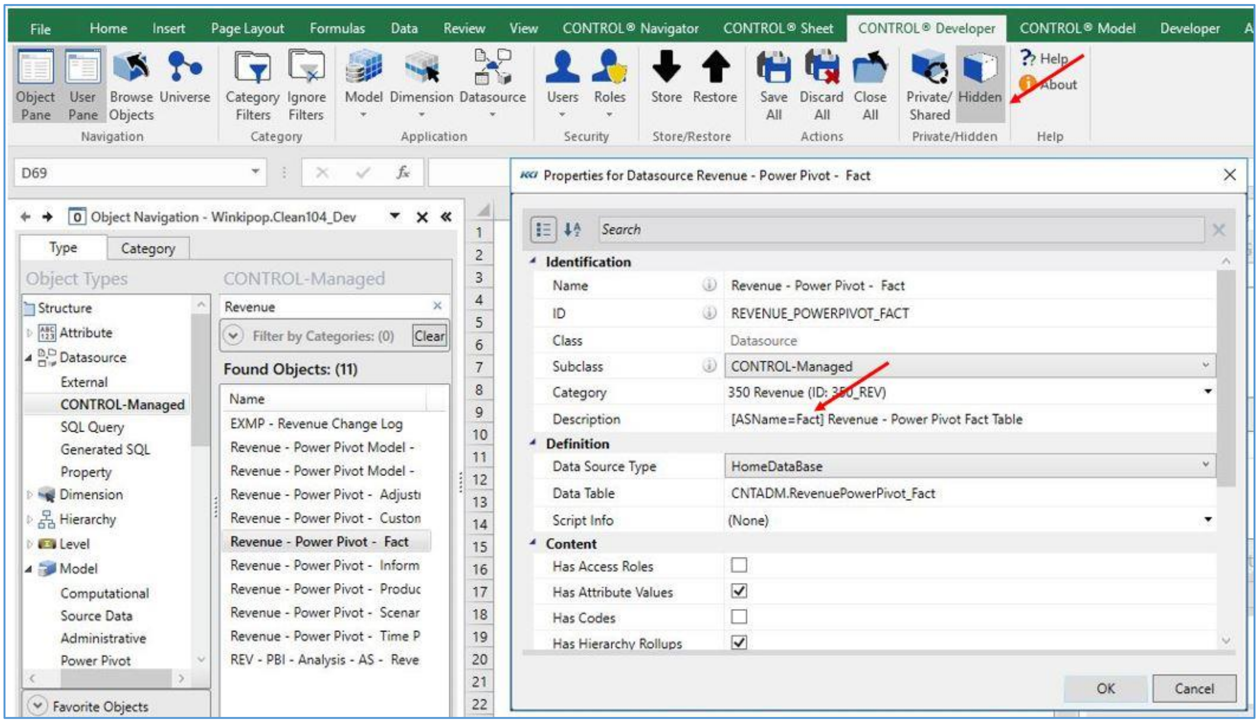
Customer

Information

Product

Scenario

To override a table name, you must find the datasource (it is hidden by default) and change the part of the description in brackets reading "[ASName=….]":



Properties for Datasource Revenue - Power Pivot - Fact

Identification	
Name	Revenue - Power Pivot - Fact
ID	REVENUE_POWERPIVOT_FACT
Class	Datasource
Subclass	CONTROL-Managed
Category	350 Revenue (ID: 350_REV)
Description	[ASName=Fact] Revenue - Power Pivot Fact Table
Definition	
Data Source Type	HomeDataBase
Data Table	CNTADM.RevenuePowerPivot_Fact
Script Info	(None)
Content	
Has Access Roles	□
Has Attribute Values	☑
Has Codes	□
Has Hierarchy Rollups	☑

Re-use exported relational tables

In an enterprise application environment, instead of creating relation tables for dimensions for each of our **Power Pivot** models we could consider setting them up once and then re-using them in multiple **Power Pivot** models.

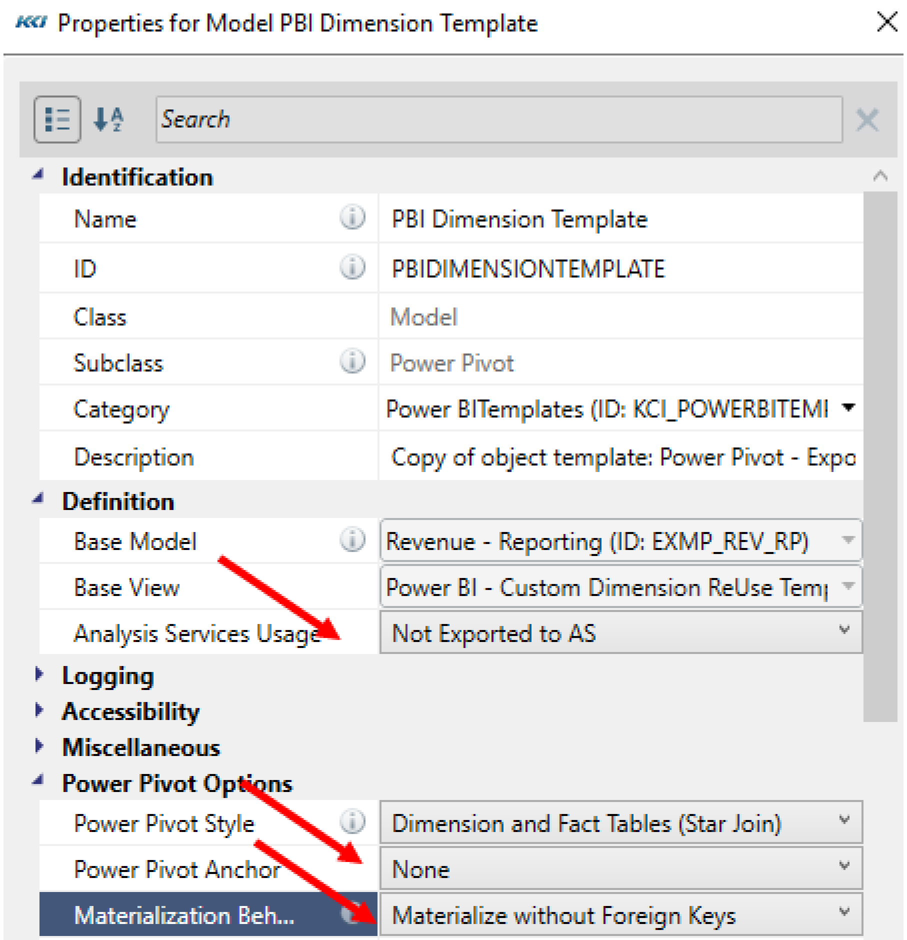
Dimensional data sources created in the export of a power pivot model will be reused when there is an existing data source that:

- Matches the dimension key.
- Matches the filter expression.
- Matches the branch.
- Is defined to be reusable.

To create reusable dimension relational tables, we suggest following the steps below:

- Create a template **Power Pivot** model based on a base and additional views that have the dimensions with the requisite settings for filter and branch generally used.
- For example, if you mostly use month and year levels for reporting and visualize data for Current and Prior year set up your time dimension in the base view accordingly.
- Set the Materialization Behavior for your **Power Pivot** models as **Materialize without Foreign Keys.** Since the same relational object is used in multiple models this ensures that we do not entangle ourselves due to the Foreign key relationships.

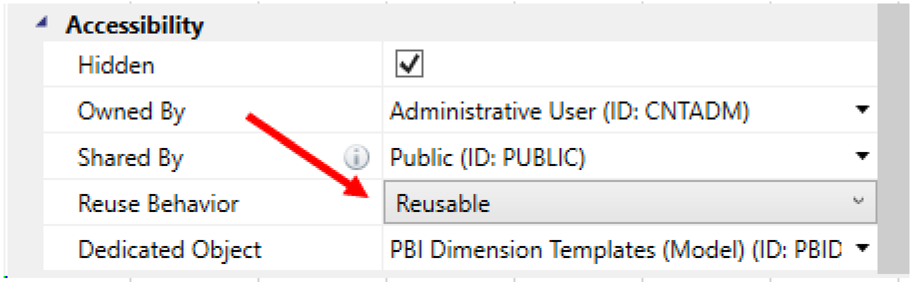
- The template model is only used to create the required dimension data sources and relational tables and not to create reports. So set the Analysis Services usage property as **Not Exported to AS.**
- Export the Power Pivot Model using the **Export > Create > Create Relational Objects** option



Properties for Model PBI Dimension Template

Identification	
Name	PBI Dimension Template
ID	PBIDIMENSIONTEMPLATE
Class	Model
Subclass	Power Pivot
Category	Power BITemplates (ID: KCI_POWERBITEMI
Description	Copy of object template: Power Pivot - Expo
Definition	
Base Model	Revenue - Reporting (ID: EXMP_REV_RP)
Base View	Power BI - Custom Dimension ReUse Temן
Analysis Services Usage	Not Exported to AS
Logging	
Accessibility	
Miscellaneous	
Power Pivot Options	
Power Pivot Style	Dimension and Fact Tables (Star Join)
Power Pivot Anchor	None
Materialization Beh...	Materialize without Foreign Keys

- After exporting the model edit the dimension data source objects' **Reuse Behavior** property to **Reusable.**



Accessibility	
Hidden	☑
Owned By	Administrative User (ID: CNTADM)
Shared By	Public (ID: PUBLIC)
Reuse Behavior	Reusable
Dedicated Object	PBI Dimension Templates (Model) (ID: PBID

- **Power Pivot** models created subsequently will be able to reuse these tables.

