DHIS2-to-PowerBI Connector

USER MANUAL

PREPARED BY

BAO Systems
2900 K St. NW
Suite 507
Washington, DC 20007

July 31, 2019

Version 1.0

Contents

Introduction

Funders are increasingly demanding robust data use for adaptive program management. While a number of global health and development organizations implement DHIS 2 to collect, manage, and visualize routine program data, users are increasingly demanding more advanced analytics capabilities beyond the software's core functionality. In addition, organizations maintain multiple datasets which are housed in other formats such as Excel, Access, ODK/Kobo/ONA, SQL, Sharepoint, Salesforce, and others. As a result, there is growing demand for more robust data analytics platforms that enable data mashups from the variety of sources where data are stored as well as more advanced visualization features.

One of many popular business intelligence ("BI") tools used by organizations is Microsoft Power BI. The tool can be broken down into several tools such as: Power Query is used to extract and transform data; Power Pivot is mainly used to model and analyze data; and Power View and Map tool is used to visualize data. Power BI enables the design of interactive visualizations which can be used to create reports and dashboards. It provides a cloud-based BI service which is known as "Power BI Services", also providing a desktop based interface which is known as "Power BI Desktop". It offers data warehouse capabilities such as data preparation, data discovery and interactive dashboards. It is also popular for its ability to load custom visualizations.

The use of PowerBI for visualizing DHIS2 data is increasing amongst global health and development organizations, however, users often require support to more efficiently exchange data between DHIS2 and Power BI. Currently, many users perform a manual and time consuming data exchange process, exporting and importing csv files of clean, aggregate data from DHIS2 into PowerBI, rather than utilizing DHIS2's API.

BAO Systems developed the DHIS2-to-PowerBI Connector as a user-friendly utility designed to enable users to more seamlessly:

- Connect to DHIS2 via user authentication
- Get data from DHIS2 for querying
- Create an optimal data model and shape data for robust analytics in PowerBI
- Refresh the data model with the latest DHIS2 data

Getting started

User profile

DHIS2 proficiency

Individuals using the DHIS2-to-PBI connector should have DHIS2 configuration and/or analytics competency, but may have little to no experience accessing the DHIS2 API. Users should have a basic understanding of DHIS2 architecture, particularly with regards to the relationships between: organisational units; category option combinations and attribute option combinations; data element group sets, data element groups, and data elements; and data element values. This knowledge is helpful for data modelling in Power BI.

Power BI proficiency

It is recommended that users of the connector have competency in Power BI in order to better take advantage of the tool's capabilities. The resulting data model from this user guide will enable the user to have maximum data granularity and analytics flexibility, allowing for drill downs and cross-filtering functions. Once data are brought in to Power BI, there will be a need to perform some basic data modelling and future formula creation using DAX.

What the connector does

As an extensible system, DHIS2 can be implemented in creative ways. Data structures may vary widely between project, program, and enterprise systems. A custom connector provides a generic interface and must make certain assumptions about the data structures, access levels, etc. The connector and this accompanying user guide was designed to help users get DHIS2 data into Power BI and set up a functional data model for querying.

At time of writing, the connector allows import of:

- Organisation Unit metadata (basic attributes including Lat/Long)
- Data Element metadata (core attributes)
- Data Element Values (numeric values associated with Data Element between Start/End Date range for OU Levels specified by the PBI data modeller)
- Dynamic Dimensions
    - Categories
    - Attributes
    - Category Option Group Sets
    - Organisation Unit Group Sets
    - Data Element Group Sets
- A Date table (dynamically generated by the Connector)
- A Periods table (a dynamically generated 'join' table to facilitate connecting Dates with Data Element Values)

This user guide provides additional support around how to perform basic data modelling functions to enhance the data model in Power BI, making it easier for the end user to query the data.

Connector limitations

As mentioned above, assumptions were made in the development of this connector, and as such, the connector and our recommended data modelling steps may not apply for 100% of use cases. Development of this connector and user guide made some assumptions on the quality of DHIS2 configuration, and thus your resulting data model after import of DHIS2 data may look different.

For illustrative purposes, this guide provides steps and examples from connecting to the play.dhis2 instance.

At the time of writing, connector limitations are described below:

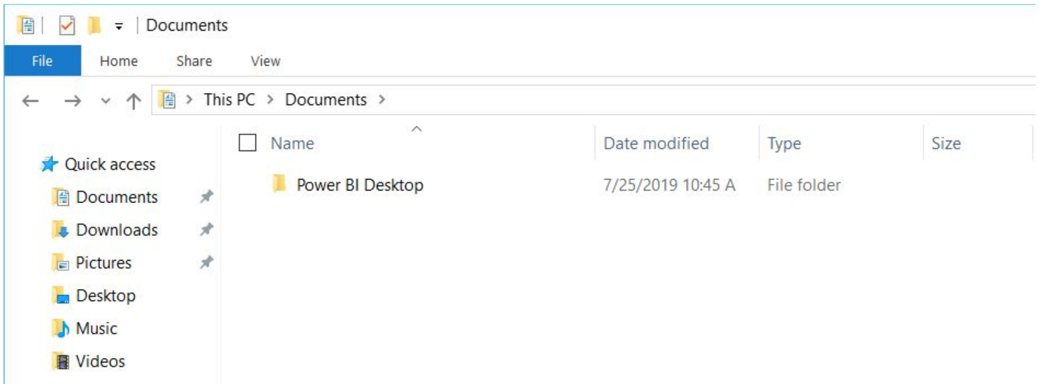
- The connector currently functions with DHIS2 versions 2.29, 2.30, 2.31 , 2.32 (API versions 29, 30, 31, and 32 respectively). The connector has a known incompatibility with DHIS2 version 2.28 and has not been tested with versions prior.
- The connector imports data element values in the DHIS2 Aggregate data model. It does not currently import Indicators, Events, and Tracker models, however these features are on the connector's roadmap for future development efforts.
- The connector does not provide a mechanism to pre-filter imported data element values using, for example, selected data element groups, data sets, or organisational units. All data element values available to the users' credentials (per DHIS2 user org assignment) will be imported.
- The connector retrieves Data Element Values using the /analytics/rawData API endpoint; to ensure your Power BI data is up to date, be sure to refresh your DHIS2 analytics regularly.
- The Date and related Periods tables contain only a subset of the possible intervals available in DHIS2. While your Data Element Values will include every value between the start and end dates provided, some values may not appear in your visuals because there is no related period on which to join to the Date table. Out of the box, the connector generates intervals:
    - October based Financial Year, e.g., 2017Oct
    - Month, e.g., 201704
    - Quarter, e.g., 2018Q3
    - Week, e.g., 2017W19
    - Year, e.g., 2019

  Using the Query Editor, it is possible to extend the Date table to include additional intervals and build out the related Periods table, but this is an advanced topic beyond the scope of this document.

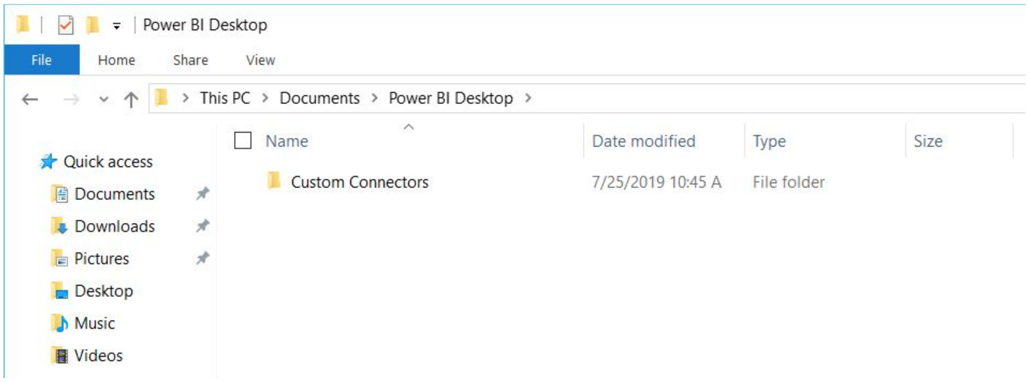
Installing the custom connector

The following steps will guide you through installation of the connector. If you encounter problems with the installation, please review [Microsoft's guidance](#) as it is possible the procedure has changed since the time of this writing.

1. Install the DHIS2 Custom Connector on your computer.
    a. Create the following folder path (*make sure that all spaces and casing are maintained)*:
    C:\Users\<Your User Name>\Documents\Power BI Desktop\Custom Connectors

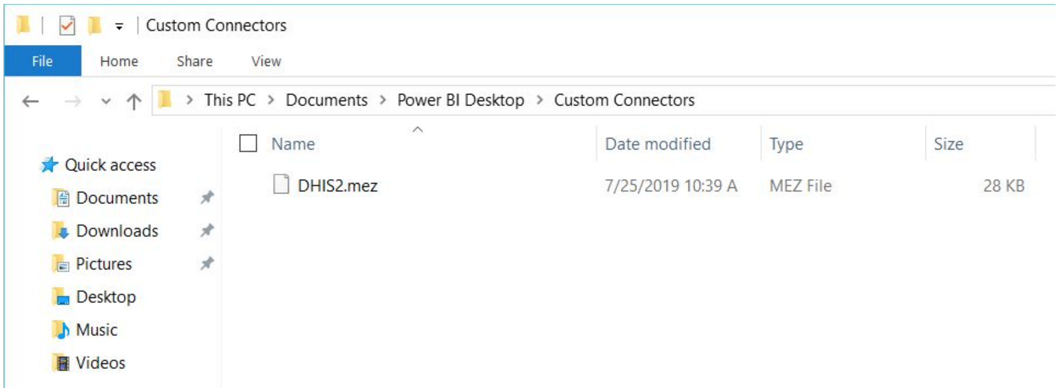


Name	Date modified	Type	Size
Power BI Desktop	7/25/2019 10:45 A	File folder	



Name	Date modified	Type	Size
Custom Connectors	7/25/2019 10:45 A	File folder	

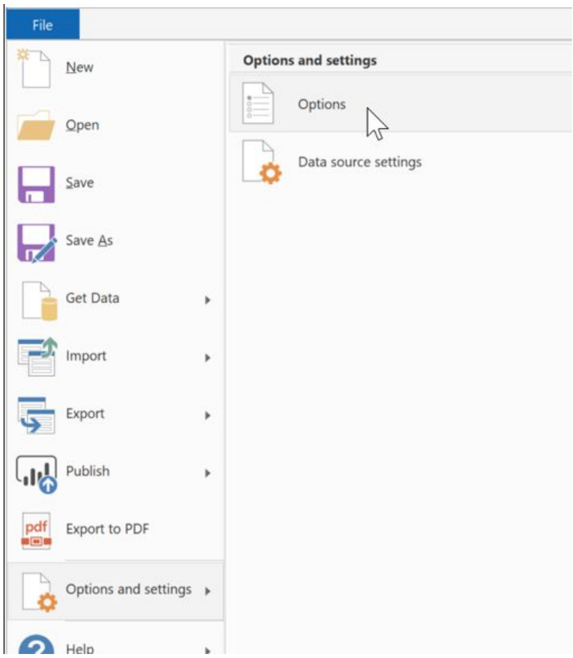
    b. Save the connector **.mez file** in the directory above.



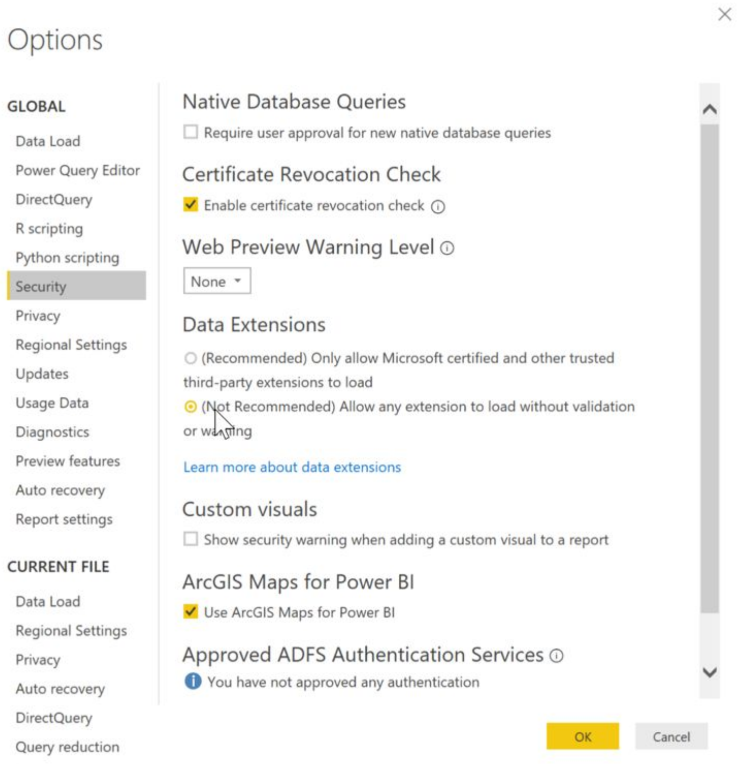
2. Once you have saved the .mez file on your computer, open up Power BI Desktop.

> **User tip |** Microsoft frequently releases new versions of Power BI. We suggest updating your Power BI Desktop to ensure you have the latest functionality and features available to you. The newest Power BI version can be downloaded here.
>
> The flow of steps above may look different in older Power BI versions.