- The reusable datasources need to be kept up to date for changes in the dimensions by exporting the template **Power Pivot** model.

Table Storage

Power Pivot models with Materialization behavior set to **Materialize** or **Materialize without Foreign Keys** create relational tables when the model is exported. Materialized data tables on SQL Server 2019 and later will use a highly efficient Column Store Index.

Creating Visualizations in Microsoft Power BI desktop

The **Power Pivot** model's export works in two stages:

- **Stage 1** (Required): Creating relational tables or relational views of data and meta-data based on one or more CONTROL® computational or source data views.
- **Stage 2** (Optional): Creating a SQL Server Analysis Services Tabular model based on those relational tables or relational views.

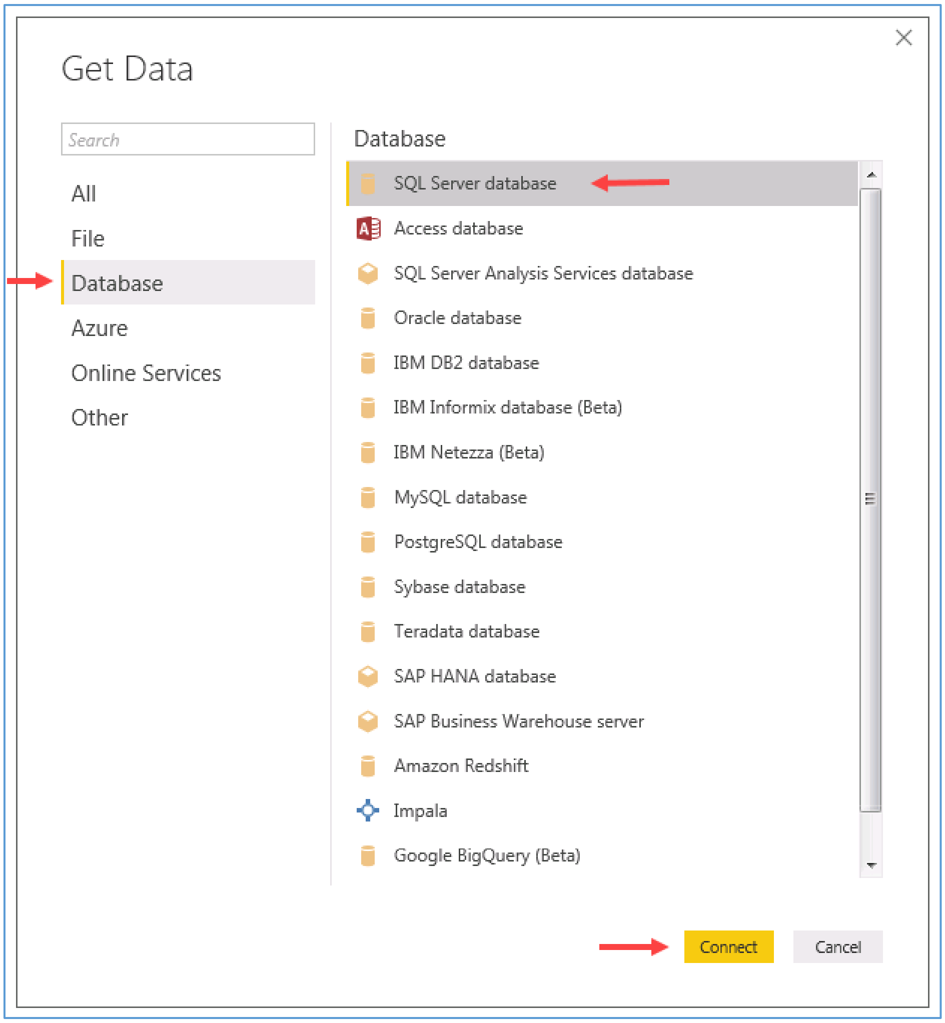
The output of either stage can be used by Power BI, Excel, or other products.

In the examples below, you will learn how to access data exported from a **Power Pivot** model to create visualizations in Microsoft Power BI:

SQL Server Database

You can create visualizations in Power BI by getting data from the relational tables and views in the SQL Server database.

1. Open your Power BI Desktop application and click **Get Data**.
2. Click **Database** > **SQL Server database** > **Connect**.



Get Data

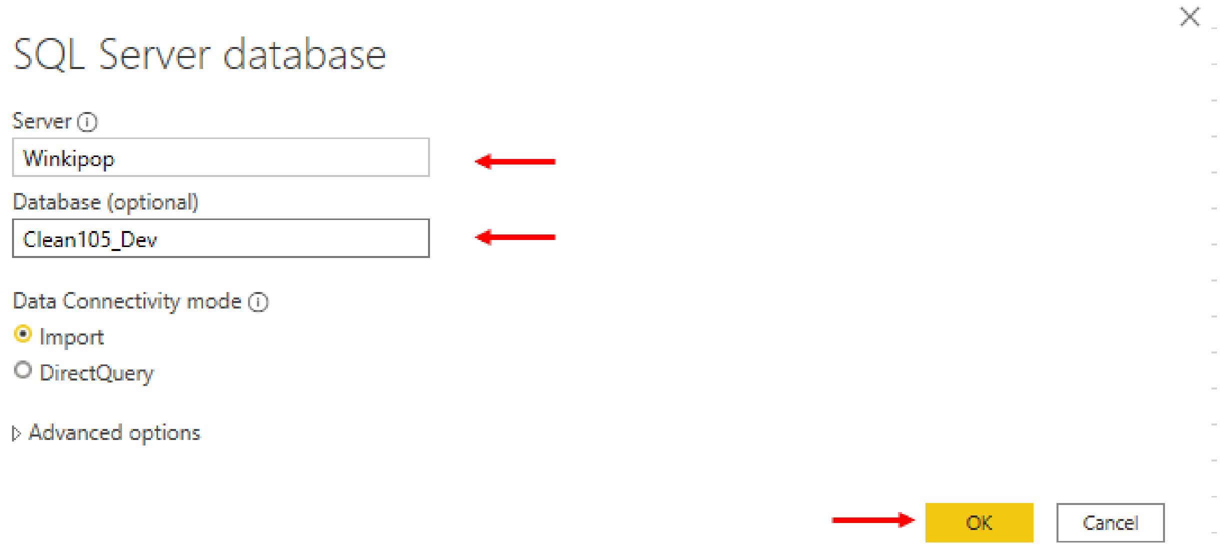
Search

All
File
Database
Azure
Online Services
Other

Database

SQL Server database
Access database
SQL Server Analysis Services database
Oracle database
IBM DB2 database
IBM Informix database (Beta)
IBM Netezza (Beta)
MySQL database
PostgreSQL database
Sybase database
Teradata database
SAP HANA database
SAP Business Warehouse server
Amazon Redshift
Impala
Google BigQuery (Beta)

3. Enter the name of your SQL Server and Database.



SQL Server database

Server: Winkipop

Database (optional): Clean105_Dev

Data Connectivity mode

● Import

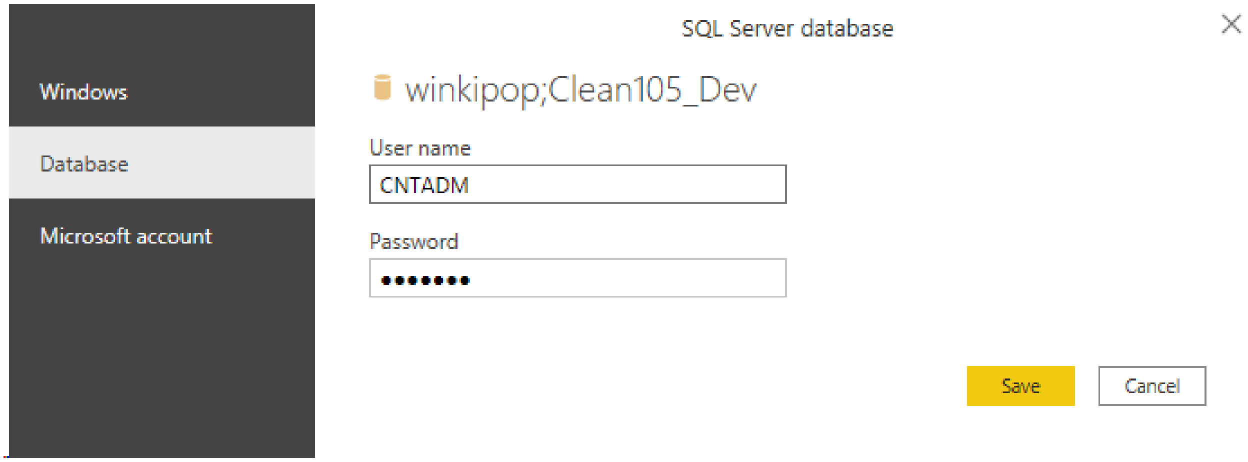
○ DirectQuery

There are two data connectivity modes **Import and DirectQuery** available when connecting to the SQL Server database. Here are the differences between the two modes:

Import: The selected tables and columns are imported into Power BI Desktop. As you create or interact with a visualization, Power BI Desktop uses the imported data. To see underlying data changes since the initial import or the most recent refresh, you must refresh the data, which imports the full dataset again.

DirectQuery: No data is imported or copied into Power BI Desktop. For relational sources, the selected tables and columns appear in the **Fields** list. As you create or interact with a visualization, Power BI Desktop queries the underlying data source, so you're always viewing current data.

4. Enter your user credentials for SQL Server.



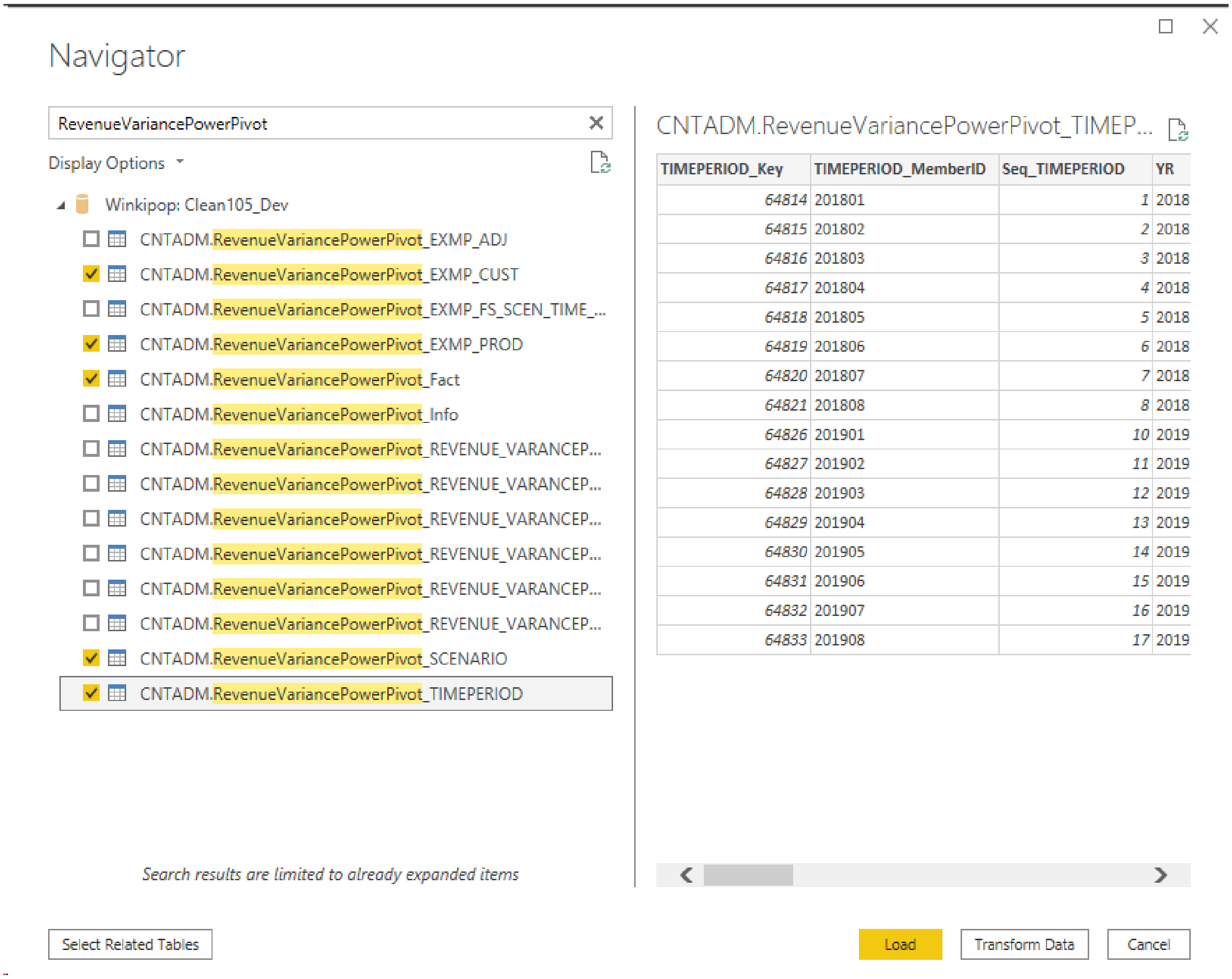
SQL Server database

winkipop;Clean105_Dev

User name: CNTADM

Password: •••••••

5. Search for your tables by name or prefix, check the box next to each of them, and then click **Load**.



Navigator

RevenueVariancePowerPivot

Winkipop: Clean105_Dev

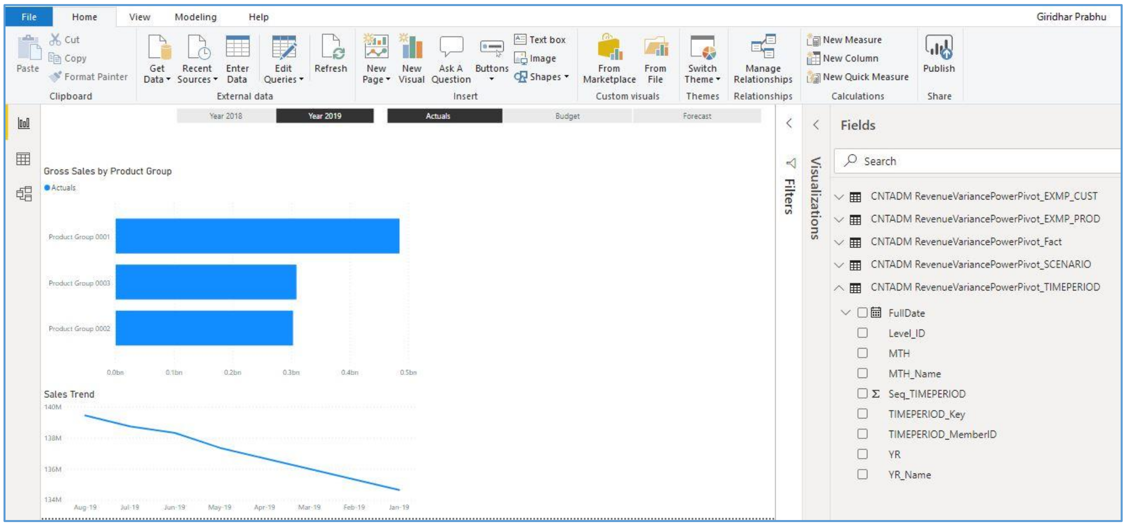
- ☐ CNTADM.RevenueVariancePowerPivot_EXMP_ADJ
- ☑ CNTADM.RevenueVariancePowerPivot_EXMP_CUST
- ☐ CNTADM.RevenueVariancePowerPivot_EXMP_FS_SCEN_TIME_...
- ☑ CNTADM.RevenueVariancePowerPivot_EXMP_PROD
- ☑ CNTADM.RevenueVariancePowerPivot_Fact
- ☐ CNTADM.RevenueVariancePowerPivot_Info
- ☐ CNTADM.RevenueVariancePowerPivot_REVENUE_VARANCEP...
- ☐ CNTADM.RevenueVariancePowerPivot_REVENUE_VARANCEP...
- ☐ CNTADM.RevenueVariancePowerPivot_REVENUE_VARANCEP...
- ☐ CNTADM.RevenueVariancePowerPivot_REVENUE_VARANCEP...
- ☐ CNTADM.RevenueVariancePowerPivot_REVENUE_VARANCEP...
- ☐ CNTADM.RevenueVariancePowerPivot_REVENUE_VARANCEP...
- ☑ CNTADM.RevenueVariancePowerPivot_SCENARIO
- ☑ CNTADM.RevenueVariancePowerPivot_TIMEPERIOD

Search results are limited to already expanded items

CNTADM.RevenueVariancePowerPivot_TIMEP...