3. In Power BI Desktop, under File, go to "Options and settings". Select "Options".



4. In the Options dialog box, select "Security". Select "(Not recommended) Allow any extension to load without validation or warning" and then click "OK".



Options

GLOBAL

Data Load
Power Query Editor
DirectQuery
R scripting
Python scripting
Security
Privacy
Regional Settings
Updates
Usage Data
Diagnostics
Preview features
Auto recovery
Report settings

CURRENT FILE

Data Load
Regional Settings
Privacy
Auto recovery
DirectQuery
Query reduction

Native Database Queries

□ Require user approval for new native database queries

Certificate Revocation Check

☑ Enable certificate revocation check ⓘ

Web Preview Warning Level ⓘ

None ▾

Data Extensions

○ (Recommended) Only allow Microsoft certified and other trusted third-party extensions to load

● (Not Recommended) Allow any extension to load without validation or warning

Learn more about data extensions

Custom visuals

□ Show security warning when adding a custom visual to a report

ArcGIS Maps for Power BI

☑ Use ArcGIS Maps for Power BI

Approved ADFS Authentication Services ⓘ

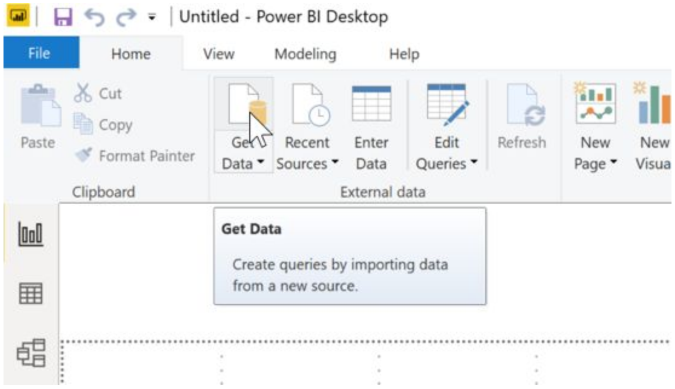
ⓘ You have not approved any authentication

OK Cancel

Get data

Once the connector is installed, you can import your DHIS2 data into Power BI.

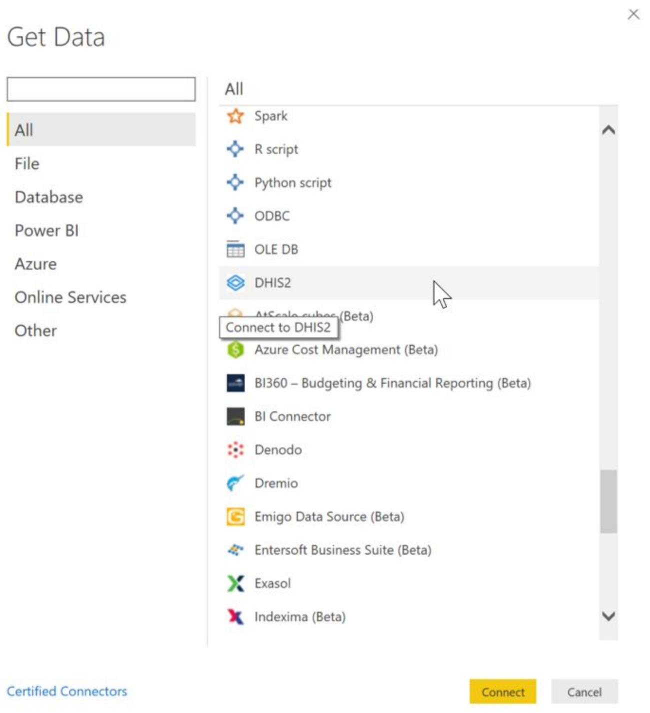
1. In the Home tab, go to “Get Data”



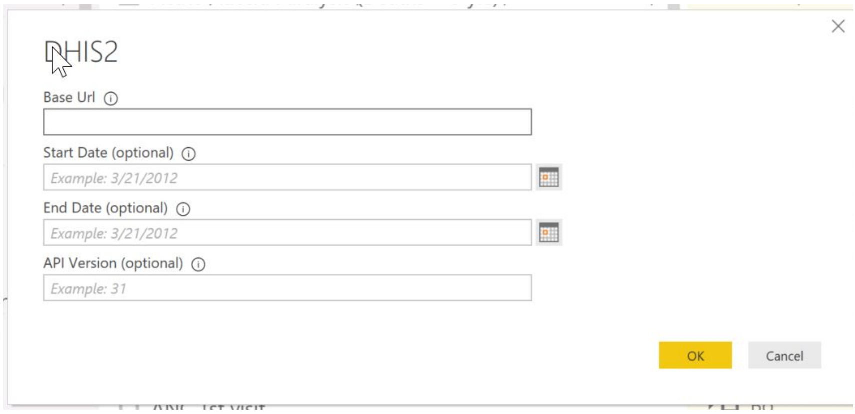
Get Data

Create queries by importing data from a new source.

2. From the list of data sources, select “DHIS2”.



a. Enter the DHIS2 URL you wish to extract data from.



b. Select a start date. The start date should reflect the date the first date for which data are available in DHIS2 that you want to analyze.

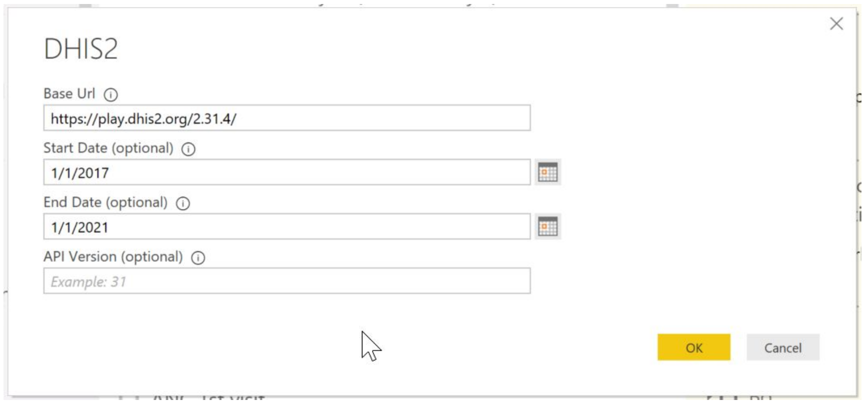
   For example: *If your project began collecting data in DHIS2 on October 1, 2018, then you should select that date.*

   Currently, if no start date is specified, the connector will default to January 1, 2010.

c. Select an end date. The end date should reflect the anticipated end date for your data. If you anticipate refreshing data in the future, a future date should be selected.

***For example:** If your project has an end date of September 31, 2023, then you should use that as your end date (or the date by which all data will be entered into the DHIS2 instance).*

If there is no anticipated end date, or you do not know what the end date is, then select an arbitrary date in the future. ***Currently, if no end date is specified, the connector will default to January 1, 2025.*** It will be possible to change the end date at a later time.



DHIS2

Base Url: https://play.dhis2.org/2.31.4/

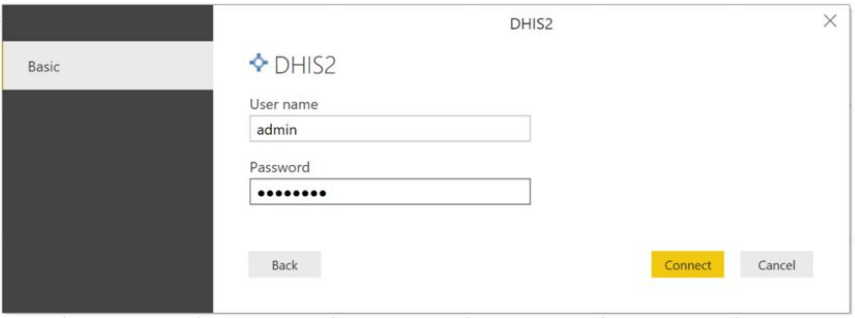
Start Date (optional): 1/1/2017

End Date (optional): 1/1/2021

API Version (optional): Example: 31

5. Enter your DHIS2 credentials. Power BI Desktop will cache credentials for each unique instance of the DHIS2 Url parameter.

   Note that you will only be able to access data for which your DHIS2 user role and permissions allows. The connector currently brings in data based on your Org Unit assignment, not your Data View Org Unit. If no Org Unit is assigned, the root level of the OU hierarchy is used.

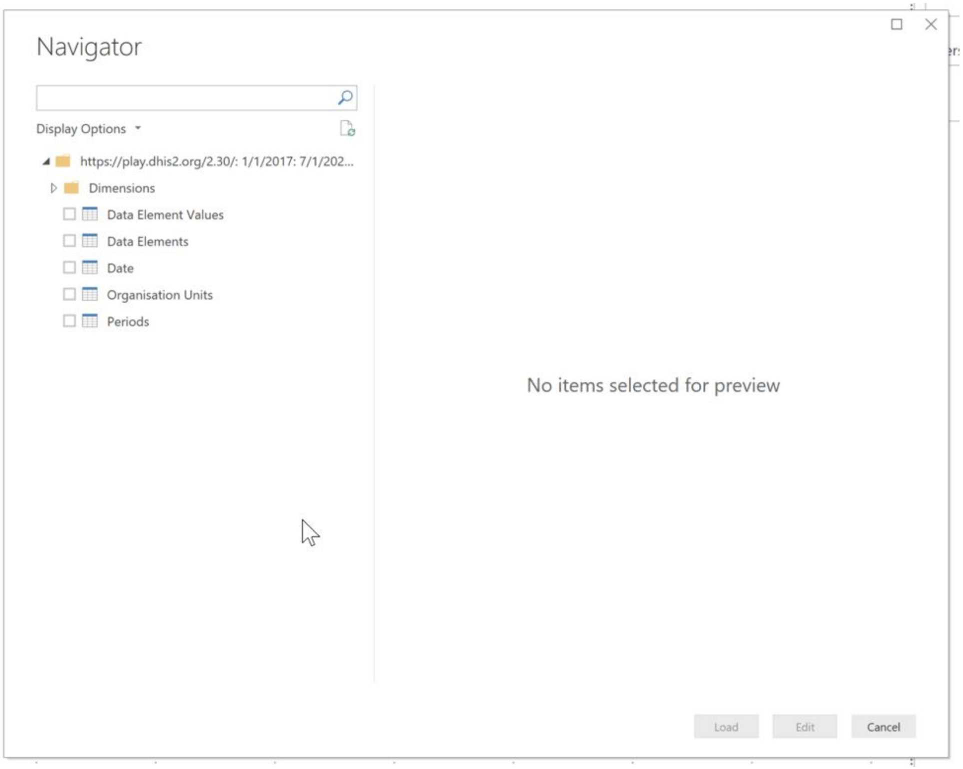


DHIS2

User name: admin

Password: ••••••••

6. Once your credentials are authenticated, a **Navigation box** will pop up, displaying the various DHIS2 tables you may import. You will find the following tables available:
   a. Dimensions - This folder can be expanded to reveal multiple tables representing category option combinations and attribute option combinations.
   b. Data Element Values (the values associated with each individual data element)
   c. Data Elements (all data elements in your DHIS2 instance)
   d. Date
   e. Organisation Units
   f. Periods



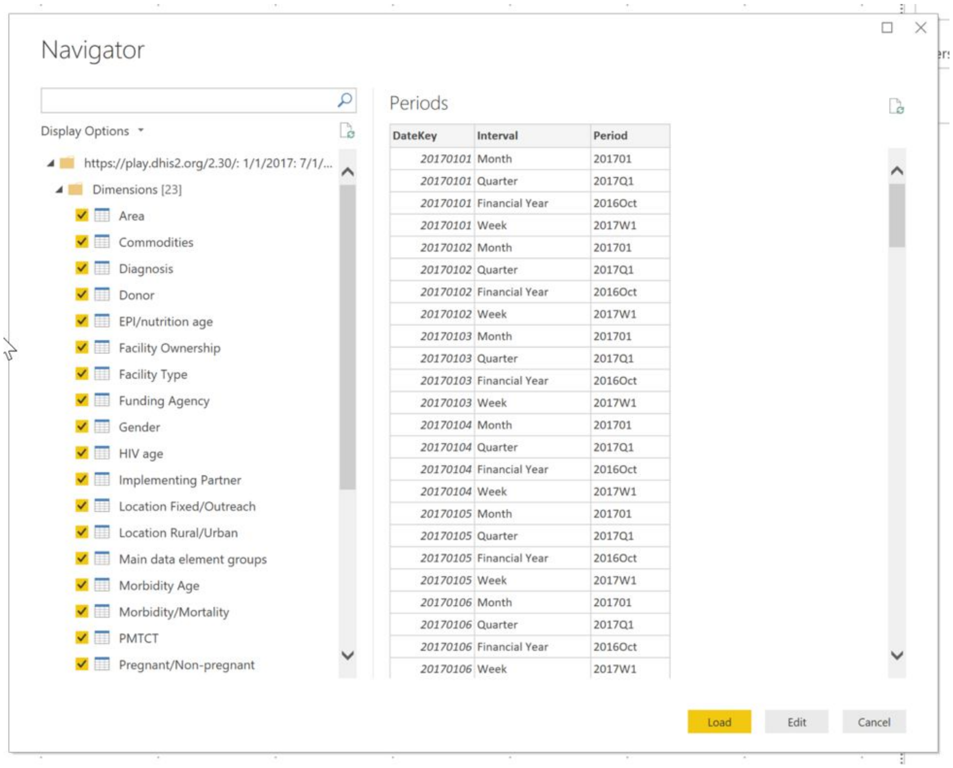
Navigator

Display Options

- https://play.dhis2.org/2.30/: 1/1/2017: 7/1/202...
  - Dimensions
  - □ Data Element Values
  - □ Data Elements
  - □ Date
  - □ Organisation Units
  - □ Periods