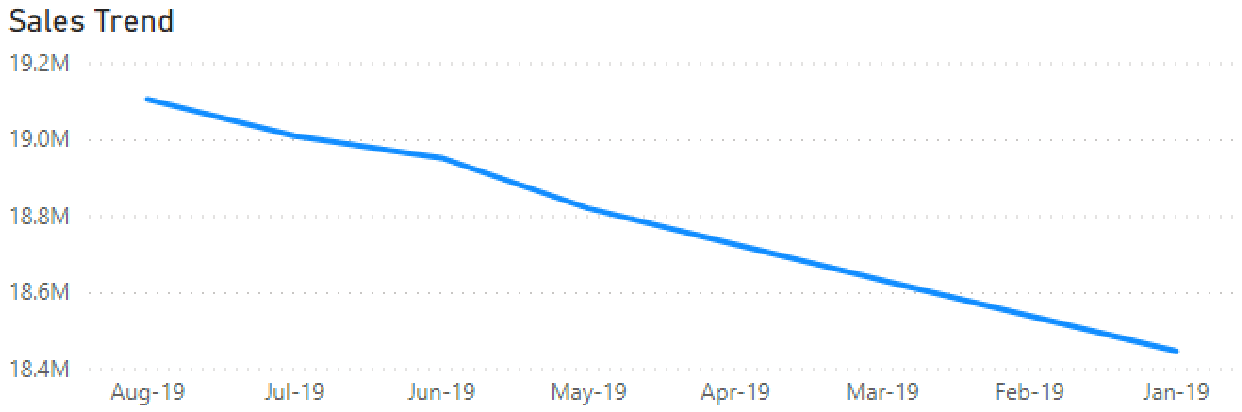
TIMEPERIOD_Key	TIMEPERIOD_MemberID	Seq_TIMEPERIOD	YR
64814	201801	1	2018
64815	201802	2	2018
64816	201803	3	2018
64817	201804	4	2018
64818	201805	5	2018
64819	201806	6	2018
64820	201807	7	2018
64821	201808	8	2018
64826	201901	10	2019
64827	201902	11	2019
64828	201903	12	2019
64829	201904	13	2019
64830	201905	14	2019
64831	201906	15	2019
64832	201907	16	2019
64833	201908	17	2019

6. After the tables load, you can begin creating your visualizations.



Note: If the **Materialization Behavior** property of your Power Pivot table is **Materialize**, the export process will automatically create foreign key relationships between the fact and the dimension tables, so you do not need define those in Power BI.

The variables/accounts are exported with default summation logic from the **Power Pivot** model to the relational tables in SQL Server database. You need to create measures involving ratios or any additional measures in Power BI.



You may notice in the screenshot above that the **Sales Trend** line chart does not have the months in the right order. No comparable type of order is available for hierarchy members when directly using the data from the SQL Server staging tables. You will need to perform additional steps to get the sorting working in your visualizations. Similarly, you need to rename the column names in Power BI to the desired name.

SQL Server Analysis Services

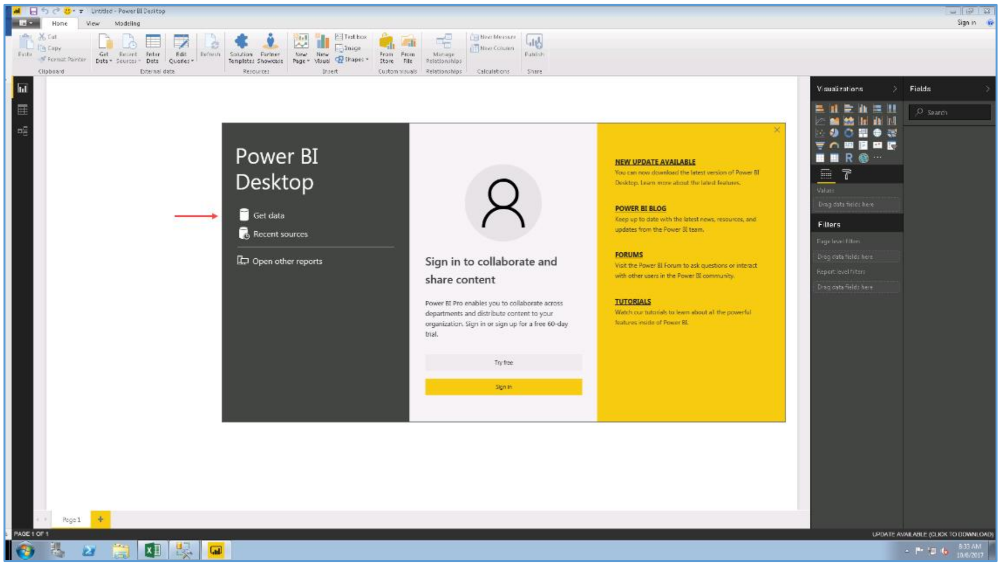
If you are already using Analysis Services in your organization then we suggest using it to create visualizations in Power BI.

The table below lists the advantages of using Analysis Services compared to connecting to SQL Server database

Functionality	Analysis Services
Naming and Clutter	The design pane in Power BI is readable and uncluttered. CONTROL® uses level and dimension names for columns in the dimension tables during the export to Analysis Services. However, if you wish you can override the names by updating the **Alias** field. Using **AS Usage** property, you can hide columns that are unlikely to be used.
Measures	Variables/accounts part of the CONTROL® view(s) in the **Power Pivot** model are exported with default summation logic. You can override the default summation logic and define additional DAX measures using the **AS Usage** and **AS Definitions** properties in the **Power Pivot** model. With everything already part of the AS Tabular model you are ready to create visualizations in Power BI without any additional work.
Sequencing	The member sequence in the hierarchies is passed on to Analysis Services. Visualizations in Power BI like a line chart using the time dimensions benefit with this ordering available by default.
Hierarchies	The parent-child relationship that is part of the hierarchy is used to create a hierarchy in the exported dimensions. You can use the hierarchy in Power BI to drill data starting at summary levels.
Calculation Groups	Calculation Groups included in your **Power Pivot** model are available for use in Power BI.

The following steps show you how to create visualizations in Power BI by getting data from an Analysis Services Tabular model:

1. Open your Power BI Desktop application and click **Get Data**.



Sign in to collaborate and share content

Power BI Pro enables you to collaborate across departments and distribute content to your organization. Sign in or sign up for a free 60-day trial.

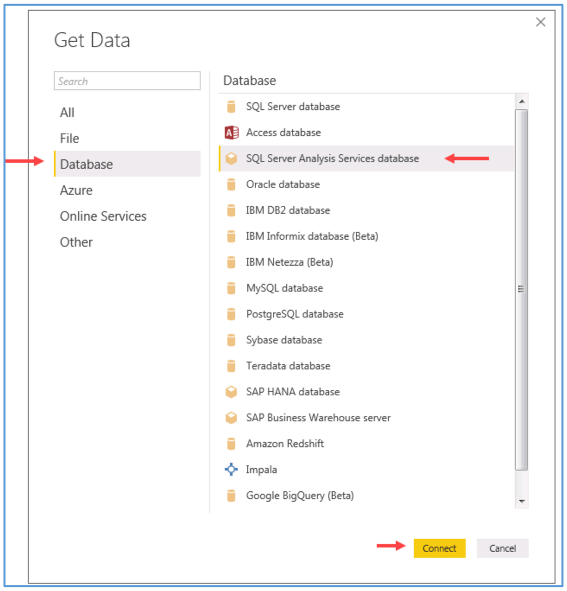
NEW UPDATE AVAILABLE
You can now download the latest version of Power BI Desktop. Learn more about the latest features.

POWER BI BLOG
Keep up to date with the latest news, resources, and updates from the Power BI team.

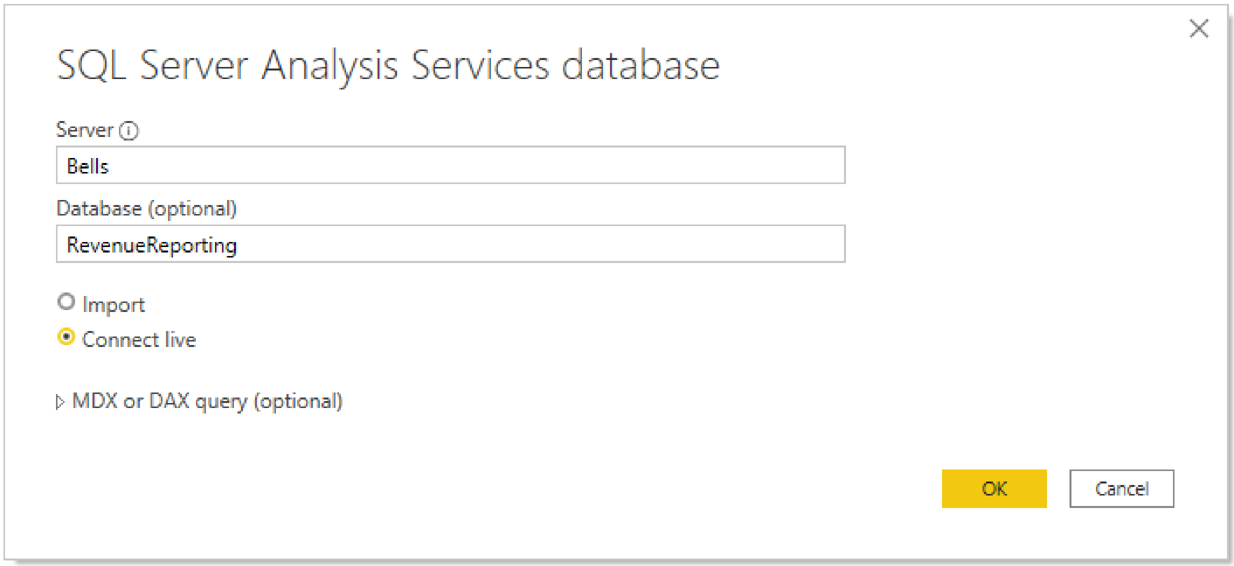
FORUMS
Visit the Power BI Forum to ask questions or interact with other users in the Power BI community.

TUTORIALS
Watch our tutorials to learn about all the powerful features inside of Power BI.

2. Click **Database** > **SQL Server Analysis Services database** > **Connect**.



3. Enter the name of your SQL Server Analysis Services server and database, leave the connection mode as **connect live** and click **OK**.



SQL Server Analysis Services database

Server ⓘ: Bells

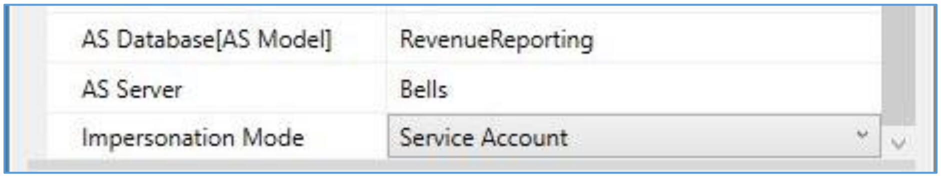
Database (optional): RevenueReporting

○ Import

● Connect live

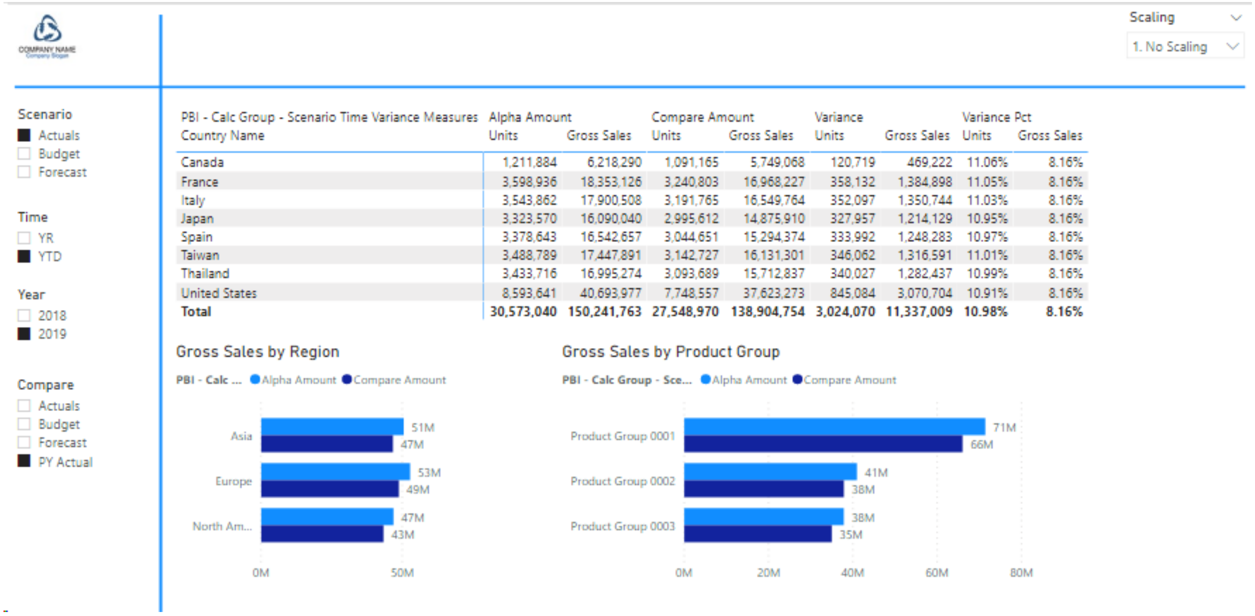
▷ MDX or DAX query (optional)

Note: You can find this information in your **Power Pivot** model's properties dialogue under the **Power Pivot Options** group. If **AS Server** is not specified in the **Power Pivot** model the replacement value of keyword **KCI_ASSERVER** serves as the Analysis Services Server for the export.



AS Database[AS Model]	RevenueReporting
AS Server	Bells
Impersonation Mode	Service Account

4. Once you are in the Power BI designer, you can easily create interesting visualizations that you can save and share or publish to either an on-premises server or Microsoft's Power BI cloud server.



Scaling: 1. No Scaling

Scenario
- ■ Actuals
- □ Budget
- □ Forecast

Time
- □ YR
- ■ YTD

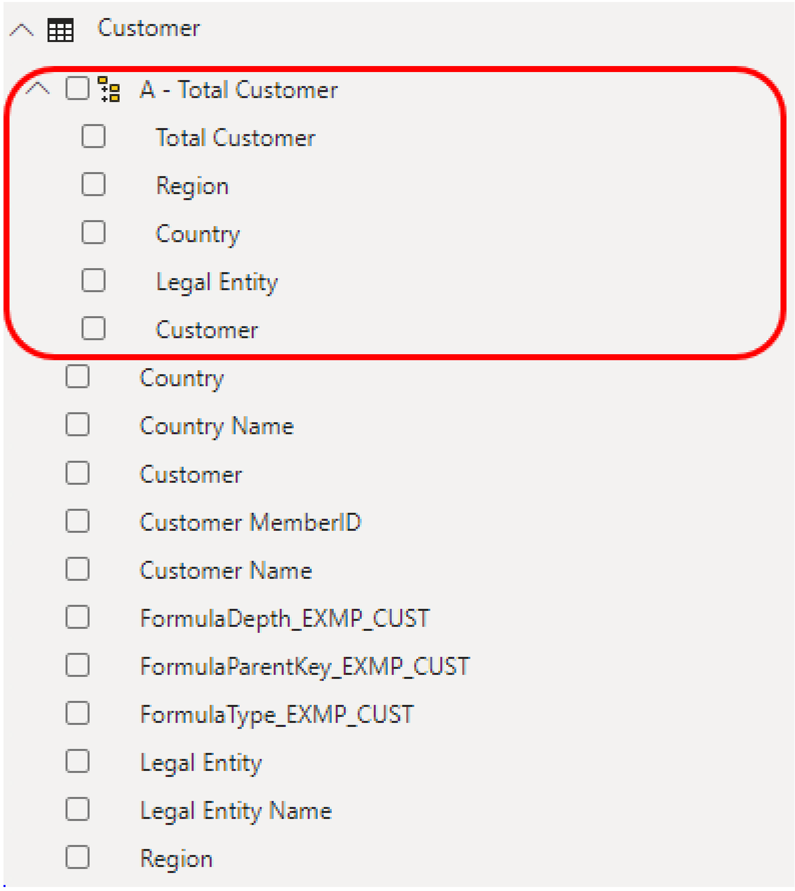
Year
- □ 2018
- ■ 2019

Compare
- □ Actuals
- □ Budget
- □ Forecast
- ■ PY Actual

PBI - Calc Group - Scenario Time Variance Measures	Alpha Amount		Compare Amount		Variance		Variance Pct	
Country Name	Units	Gross Sales	Units	Gross Sales	Units	Gross Sales	Units	Gross Sales
Canada	1,211,884	6,218,290	1,091,165	5,749,068	120,719	469,222	11.06%	8.16%
France	3,598,936	18,353,126	3,240,803	16,968,227	358,132	1,384,898	11.05%	8.16%
Italy	3,543,862	17,900,508	3,191,765	16,549,764	352,097	1,350,744	11.03%	8.16%
Japan	3,323,570	16,090,040	2,995,612	14,875,910	327,957	1,214,129	10.95%	8.16%
Spain	3,378,643	16,542,657	3,044,651	15,294,374	333,992	1,248,283	10.97%	8.16%
Taiwan	3,488,789	17,447,891	3,142,727	16,131,301	346,062	1,316,591	11.01%	8.16%
Thailand	3,433,716	16,995,274	3,093,689	15,712,837	340,027	1,282,437	10.99%	8.16%
United States	8,593,641	40,693,977	7,748,557	37,623,273	845,084	3,070,704	10.91%	8.16%
Total	**30,573,040**	**150,241,763**	**27,548,970**	**138,904,754**	**3,024,070**	**11,337,009**	**10.98%**	**8.16%**

Here are some important details about the exported structures:

1. Each view dimension-branch has become a dimension table, and all the levels in the branch are available for use in visualizations.
2. The branch itself has become a "hierarchy" (in the Analysis Services sense), so you can navigate up and down the levels of detail.