No items selected for preview

7. To import data, check all of the tables, and select “Load”.



Navigator

Display Options

- https://play.dhis2.org/2.30/: 1/1/2017: 7/1/...
  - Dimensions [23]
    - ☑ Area
    - ☑ Commodities
    - ☑ Diagnosis
    - ☑ Donor
    - ☑ EPI/nutrition age
    - ☑ Facility Ownership
    - ☑ Facility Type
    - ☑ Funding Agency
    - ☑ Gender
    - ☑ HIV age
    - ☑ Implementing Partner
    - ☑ Location Fixed/Outreach
    - ☑ Location Rural/Urban
    - ☑ Main data element groups
    - ☑ Morbidity Age
    - ☑ Morbidity/Mortality
    - ☑ PMTCT
    - ☑ Pregnant/Non-pregnant

Periods

DateKey	Interval	Period
20170101	Month	201701
20170101	Quarter	2017Q1
20170101	Financial Year	2016Oct
20170101	Week	2017W1
20170102	Month	201701
20170102	Quarter	2017Q1
20170102	Financial Year	2016Oct
20170102	Week	2017W1
20170103	Month	201701
20170103	Quarter	2017Q1
20170103	Financial Year	2016Oct
20170103	Week	2017W1
20170104	Month	201701
20170104	Quarter	2017Q1
20170104	Financial Year	2016Oct
20170104	Week	2017W1
20170105	Month	201701
20170105	Quarter	2017Q1
20170105	Financial Year	2016Oct
20170105	Week	2017W1
20170106	Month	201701
20170106	Quarter	2017Q1
20170106	Financial Year	2016Oct
20170106	Week	2017W1

Loading time will vary depending on the amount of data being extracted from DHIS2 and imported into Power BI and may take up to 20 minutes. Power BI data loading dialog will update occasionally, to display the total records fetched.

User tip | If you need to shape, transform, filter, or clean your data, you can proceed directly to the Query Editor (see below) by selecting "Edit" at this step instead of "Load". You can enter the Query Editor at any time.

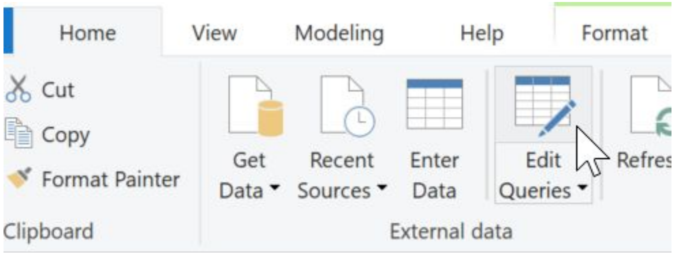
Data modelling

Data modelling is a process that defines the data elements, their attributes, and their relationships. Once you select your DHIS2 data for import into Power BI, you may need to perform a few basic steps to optimize the data model.

While the following data modelling section is out of scope of the connector, recommendations for improving your data model are provided below.

Shape data using the Query Editor

After the DHIS2 connector import operation has been completed, you may want to apply additional operations to further shape, transform, filter, or clean your data. These operations only affect data in Power BI, and data in your DHIS2 instance will be unchanged. You can use the Query Editor at any time, by selecting "Query Editor" from the Home tab.



Some examples include:

- Correcting spelling mistakes in metadata
- Removing unused columns (e.g., Org Unit Code)
- Filtering out unused levels of Org Units
- Adding additional date intervals to the Date and Periods tables
- Updating the Base Url
- Updating the start and end dates
- Adding previously omitted tables

Deleting and re-importing tables will undo relationships, data type assignments, hidden fields, etc. You can use the Options dialog to help avoid destroying the relationships, but you will still need to redo any hidden fields, decimal precision, etc.

If you later realize you have brought in a table (e.g., dynamic Dimension) which is unused, you should delete if from the model to reduce file size, memory usage, complexity, and load time.

Manage data relationships

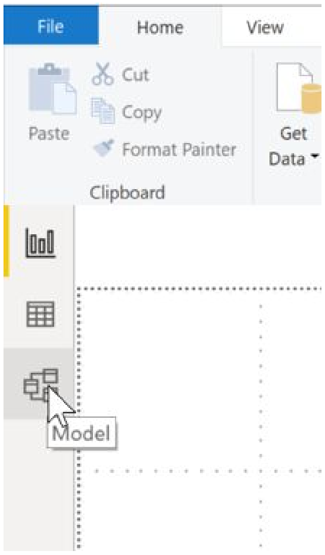
Power BI will attempt to automatically detect relationships between data tables or elements where possible, however these relationships need to be modified to achieve a more optimal data model. Setting accurate relationships between your data allows you to create complex calculations across multiple data elements.

User tip | A star schema data model comprises a single fact table with direct relationships to multiple dimension tables. A snowflake schema is similar to a star schema, however has multiple joins. A star or snowflake schema enables the greatest flexibility and performance in Power BI implementation. Simply put, Power BI will work faster and give the user the most options for slicing data using this model.

In your model, the Data Element Values table is the fact table, and all other tables will have a relationship to this table, either directly, or indirectly.

Managing relationships in Power BI is simple.

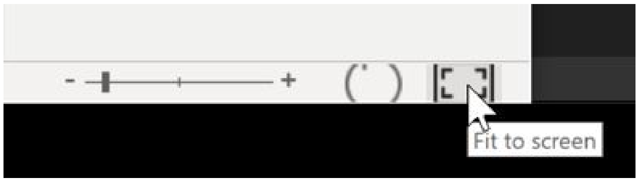
1. To see the relationships between your data tables, use the **Relationship view**, found on the far left side of the screen next to the Report canvas.



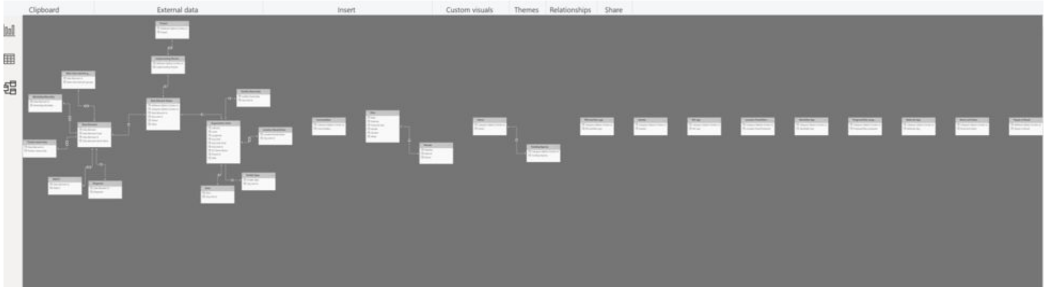
2. Each block represents a data table and its columns, and the lines between them represents relationships between columns. As stated above, Power BI will automatically detect some

relationships between tables; however, some of these relationships should be edited to improve your data model.

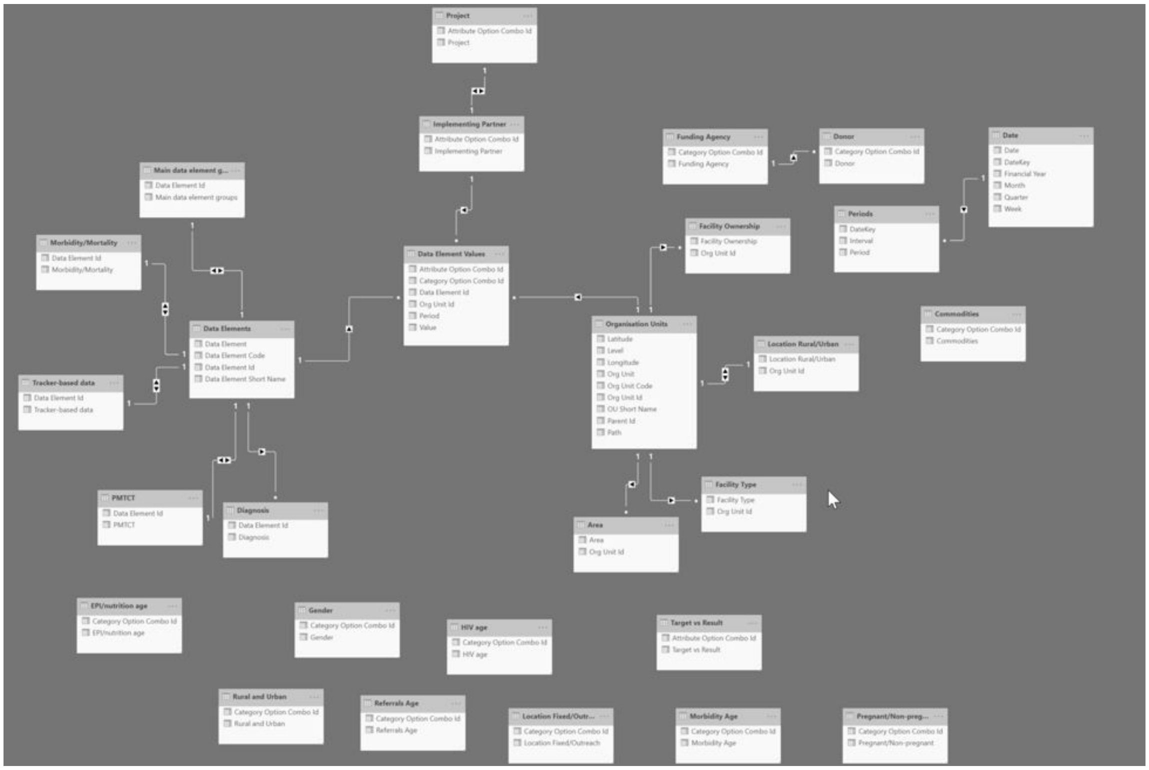
a. To fit all of your tables on the screen, select "Fit to screen" (as there may be multiple tables that you cannot see when you initially land on the Relationship view).



b. Once you are able to see all of the tables you imported from DHIS2, you may begin to organize and edit their relationships.



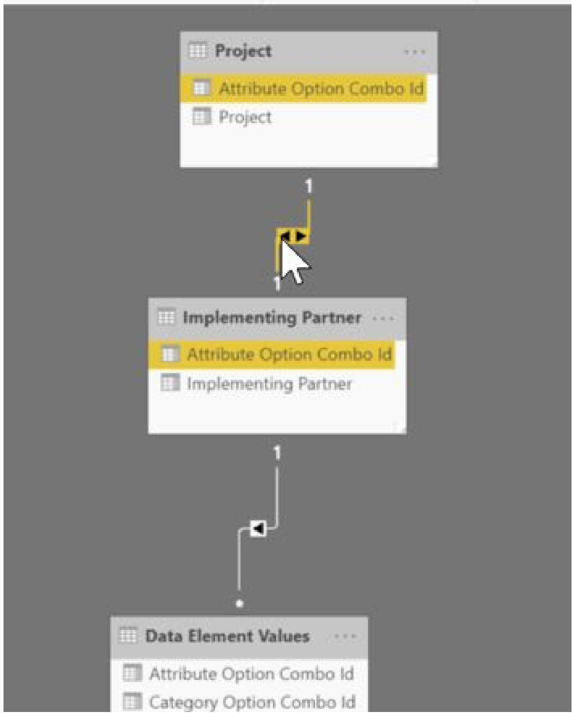
c. One suggestion for organizing your tables is to spatially group them so that: tables of attribute option combinations are oriented above the Data Elements Value table; tables of category option combinations are moved to below the Data Elements Value table. To move a table, simply select and drag it to the new location.



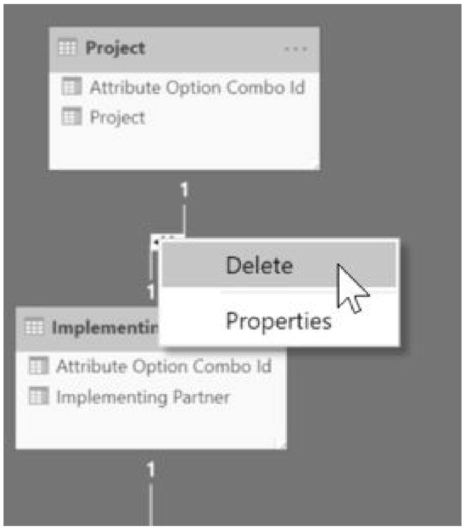
3. To remove a relationship, right-click on it and select 'Delete'. Relationships can be deleted without loss of data. You may encounter cases where Power BI detects a relationship between two or more dimension tables that is not ideal to maintain.

   For example: *In the play.dhis2 instance, Power BI automatically detected a relationship between two attribute option combinations - the tables "Project" and "Implementing Partner" because they have the column 'Attribute option combo Id' in common. Since the Partner table is not dependent on the Implementing Partner table, the data model would be improved by deleting this relationship.*

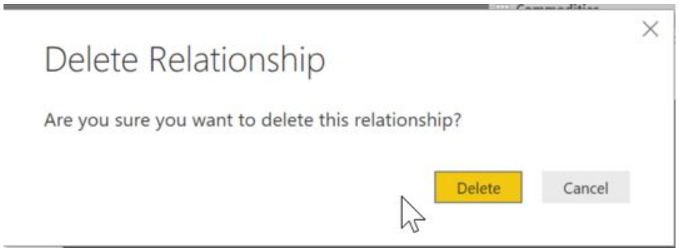
   a. Right click "◂▸"



b. Select “Delete”.