Customer

☐ A - Total Customer

☐ Total Customer

☐ Region

☐ Country

☐ Legal Entity

☐ Customer

☐ Country

☐ Country Name

☐ Customer

☐ Customer MemberID

☐ Customer Name

☐ FormulaDepth_EXMP_CUST

☐ FormulaParentKey_EXMP_CUST

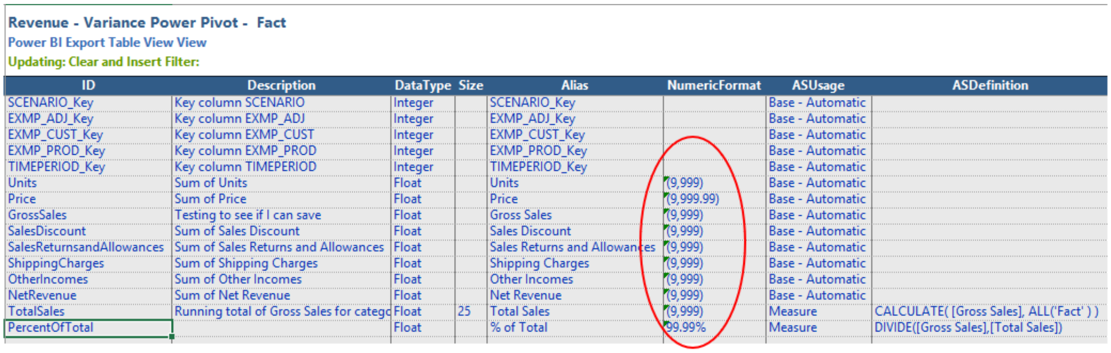
☐ FormulaType_EXMP_CUST

☐ Legal Entity

☐ Legal Entity Name

☐ Region

3. CONTROL® exports its numeric formatting, to the extent possible, to display in Power BI.



Revenue - Variance Power Pivot - Fact

Power BI Export Table View View

Updating: Clear and Insert Filter:

ID	Description	DataType	Size	Alias	NumericFormat	ASUsage	ASDefinition
SCENARIO_Key	Key column SCENARIO	Integer		SCENARIO_Key		Base - Automatic	
EXMP_ADJ_Key	Key column EXMP_ADJ	Integer		EXMP_ADJ_Key		Base - Automatic	
EXMP_CUST_Key	Key column EXMP_CUST	Integer		EXMP_CUST_Key		Base - Automatic	
EXMP_PROD_Key	Key column EXMP_PROD	Integer		EXMP_PROD_Key		Base - Automatic	
TIMEPERIOD_Key	Key column TIMEPERIOD	Integer		TIMEPERIOD_Key		Base - Automatic	
Units	Sum of Units	Float		Units	(9,999)	Base - Automatic	
Price	Sum of Price	Float		Price	(9,999.99)	Base - Automatic	
GrossSales	Testing to see if I can save	Float		Gross Sales	(9,999)	Base - Automatic	
SalesDiscount	Sum of Sales Discount	Float		Sales Discount	(9,999)	Base - Automatic	
SalesReturnsandAllowances	Sum of Sales Returns and Allowances	Float		Sales Returns and Allowances	(9,999)	Base - Automatic	
ShippingCharges	Sum of Shipping Charges	Float		Shipping Charges	(9,999)	Base - Automatic	
OtherIncomes	Sum of Other Incomes	Float		Other Incomes	(9,999)	Base - Automatic	
NetRevenue	Sum of Net Revenue	Float		Net Revenue	(9,999)	Base - Automatic	
TotalSales	Running total of Gross Sales for catego	Float	25	Total Sales	(9,999)	Measure	CALCULATE([Gross Sales], ALL('Fact'))
PercentOfTotal		Float		% of Total	99.99%	Measure	DIVIDE([Gross Sales],[Total Sales])

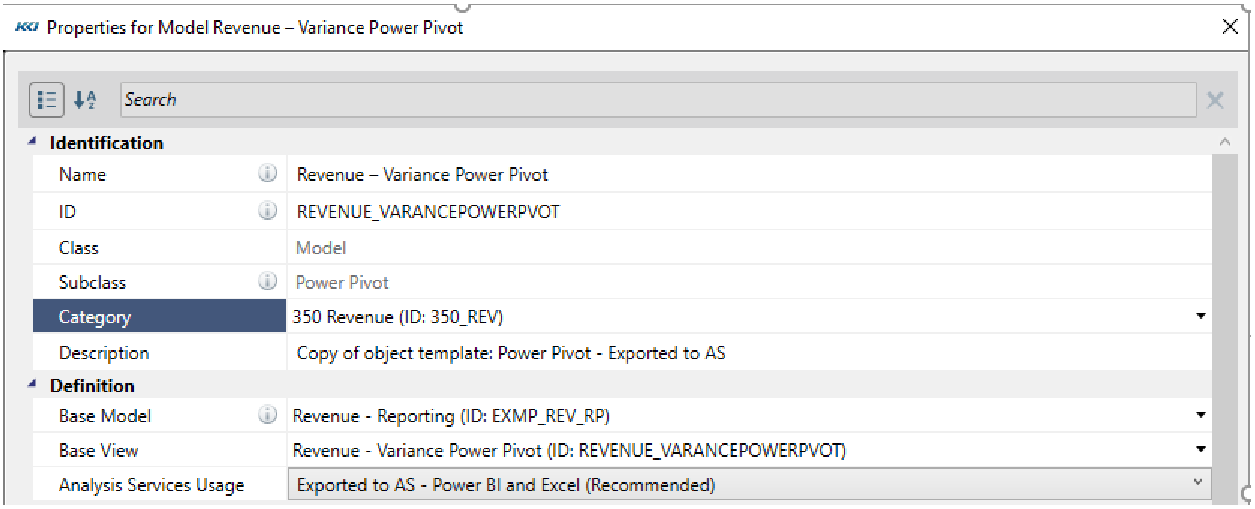
4. CONTROL® exports the member sequence to Power BI. In the screenshot below, you can see that the Month ID sorts the month, though the visualization shows the Month names.

Sales Trend
● Forecast
160M
150M
140M
Jan-19
Feb-19
Mar-19
Apr-19
May-19
Jun-19
Jul-19
Aug-19

Excel Pivot Table using an Analysis Services Model Example

If you like using Pivot Table in Excel, it is very simple to connect to AS and access a model built by CONTROL®, and you do not need a CONTROL® license to do so.

Please note that to use Excel Pivot Table with your **Power Pivot** model, you need to set the **Analysis Services Usage** property to **Exported to AS – Power BI and Excel (Recommended).** This setting is required as Power BI and Excel Pivot Table handle **Sum** measures differently. Setting this property appropriately will ensure the data is appropriately exported for Excel Pivot Table.



Identification

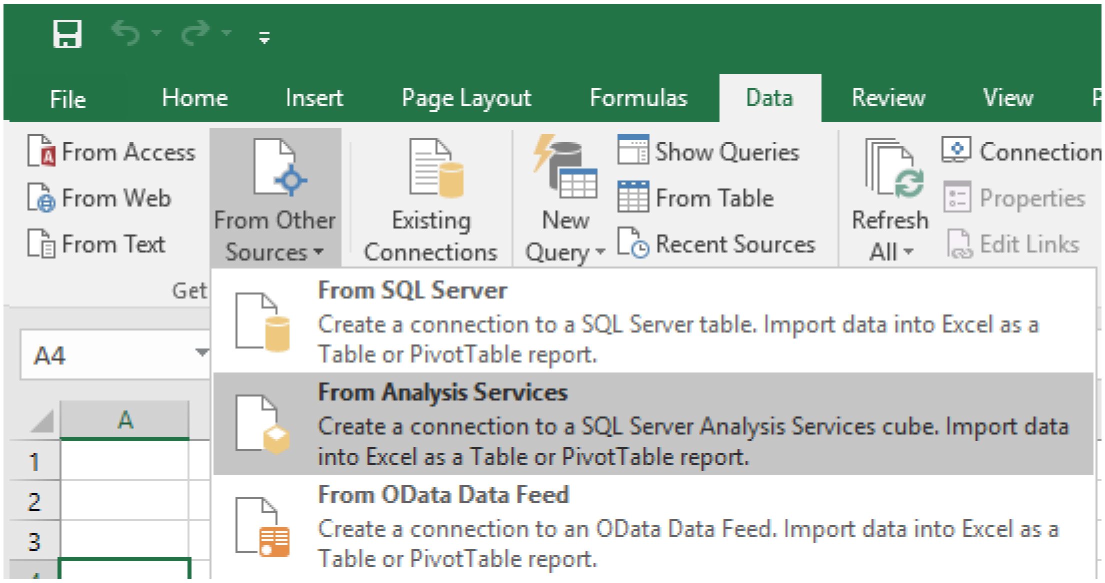
Name	Revenue – Variance Power Pivot
ID	REVENUE_VARANCEPOWERPVOT
Class	Model
Subclass	Power Pivot
Category	350 Revenue (ID: 350_REV)
Description	Copy of object template: Power Pivot - Exported to AS

Definition

Base Model	Revenue - Reporting (ID: EXMP_REV_RP)
Base View	Revenue - Variance Power Pivot (ID: REVENUE_VARANCEPOWERPVOT)
Analysis Services Usage	Exported to AS - Power BI and Excel (Recommended)

In Excel (2016 or later):

1. From the **Data** ribbon, select **From Other Sources > From Analysis Services**

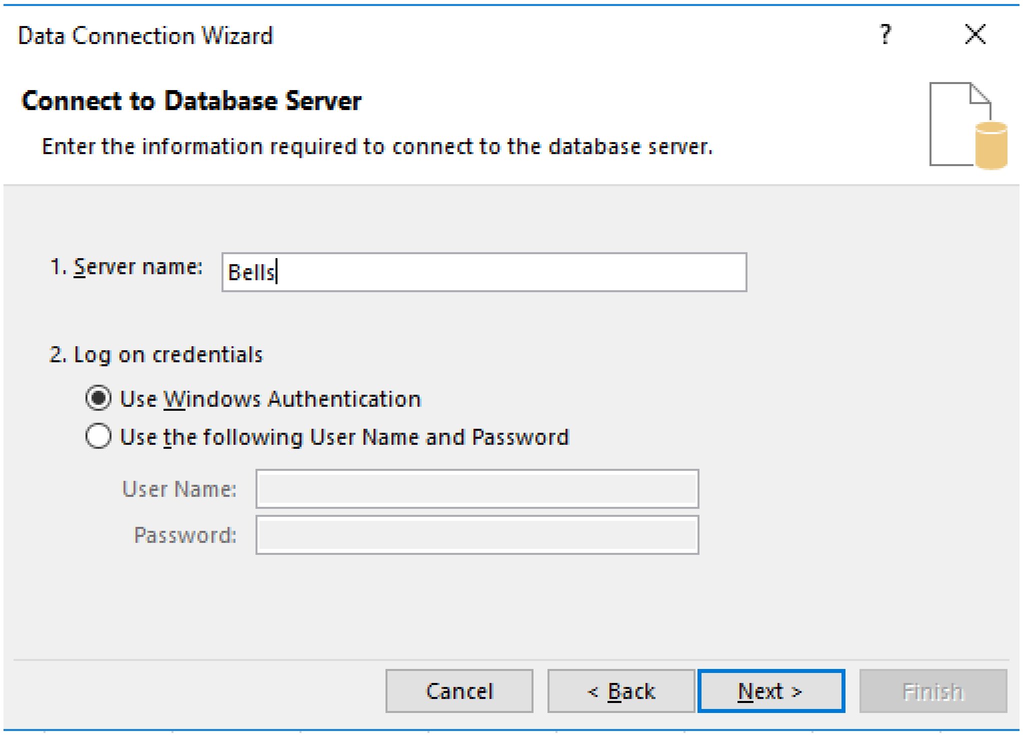


From SQL Server
Create a connection to a SQL Server table. Import data into Excel as a Table or PivotTable report.

From Analysis Services
Create a connection to a SQL Server Analysis Services cube. Import data into Excel as a Table or PivotTable report.

From OData Data Feed
Create a connection to an OData Data Feed. Import data into Excel as a Table or PivotTable report.

2. Specify the database:



Connect to Database Server

Enter the information required to connect to the database server.

1. Server name: Bells

2. Log on credentials

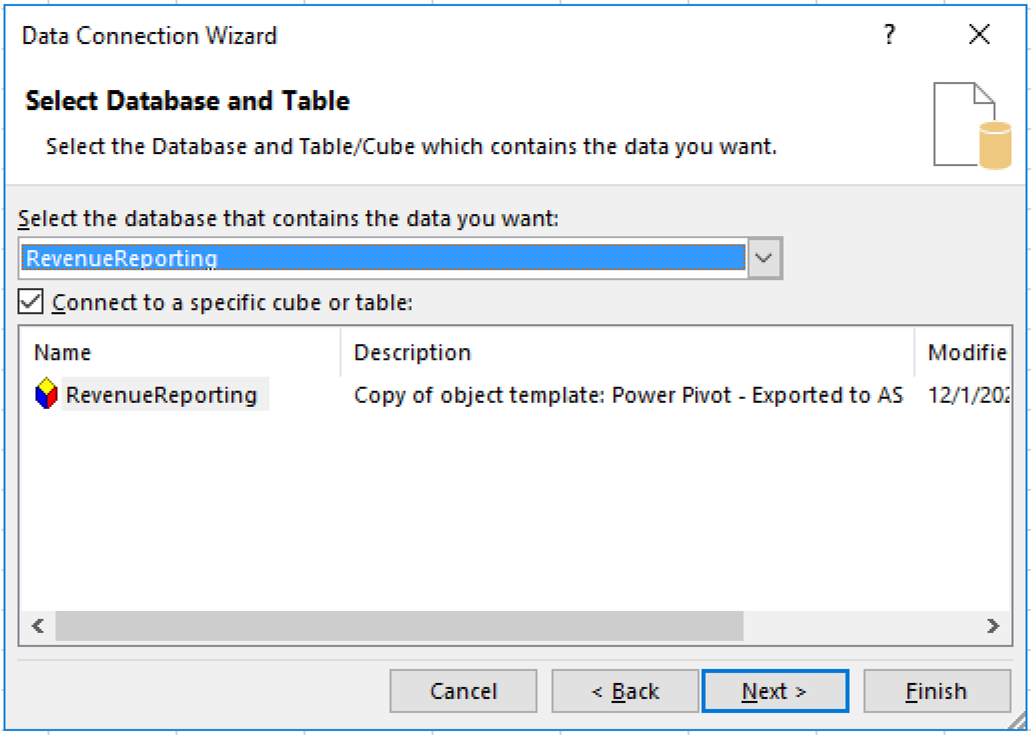
● Use Windows Authentication

○ Use the following User Name and Password

User Name:

Password:

3. Select the model and click **Next**.



Data Connection Wizard

Select Database and Table

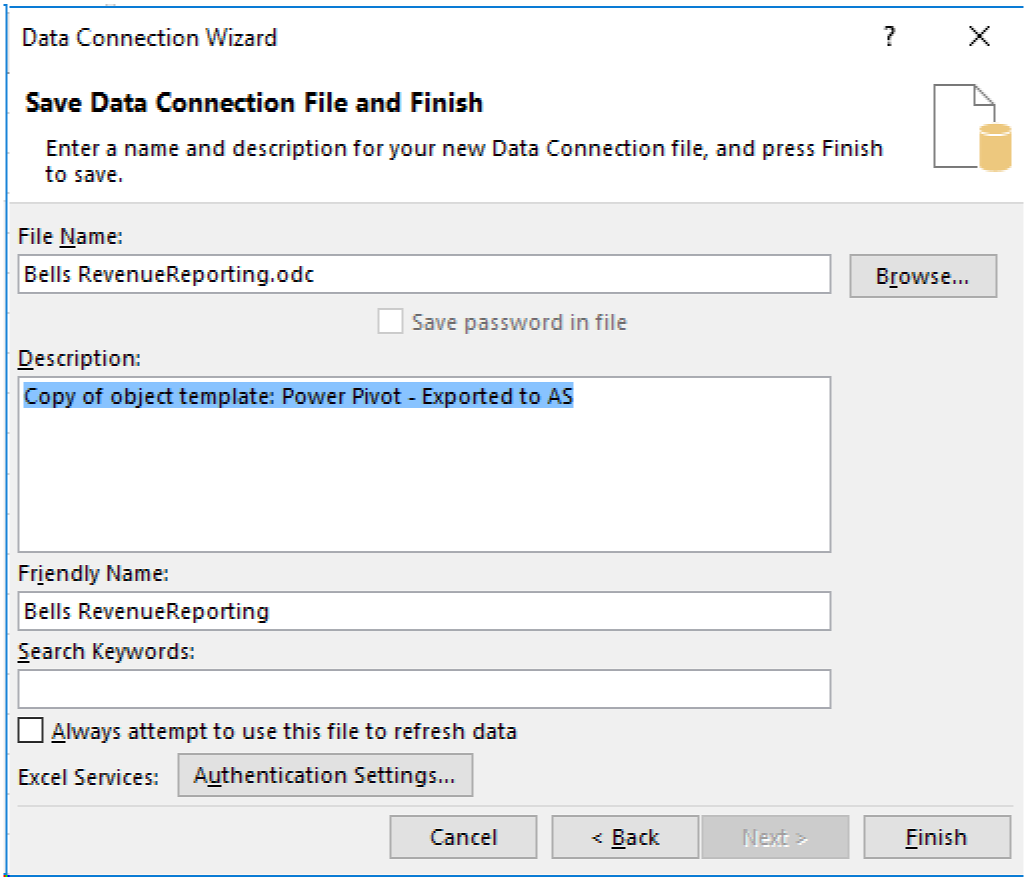
Select the Database and Table/Cube which contains the data you want.

Select the database that contains the data you want: RevenueReporting

☑ Connect to a specific cube or table:

Name	Description	Modifie
RevenueReporting	Copy of object template: Power Pivot - Exported to AS	12/1/202

4. Enter a name for the connection file and click **Finish**.



Data Connection Wizard

Save Data Connection File and Finish

Enter a name and description for your new Data Connection file, and press Finish to save.