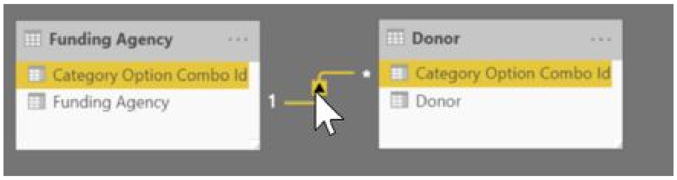


c. Confirm “Delete”.

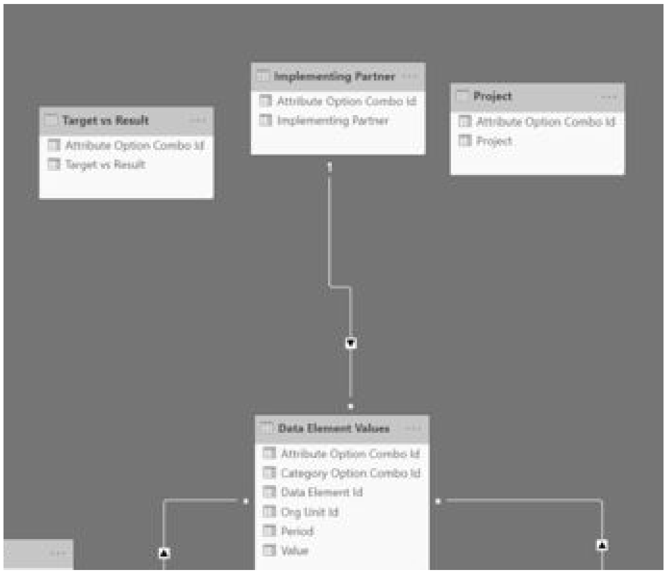

Delete Relationship

Are you sure you want to delete this relationship?

d. Delete any other relationships that you do not want, such as relationships Power BI detected between category option combination tables.



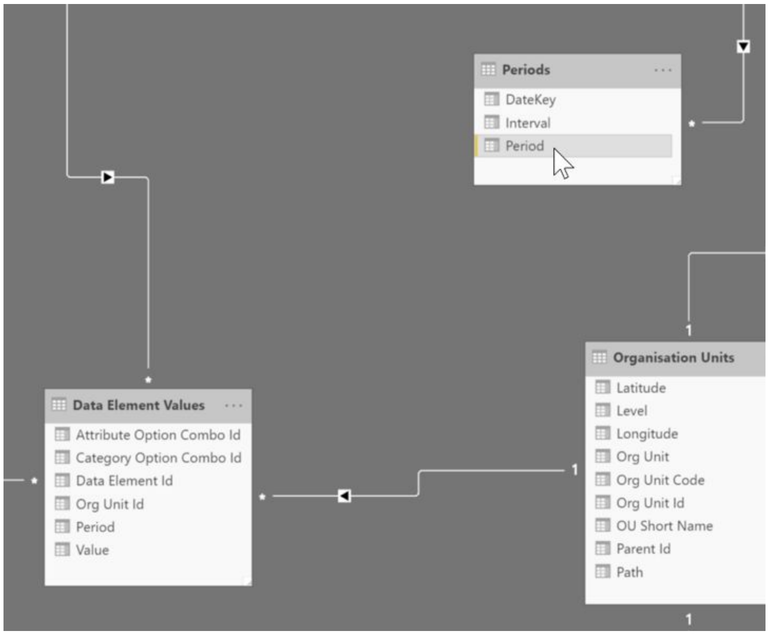
e. You can see below how the deletion(s) changed the data model:



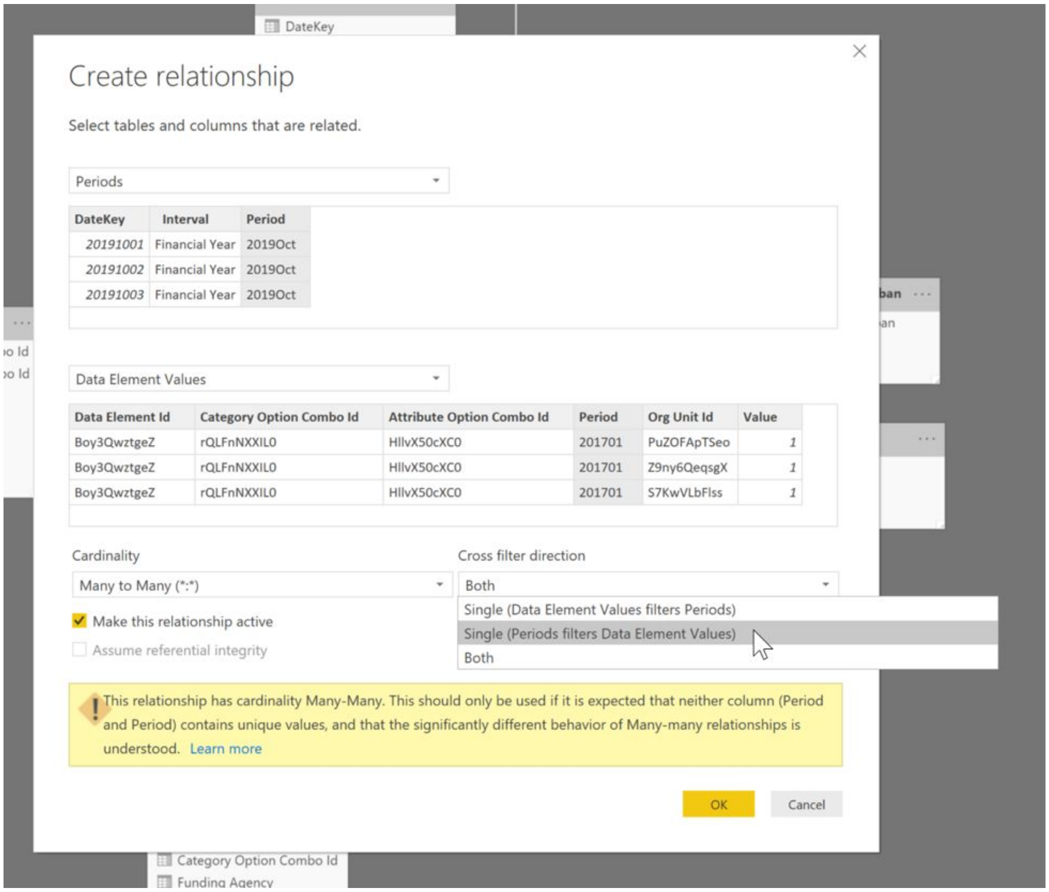
4. To create a relationship, drag and drop the fields that you want to link between tables. Creating a relationship between tables will enable you to filter one table by the other. You will likely need to create new relationships in your data model

   For example: *In the play.dhis2 instance, Power BI automatically detected a relationship between the Periods and Date table; however no relationship was automatically detected between the Periods and Data Element Values tables. You will likely encounter the same issue with your DHIS2 data, and will need to create this relationship.*

   a. Select the “Period” column in the Periods table, and drag it to the “Period” column in the Data Elements Value table.



b. A dialog box will appear, from which you can specify the nature of the relationship between the two tables.



Create relationship

Select tables and columns that are related.

Periods

DateKey	Interval	Period
20191001	Financial Year	2019Oct
20191002	Financial Year	2019Oct
20191003	Financial Year	2019Oct

Data Element Values

Data Element Id	Category Option Combo Id	Attribute Option Combo Id	Period	Org Unit Id	Value
Boy3QwztgeZ	rQLFnNXXIL0	HllvX50cXC0	201701	PuZOFApTSeo	*1*
Boy3QwztgeZ	rQLFnNXXIL0	HllvX50cXC0	201701	Z9ny6QeqsgX	*1*
Boy3QwztgeZ	rQLFnNXXIL0	HllvX50cXC0	201701	S7KwVLbFlss	*1*

Cardinality: Many to Many (*:*)

Cross filter direction: Both

- Single (Data Element Values filters Periods)
- Single (Periods filters Data Element Values)
- Both

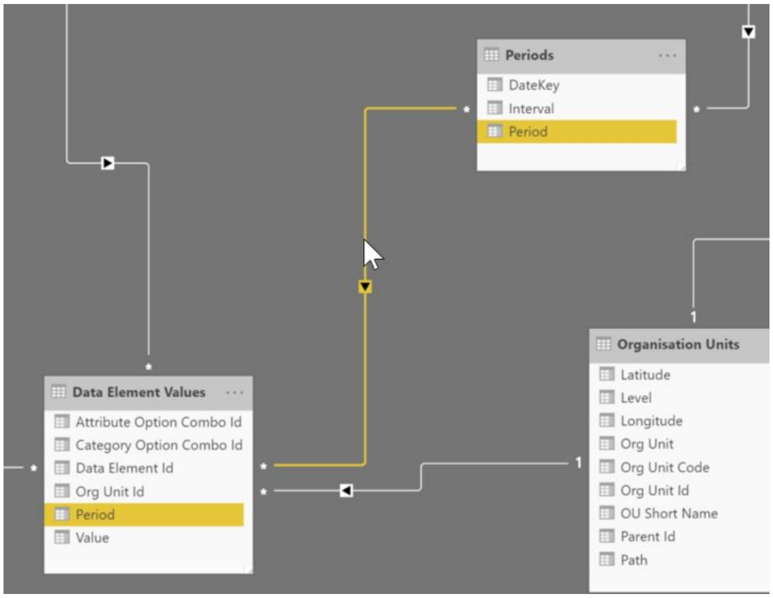
☑ Make this relationship active

□ Assume referential integrity

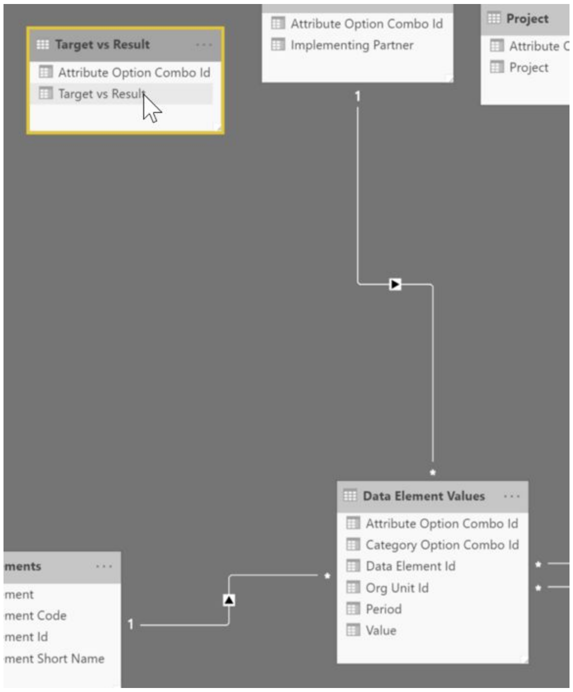
This relationship has cardinality Many-Many. This should only be used if it is expected that neither column (Period and Period) contains unique values, and that the significantly different behavior of Many-many relationships is understood. Learn more

As shown above, you will see that Power BI calls out the table names (“Periods” and “Data Element Values”) and highlights the column by which the two tables are related (“Period”). You may also specify the **cardinality** of the relationship (in this case, select“Many to Many”) and the **cross filter direction** (in this case, select “Single (Periods filters Data Element Values)”).

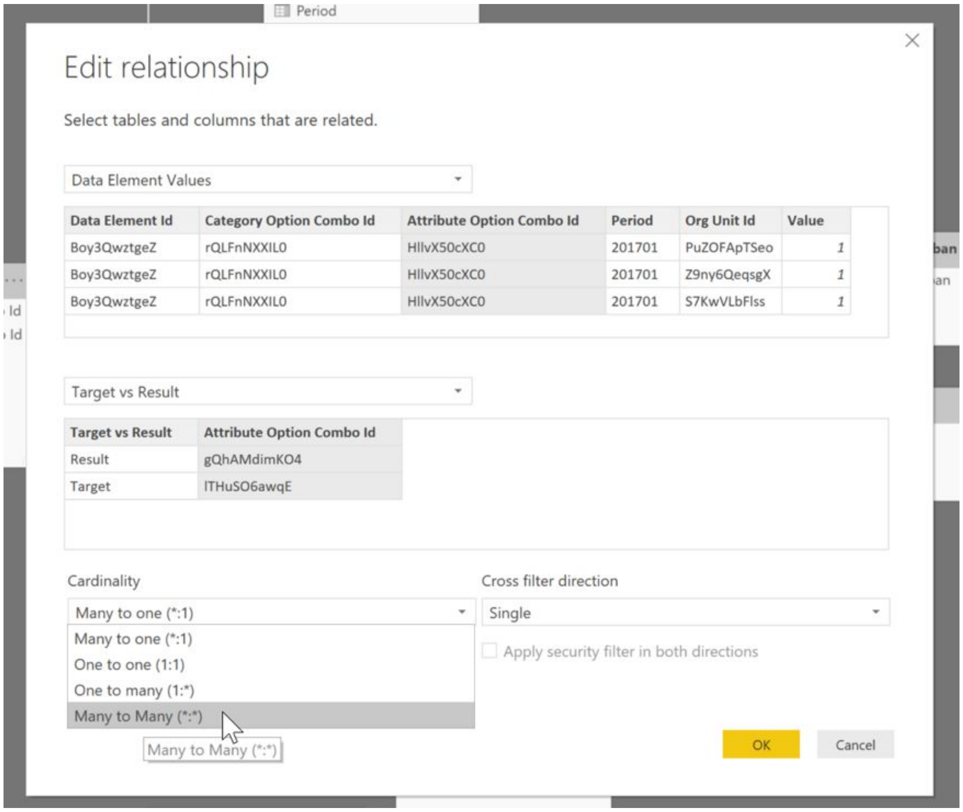
c. Select “OK”. Once you have completed this step, you will see your newly created relationship, shown below highlighted in yellow.



d. Another example would be to create a relationship between the “Target vs Result” attribute option combination table and the “Data Element Values” table. First, select the “Attribute Option Combo Id” in the Target vs Result table, and drag it to “Attribute Option Combo Id” in the Data Element Values table.



Edit the cardinality of the relationship to "Many to Many" and the cross filter direction to "Single (Attribute Option Combo Id filters Data Element Values)".



Edit relationship

Select tables and columns that are related.

Data Element Values

Data Element Id	Category Option Combo Id	Attribute Option Combo Id	Period	Org Unit Id	Value
Boy3QwztgeZ	rQLFnNXXIL0	HllvX50cXC0	201701	PuZOFApTSeo	1
Boy3QwztgeZ	rQLFnNXXIL0	HllvX50cXC0	201701	Z9ny6QeqsgX	1
Boy3QwztgeZ	rQLFnNXXIL0	HllvX50cXC0	201701	S7KwVLbFlss	1

Target vs Result

Target vs Result	Attribute Option Combo Id
Result	gQhAMdimKO4
Target	ITHuSO6awqE

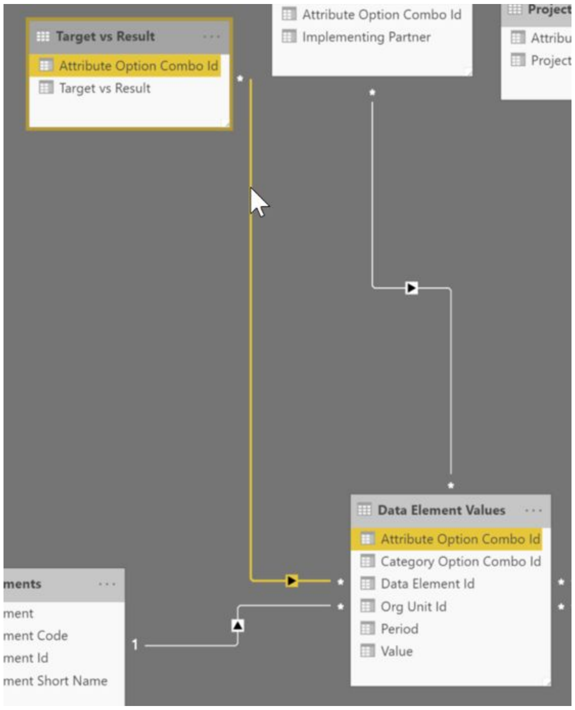
Cardinality: Many to one (*:1)

- Many to one (*:1)
- One to one (1:1)
- One to many (1:*)
- Many to Many (*:*)

Cross filter direction: Single

□ Apply security filter in both directions

View your new relationship.

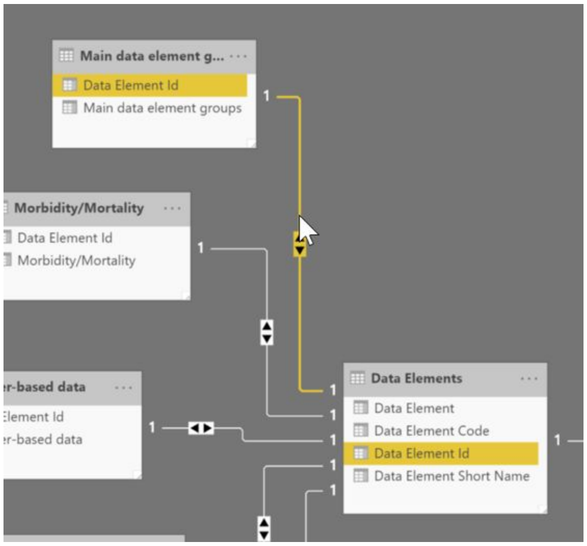


5. Edit your relationship cardinality and cross filter direction. In your final data model, the majority of your data tables will have Many to Many relationships to enhance flexibility in data analysis. Relationships that you will want to make Many to Many include those between the Data Elements Value table (your fact table) and all attribute option combination tables and category option combination tables.

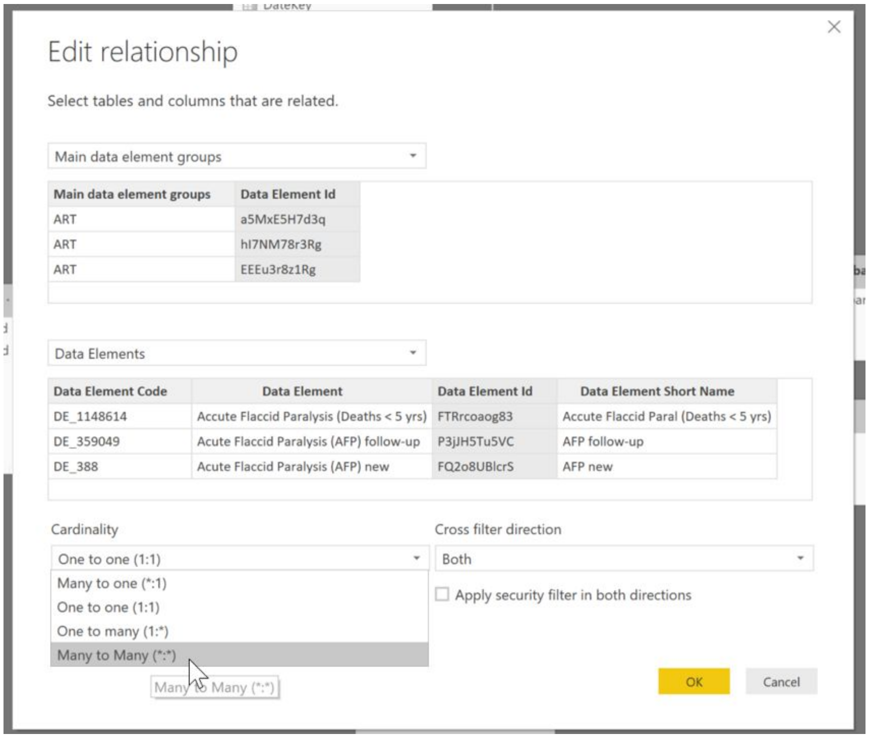
   Additionally, you should change the cardinality of the relationships between the Data Elements table and all data element group set tables as well as between organisational Units and its related tables.

   You can also edit the cross filter direction. In most cases, the cross filter direction should be set to "single" so that the outer table filters the inner table.

   a. Double click the relationship between one of your data element group set tables (name of the actual tables will vary, however in play.dhis2, one table is called "Main data element groups") and the Data Element table.



b. In the dialog box, select “Many to Many”.



Edit relationship

Select tables and columns that are related.

Main data element groups

Main data element groups	Data Element Id
ART	a5MxE5H7d3q
ART	hI7NM78r3Rg
ART	EEEu3r8z1Rg

Data Elements

Data Element Code	Data Element	Data Element Id	Data Element Short Name
DE_1148614	Accute Flaccid Paralysis (Deaths < 5 yrs)	FTRrcoaog83	Accute Flaccid Paral (Deaths < 5 yrs)
DE_359049	Acute Flaccid Paralysis (AFP) follow-up	P3jJH5Tu5VC	AFP follow-up
DE_388	Acute Flaccid Paralysis (AFP) new	FQ2o8UBlcrS	AFP new

Cardinality

One to one (1:1)

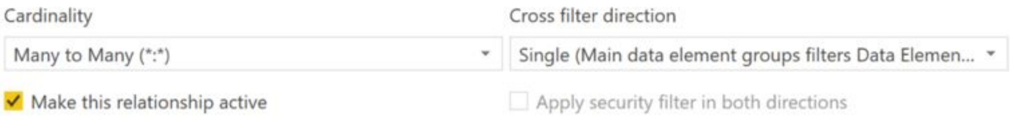
- Many to one (*:1)
- One to one (1:1)
- One to many (1:*)
- Many to Many (*:*)

Cross filter direction

Both

☐ Apply security filter in both directions

c. Ensure that the cross filter direction is set to “Single (Main data element group filters Data Elements)”. Also be sure that the relationship is checked to “active”.



Cardinality

Many to Many (*:*)

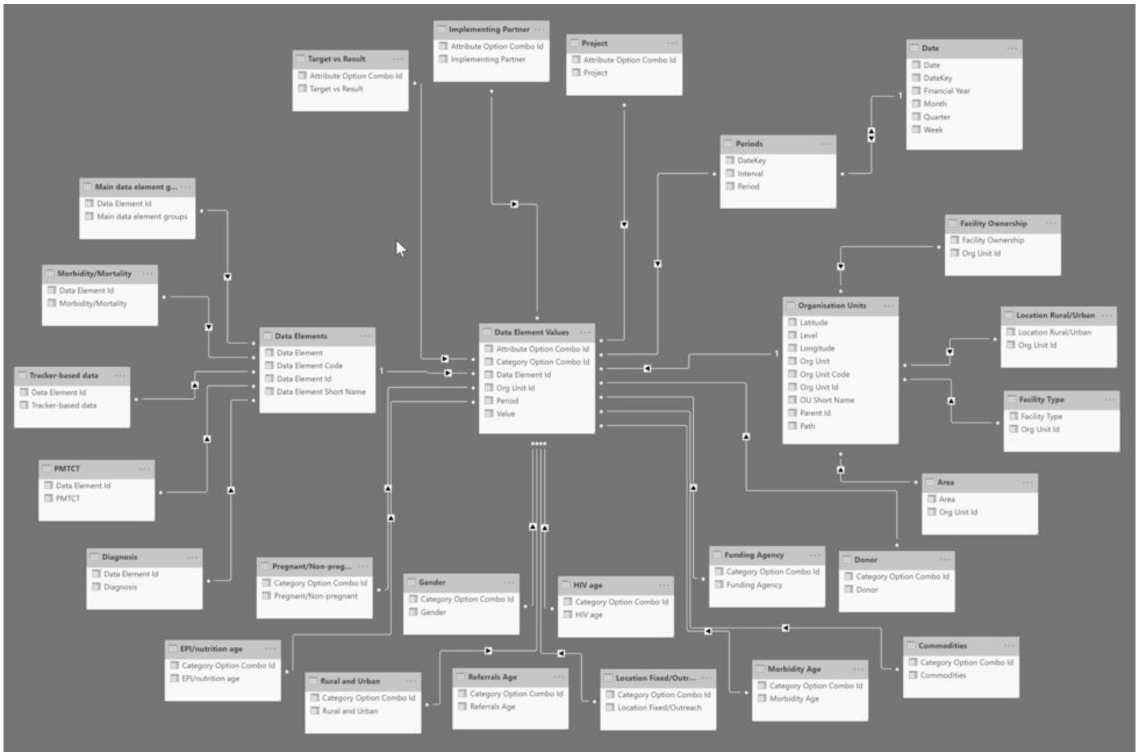
Cross filter direction

Single (Main data element groups filters Data Elemen...

☑ Make this relationship active

☐ Apply security filter in both directions

d. Repeat this process as necessary, so that your resulting data model looks similar to below, whereby:
   i. All attribute option combination tables and category option combination tables have a Many to Many relationship with the Data Element Value table (marked by a "*" on both ends of the relationship join).
   ii. The Data Elements table and Organisation Units tables have a One to Many relationship (marked by a "1" near to the dimension tables and "*" near to the Data Element Values table).
   iii. All other 'outer' tables (e.g. data element group sets) have a Many to Many relationship with the Data Elements and/or Organisational Units tables.
   iv. In general, all cross filter directions (▸) should be pointed inward, so that there is a single direction with the 'outer' table is filtering the 'inner' table (e.g. the Target vs Result table filters the Data Element Values table).



e. There are cases where you may want to maintain bi-directional filtering.

   For example: *You may want to maintain bi-directional filtering in the relationship between the "Date" and "Periods" table. In this case, instead of selecting "single" under Cross Filter Direction, you should select "Both" (date table to period table).*

f. Experiment with this setting to see its effect. Ultimately, you will need to build a data model that responds according to your analysis needs, so getting input from report consumers (e.g. those that will using and interpreting the reports) is essential.