File Name: Bells RevenueReporting.odc

□ Save password in file

Description: Copy of object template: Power Pivot - Exported to AS

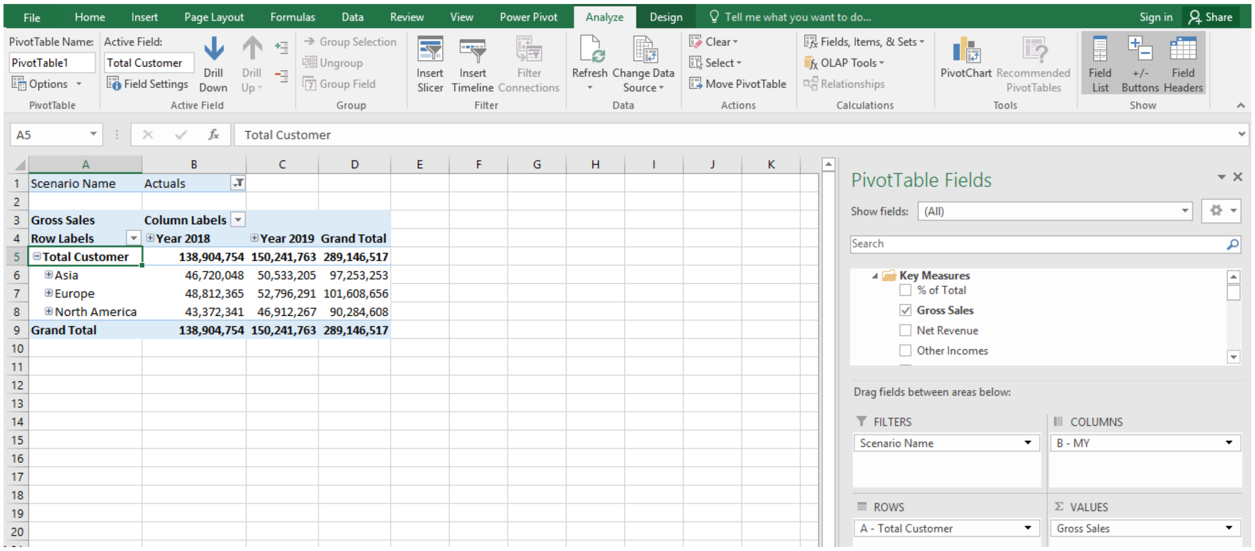
Friendly Name: Bells RevenueReporting

Search Keywords:

□ Always attempt to use this file to refresh data

Excel Services: Authentication Settings...

5. Choose what you want to do (Pivot Table in this example), and design whatever you would like:



PivotTable Name: PivotTable1

Active Field: Total Customer

A5 | Total Customer

	A	B	C	D
1	Scenario Name	Actuals		
2				
3	**Gross Sales**	**Column Labels**		
4	**Row Labels**	**Year 2018**	**Year 2019**	**Grand Total**
5	**Total Customer**	**138,904,754**	**150,241,763**	**289,146,517**
6	Asia	46,720,048	50,533,205	97,253,253
7	Europe	48,812,365	52,796,291	101,608,656
8	North America	43,372,341	46,912,267	90,284,608
9	**Grand Total**	**138,904,754**	**150,241,763**	**289,146,517**

PivotTable Fields

Show fields: (All)

Key Measures
- □ % of Total
- ☑ Gross Sales
- □ Net Revenue
- □ Other Incomes

Drag fields between areas below:

FILTERS: Scenario Name

COLUMNS: B - MY

ROWS: A - Total Customer

VALUES: Gross Sales

Pivot Table and Pivot Chart view styles

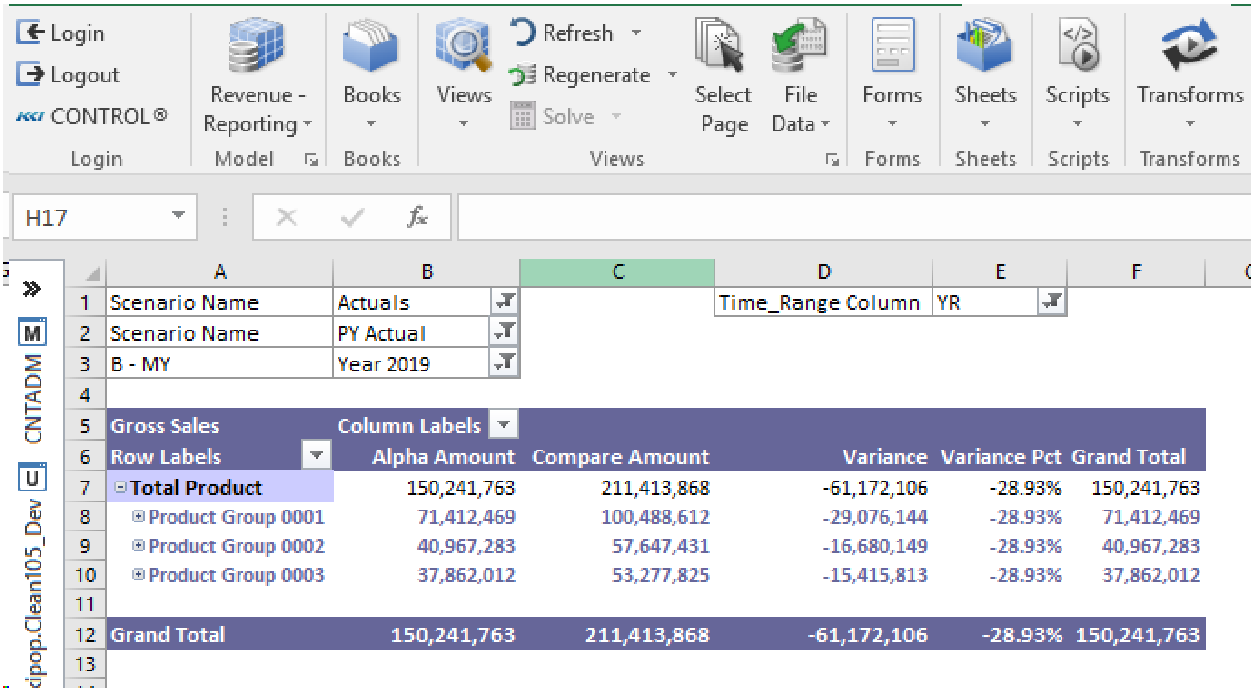
The Pivot Table and Pivot Chart *view* styles have been available in CONTROL® in previous releases, but their behavior has been significantly changed and improved with version 10.5. The motivation for this enhancement is to support casual users who are familiar and comfortable with Excel Pivot Tables, but are not trained in all the features of CONTROL® *views*.

When you save a Pivot Table or Pivot Chart style *view*, the worksheet containing the pivot table or chart is saved, including all customizations such as field assignments, formatting, added calculations, slicers, etc.

When you re-open the *view*, the data is refreshed from the underlying CONTROL® *model* and *view*, preserving all of your pivot table/chart customizations.

When you create a new *view* on a **Power Pivot** *model* that has a **CONTROL®-managed** or externally managed Analysis Services model, the Pivot Table or chart is connected to the Analysis Services model directly, and you can make use of all the capabilities on Excel's **PivotTable Analyze** ribbon, including the calculation tools for OLAP.

Here's a screenshot of a view on our **Revenue – Variance Power Pivot** model.



	A	B	C	D	E	F
1	Scenario Name	Actuals		Time_Range Column	YR	
2	Scenario Name	PY Actual				
3	B - MY	Year 2019				
4						
5	Gross Sales	Column Labels				
6	Row Labels	Alpha Amount	Compare Amount	Variance	Variance Pct	Grand Total
7	⊟Total Product	150,241,763	211,413,868	-61,172,106	-28.93%	150,241,763
8	⊞Product Group 0001	71,412,469	100,488,612	-29,076,144	-28.93%	71,412,469
9	⊞Product Group 0002	40,967,283	57,647,431	-16,680,149	-28.93%	40,967,283
10	⊞Product Group 0003	37,862,012	53,277,825	-15,415,813	-28.93%	37,862,012
11						
12	Grand Total	150,241,763	211,413,868	-61,172,106	-28.93%	150,241,763
13						

The **Power Pivot Table** and **Power Pivot Chart** *views* introduced in release 10.2 are no longer supported.

Additional resources

For a deeper understanding of the features available in various Microsoft Business Intelligence (BI) products, please see the following references:

Power BI: https://powerbi.microsoft.com/en-us/

Analysis Services: https://www.microsoft.com/en-us/sql-server/sql-business-intelligence

Excel Power Pivot: https://support.office.com/en-us/article/Power-Pivot-Powerful-data-analysis-and-data-modeling-in-Excel-A9C2C6E2-CC49-4976-A7D7-40896795D045

Power BI Deployment White Paper: https://docs.microsoft.com/en-us/power-bi/guidance/whitepapers

CONTROL® environment setup

Recommendation for the configuration of a CONTROL® environment to support Microsoft Power BI and SQL Server Analysis Services is available from KCI in the document titled **CONTROL® and Microsoft Power BI**.