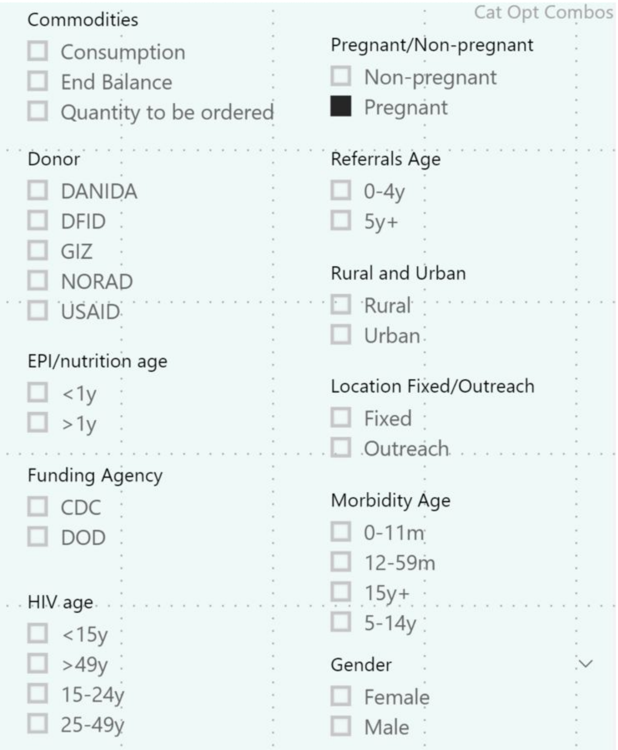
   For example: *The images below demonstrate how the cross filtering functionality behaves differently when set to "single" versus "both".*

In Case 1 (below, left) all of the Category Option Combination dimensions were set to filter the Data Element Values (in a "single" direction). Note here that by slicing (filtering) the data by "pregnant", all of the other dimensions from which you could additionally filter data with are still visible in the report. Thus, selecting an option in one slicer does not affect the visibility of others.

Conversely, in Case 2 (below, right), the relationship between all of the Category Option Combination dimensions and the Data Element VAlues table were set to filter in "both" directions. As a result, when slicing (filtering) the data by "pregnant", only dimensions for which "pregnant" is an option are visible in the report.

Either choice (single versus both cross filter directionality) will not affect the resulting data values. Oftentimes, report consumers wills till want to see the full org unit hierarchy regardless of how data are being sliced. To achieve this, the cross filter direction between the OrgUnit table and the Data Element Values table should be set to "single (OrgUnit filters Data Element Values)".

Case 1
Single (Category Option Combos filter Data Element Values)



Cat Opt Combos

Commodities
☐ Consumption
☐ End Balance
☐ Quantity to be ordered

Donor
☐ DANIDA
☐ DFID
☐ GIZ
☐ NORAD
☐ USAID

EPI/nutrition age
☐ <1y
☐ >1y

Funding Agency
☐ CDC
☐ DOD

HIV age
☐ <15y
☐ >49y
☐ 15-24y
☐ 25-49y

Pregnant/Non-pregnant
☐ Non-pregnant
■ Pregnant

Referrals Age
☐ 0-4y
☐ 5y+

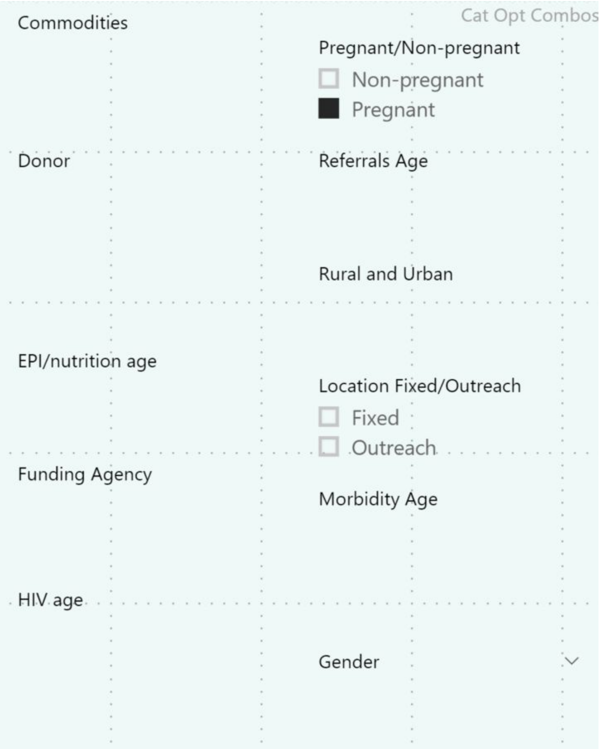
Rural and Urban
☐ Rural
☐ Urban

Location Fixed/Outreach
☐ Fixed
☐ Outreach

Morbidity Age
☐ 0-11m
☐ 12-59m
☐ 15y+
☐ 5-14y

Gender
☐ Female
☐ Male

Case 2
Both



Cat Opt Combos

Commodities

Donor

EPI/nutrition age

Funding Agency

HIV age

Pregnant/Non-pregnant
☐ Non-pregnant
■ Pregnant

Referrals Age

Rural and Urban

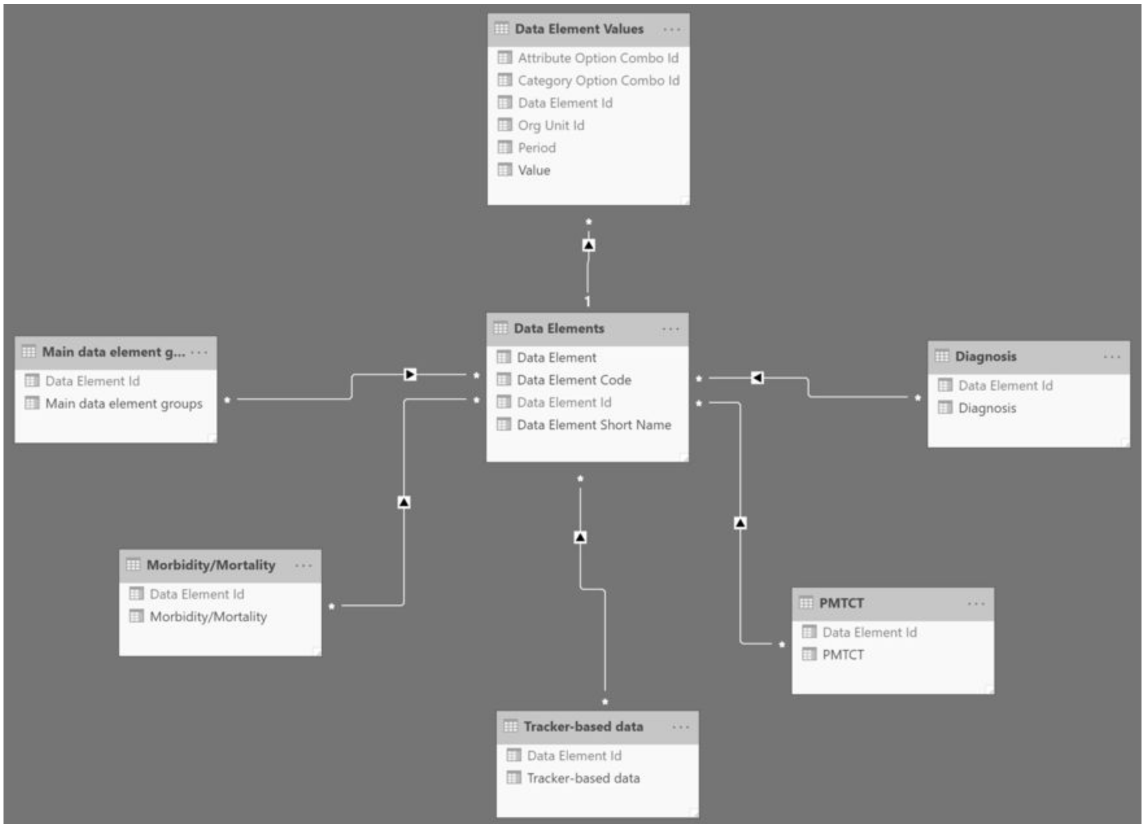
Location Fixed/Outreach
☐ Fixed
☐ Outreach

Morbidity Age

Gender

6. Additional tabs at the bottom of the Relationship window can be used to create subviews for easier relationship maintenance.

   For example: *The below image shows a subview of the Data Elements table and its relationship to the fact table (Data Element Values table) and other dimension tables (e.g. data group set tables).*



7. To test your relationships and ensure that your data model is behaving as expected to meet your analysis needs, you can create a number of slicers and lists in the **Report Builder**. This will enable you to see how slicing by one dimension will affect the other dimensions.

   For example: *These recommendations have guided you to create a snowflake schema data model. You might instead decide to create a pure star schema, in which all dimension tables have direct relationships to the fact (Data Element Values) table. You can test building a direct relationship between a data group set table with the Data Element Values table, and see how the cross filtering function behaves differently compared to before.*

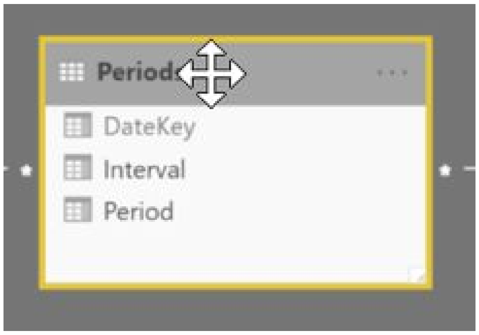
Finalizing the model

A clean, organized model is critical in enabling report developers to more easily create visuals for data analysis … yada yada

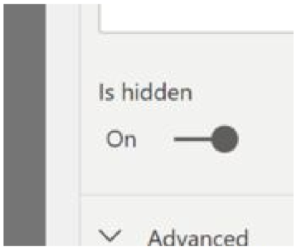
1. Hide tables. You may opt to hide a table so that the user responsible for building reports does not see the corresponding data table.

   For example: *You may not need to use the "Periods" table in your analysis. In this case, it would be ideal to hide any tables that will not be used during analysis. Hiding tables will not delete the table, or its relationships from the data model. You can always unhide a hidden table in your data model at a later time if you decide that you want to include those data elements in your analysis.*

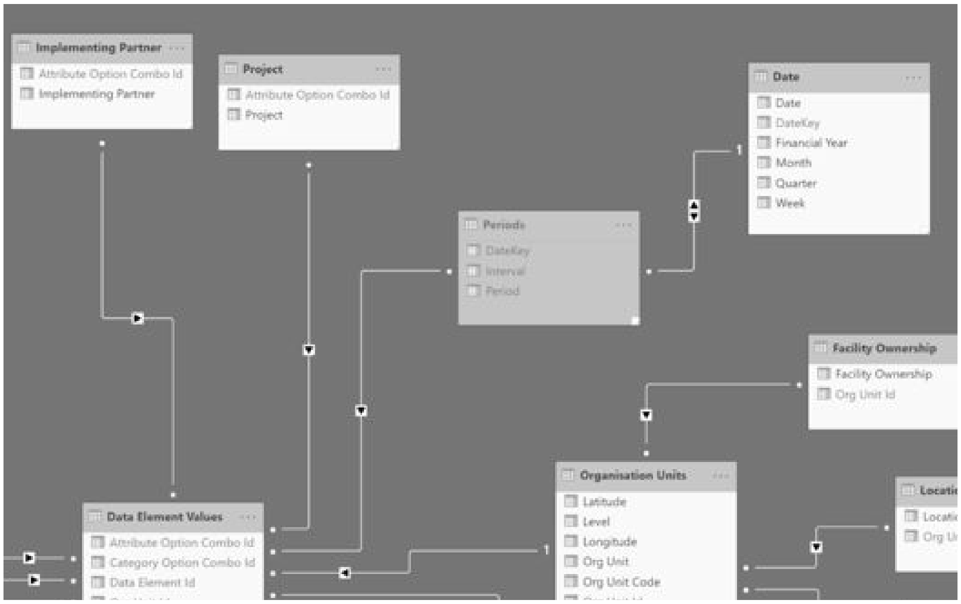
a. Select the table you wish to hide, in this case, the “Periods” table and click on the ellipses (“. . .”) at the top right corner.



b. Scroll down the options and slide **Is hidden** from “Off” to “On”.


Is hidden
On

c. Once completed, you will see in your data model view that the hidden table is greyed.



d. You can test this in the **Report Builder** view. The “Periods” table should not be seen in the Variables list.

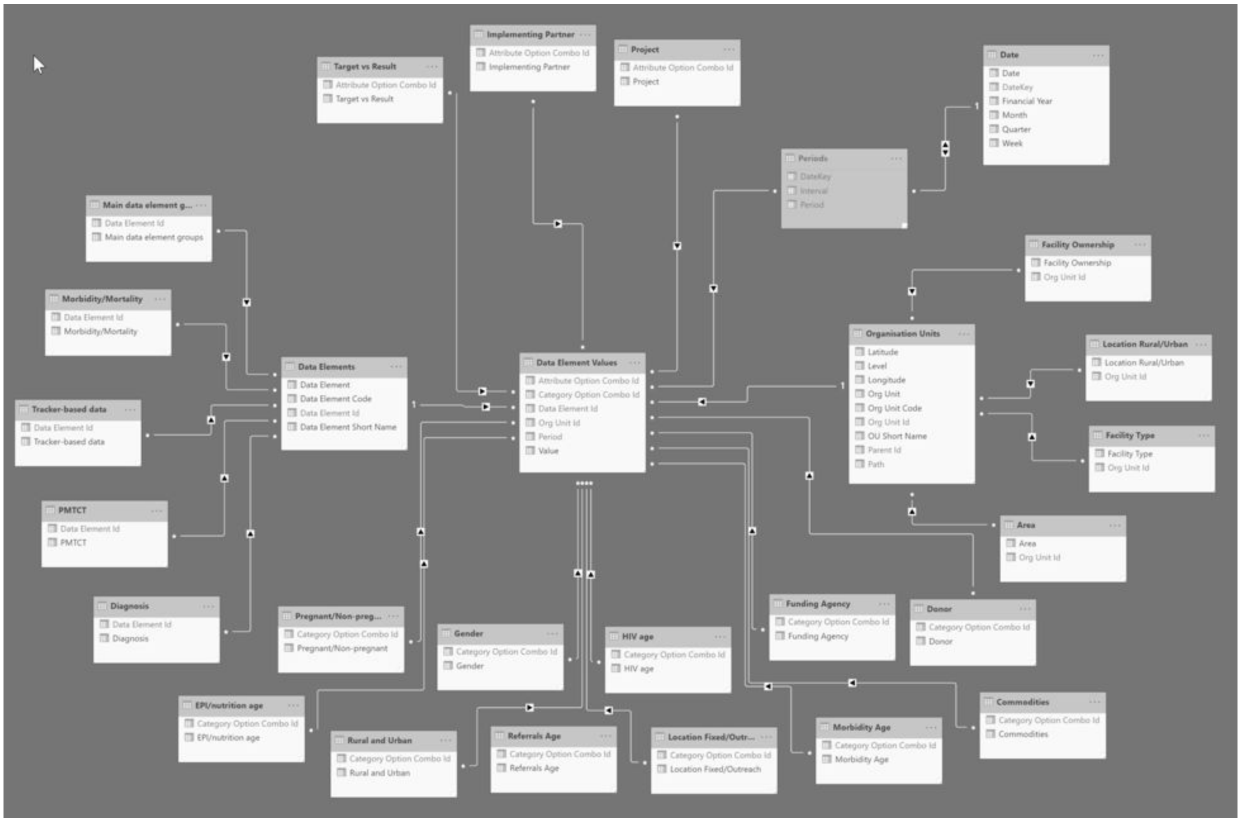
2. Hide the Key (Id) columns. ID columns are not used for analysis, therefore it makes sense to hide these columns from the report builder to minimize extraneous variables. Hide

anything that you do not want the end user on the front end to see as an available field during analysis

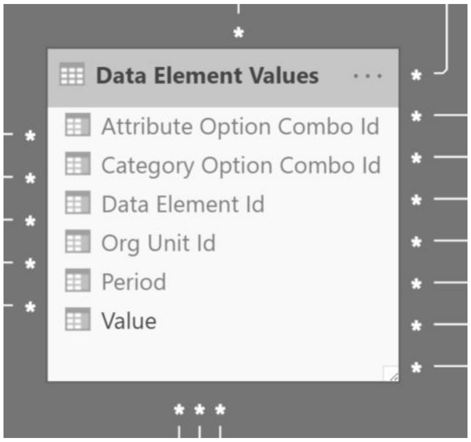
a. Select one Id column and control click to select all the other Id columns, “DateKey” in the Periods and Date tables, and “Period” in the Data Element Values table.

b. Other examples might include OU level, OU code.

c. Hidden columns text will appear a lighter grey.



If you were to zoom in to the Data Element Values tables, you would see that all of the columns in the table are greyed out (e.g. hidden) except for the “Value” column, which just contains the raw data values.


Data Element Values

- Attribute Option Combo Id
- Category Option Combo Id
- Data Element Id
- Org Unit Id
- Period
- Value

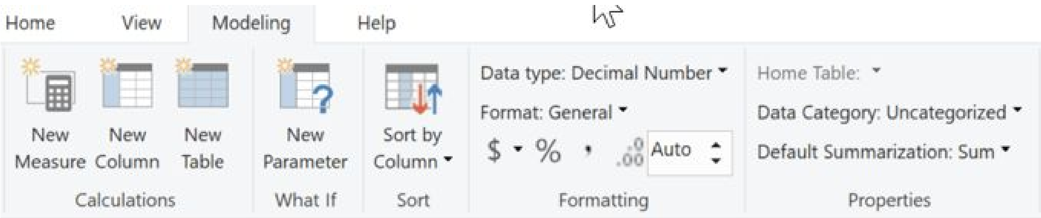
d. You can test this in the **Report Builder** view. None of the hidden columns or tables should show up. test in report view - none show up).

3. Power BI assumes numeric columns should always be aggregatable, though in some cases this doesn't make sense. In these cases, you will need to disable aggregation for those columns.

   a. In the Report Builder view, scroll down your list of tables in the **Fields pane**, and click on the column where it does not make sense to summarize values for.

   For example: *OrgUnit Level.*

   b. Go to the Modelling tab.



Data type: Decimal Number

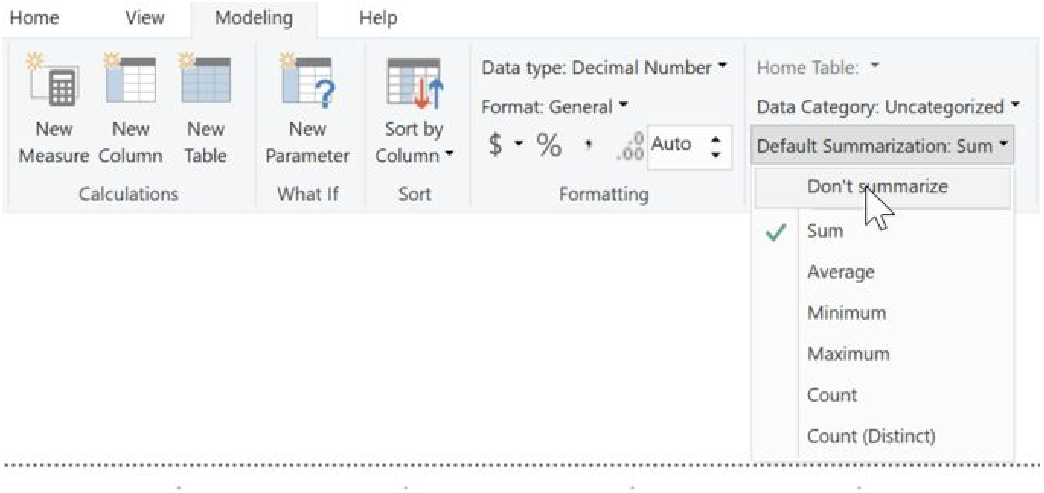
Format: General

Home Table:

Data Category: Uncategorized

Default Summarization: Sum

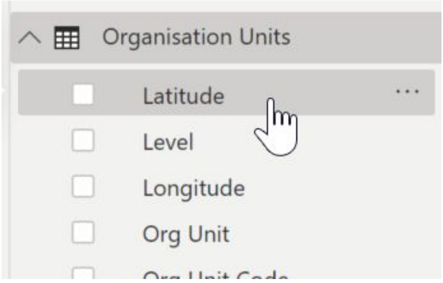
   c. Under "Default Summarization", select "Don't summarize".



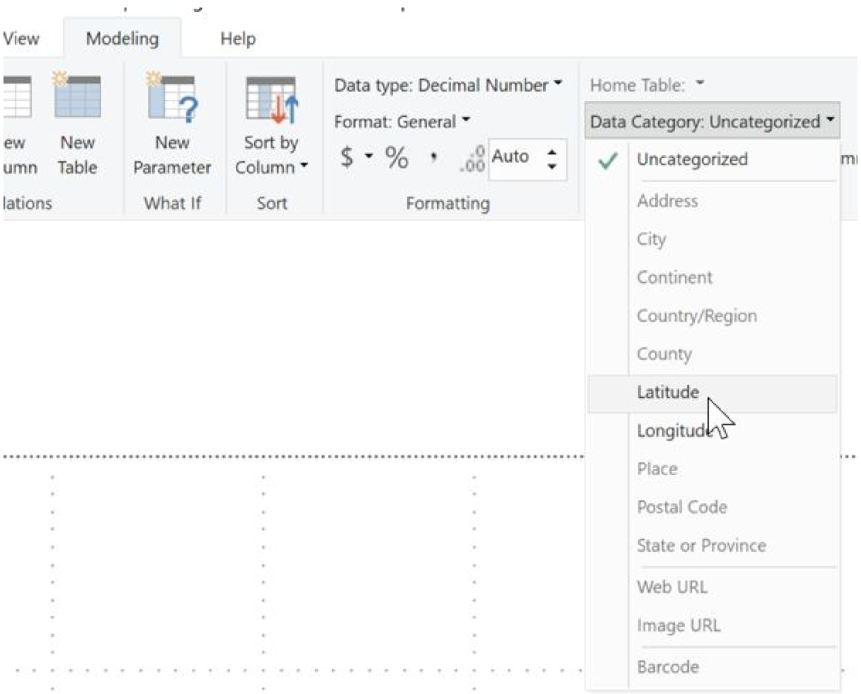
4. Power BI will work optimally if you categorize your data, to the extent possible.

   For example: *If your data set contains latitude and longitude coordinates, the map features in Power BI will be optimized if you categorize your variable fields as such.*

   a. In the Report Builder View, under your list of Fields, select your latitude column.

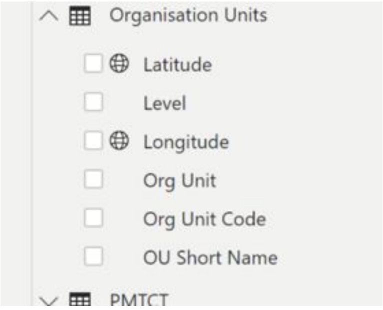


   b. In the Modelling tab, under Data Category, select “Latitude”.



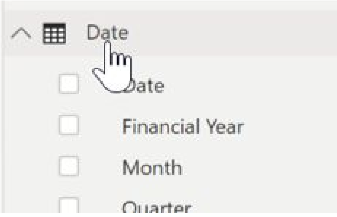
c. Repeat for your Longitude column.

d. In your Fields list, you will see a new icon in front of Latitude and Longitude column names.

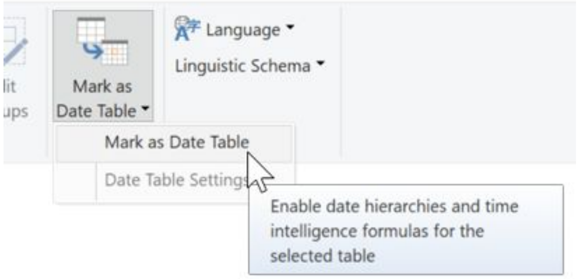


5. Mark your Date table. To support Time Intelligence functions, by default, Power BI automatically creates hidden date tables to support calculations, date hierarchies, and quick measures. By explicitly appointing the date table, we optimize the model and ensure calculations have a contiguous, inclusive set of dates to work with.

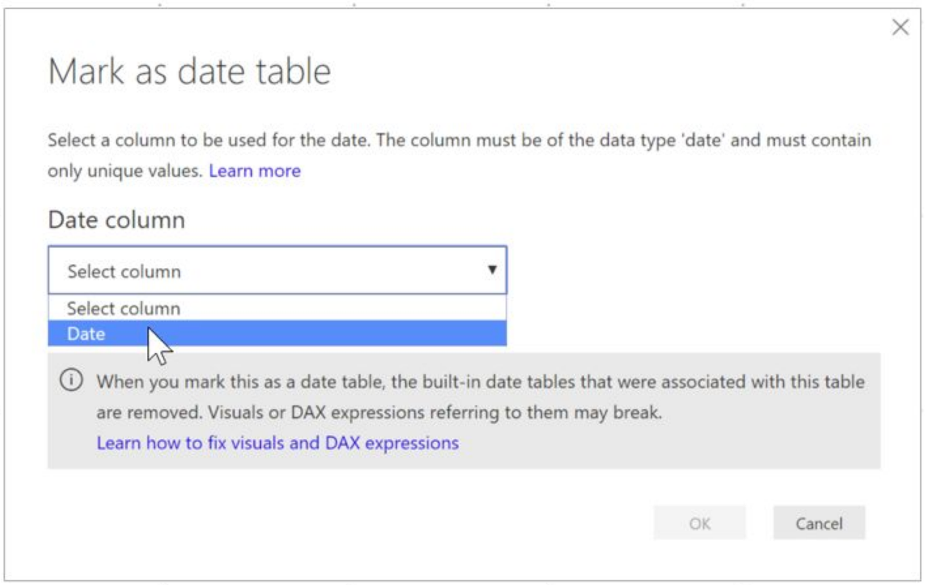
    a. Select your Date table header.



b. Under the Modelling tab, select “Mark as Date Table”.



c. The dialog box will ask you which column in the Date table you would like to mark as a date. Select the “Date” column.



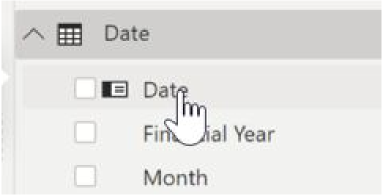
Mark as date table

Select a column to be used for the date. The column must be of the data type 'date' and must contain only unique values. Learn more

Date column

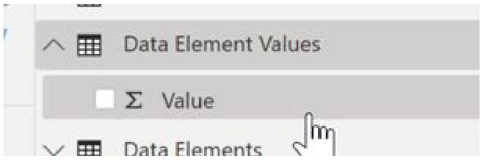
When you mark this as a date table, the built-in date tables that were associated with this table are removed. Visuals or DAX expressions referring to them may break.
Learn how to fix visuals and DAX expressions

d. Under the Fields list, you will see a new icon in front of you Date column, indicating that it is marked as such.

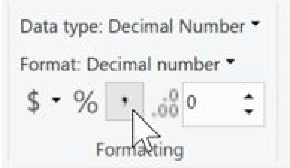


6. Configure the default display format for number columns. Configuring your number columns at this early stage will save you a lot of time later on when you are actually building reports. Depending on what your preference is for number formatting (e.g. which digits you want displayed on your visuals), you can set the default now, then latter change any exceptions to the rule at the time of report building.

a. Under your Fields list, select the “Value” column in your Data Element Values table.



b. In the Modelling tab, configure your Data type and Format. You can choose the number of places after the decimal point as well as select “,” to separate thousands in your number (e.g. 1,000 versus 1000.00).

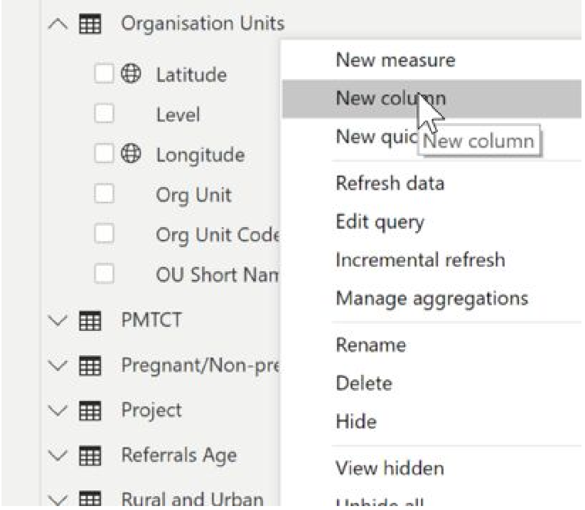


Create org unit hierarchies

Power BI will not automatically configure your organisational unit hierarchies as they are built in DHIS2. In order to recreate your hierarchy, you will need to perform some basic DAX functions. This step is essential to enabling drill downs by org units during your analysis.

1. Create a new calculated column. The sample code provided names the column, e.g. Level1, Level2, etc., but you are free to name them as you see fit (e.g. Country, Province, District, etc.) as long as the names do not conflict with an existing column name in the same table.

   a. Under your Fields list, click the ellipsis (“. . .”) at the top right hand corner of the Organisation Unit table, or right click the Organisation Unit table, and choose “New column”.

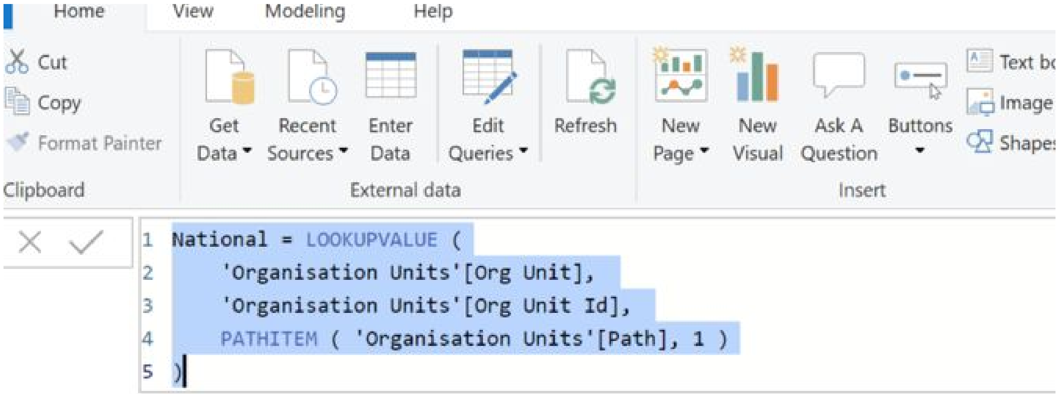


   b. In the function box at the top of the window, enter the following:

```
        Level1 = LOOKUPVALUE (
                'Organisation Units'[Org Unit],
                'Organisation Units'[Org Unit Id],
                PATHITEM ( 'Organisation Units'[Path], 1 )
        )
```

c. At this time, you can choose to rename "Level1" to "National" or other label based on your context.

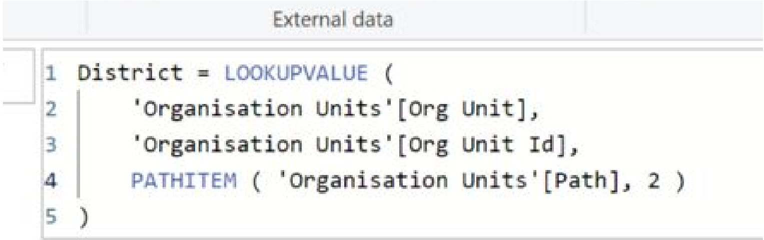
In the example below, we have selected "National" in line with play.DHIS2



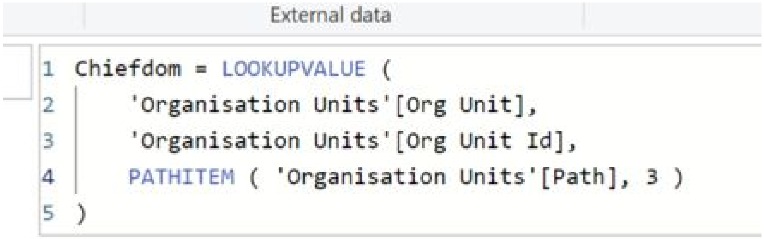
```
National = LOOKUPVALUE (
    'Organisation Units'[Org Unit],
    'Organisation Units'[Org Unit Id],
    PATHITEM ( 'Organisation Units'[Path], 1 )
)
```

d. Repeat the process for remaining OU levels, incrementing the number in the equation accordingly. **Do not forget to increment the PATHITEM, as that is what tells Power BI how to order your organisational units.**

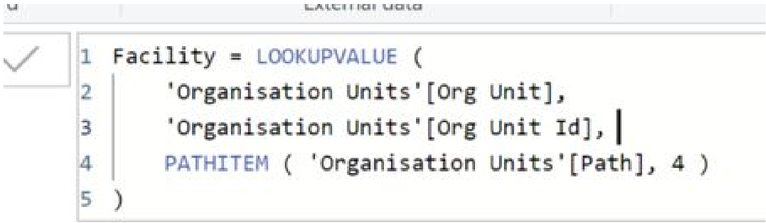
```
        Level2 = LOOKUPVALUE (
                'Organisation Units'[Org Unit],
                'Organisation Units'[Org Unit Id],
                PATHITEM ( 'Organisation Units'[Path], 2 )
        )
```



```
District = LOOKUPVALUE (
    'Organisation Units'[Org Unit],
    'Organisation Units'[Org Unit Id],
    PATHITEM ( 'Organisation Units'[Path], 2 )
)
```



```
Chiefdom = LOOKUPVALUE (
    'Organisation Units'[Org Unit],
    'Organisation Units'[Org Unit Id],
    PATHITEM ( 'Organisation Units'[Path], 3 )
)
```

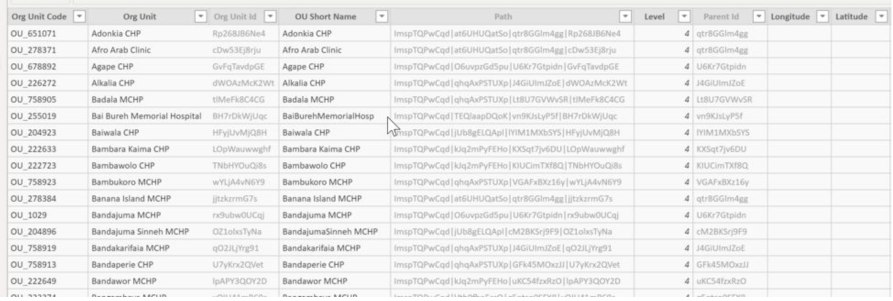


```
Facility = LOOKUPVALUE (
    'Organisation Units'[Org Unit],
    'Organisation Units'[Org Unit Id],
    PATHITEM ( 'Organisation Units'[Path], 4 )
)
```

e. Don’t create columns deeper than you have data, that is, if you only have data at Level4, don’t create Level 5, 6, and 7 columns.

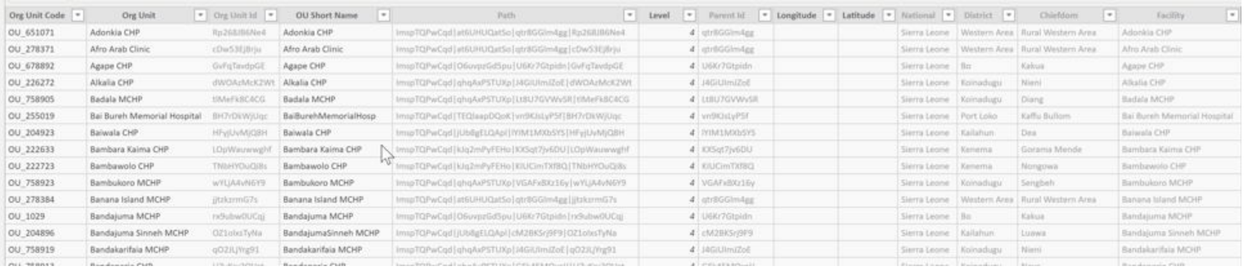
f. Note that when performing these functions, you are adding new metadata structures (data columns) to your data model. If you go into **Table view**, you will see the newly created columns in the OrgUnit table (titled “National”, “District”, “Chiefdom”, “Facility”).

Before



Org Unit Code	Org Unit	Org Unit Id	OU Short Name	Path	Level	Parent Id	Longitude	Latitude
OU_651071	Adonkia CHP	Rp268JB6Ne4	Adonkia CHP	ImspTQPwCqd|at6UHUQatSo|qtr8GGlm4gg|Rp268JB6Ne4	4	qtr8GGlm4gg		
OU_278371	Afro Arab Clinic	cDw53Ej8rju	Afro Arab Clinic	ImspTQPwCqd|at6UHUQatSo|qtr8GGlm4gg|cDw53Ej8rju	4	qtr8GGlm4gg		
OU_678892	Agape CHP	GvFqTavdpGE	Agape CHP	ImspTQPwCqd|O6uvpzGd5pu|U6Kr7Gtpidn|GvFqTavdpGE	4	U6Kr7Gtpidn		
OU_226272	Alkalia CHP	dWOAzMcK2Wt	Alkalia CHP	ImspTQPwCqd|qhqAxPSTUXp|J4GiUImJZoE|dWOAzMcK2Wt	4	J4GiUImJZoE		
OU_758905	Badala MCHP	tlMeFk8C4CG	Badala MCHP	ImspTQPwCqd|qhqAxPSTUXp|Lt8U7GVWvSR|tlMeFk8C4CG	4	Lt8U7GVWvSR		
OU_255019	Bai Bureh Memorial Hospital	BH7rDkWjUqc	BaiBurehMemorialHosp	ImspTQPwCqd|TEQlaapDQoK|vn9KJsLyP5f|BH7rDkWjUqc	4	vn9KJsLyP5f		
OU_204923	Baiwala CHP	HFyjUvMjQ8H	Baiwala CHP	ImspTQPwCqd|jUb8gELQApl|lYIM1MXbSYS|HFyjUvMjQ8H	4	lYIM1MXbSYS		
OU_222633	Bambara Kaima CHP	LOpWauwwghf	Bambara Kaima CHP	ImspTQPwCqd|kJq2mPyFEHo|KXSqt7jv6DU|LOpWauwwghf	4	KXSqt7jv6DU		
OU_222723	Bambawolo CHP	TNbHYOuQi8s	Bambawolo CHP	ImspTQPwCqd|kJq2mPyFEHo|KIUCimTXf8Q|TNbHYOuQi8s	4	KIUCimTXf8Q		
OU_758923	Bambukoro MCHP	wYLjA4vN6Y9	Bambukoro MCHP	ImspTQPwCqd|qhqAxPSTUXp|VGAFxBXz16y|wYLjA4vN6Y9	4	VGAFxBXz16y		
OU_278384	Banana Island MCHP	jjtzkzrmG7s	Banana Island MCHP	ImspTQPwCqd|at6UHUQatSo|qtr8GGlm4gg|jjtzkzrmG7s	4	qtr8GGlm4gg		
OU_1029	Bandajuma MCHP	rx9ubw0UCqj	Bandajuma MCHP	ImspTQPwCqd|O6uvpzGd5pu|U6Kr7Gtpidn|rx9ubw0UCqj	4	U6Kr7Gtpidn		
OU_204896	Bandajuma Sinneh MCHP	OZ1olxsTyNa	BandajumaSinneh MCHP	ImspTQPwCqd|jUb8gELQApl|cM2BKSrj9F9|OZ1olxsTyNa	4	cM2BKSrj9F9		
OU_758919	Bandakarifaia MCHP	qO2JLjYrg91	Bandakarifaia MCHP	ImspTQPwCqd|qhqAxPSTUXp|J4GiUImJZoE|qO2JLjYrg91	4	J4GiUImJZoE		
OU_758913	Bandaperie CHP	U7yKrx2QVet	Bandaperie CHP	ImspTQPwCqd|qhqAxPSTUXp|GFk45MOxzJJ|U7yKrx2QVet	4	GFk45MOxzJJ		
OU_222649	Bandawor MCHP	IpAPY3QOY2D	Bandawor MCHP	ImspTQPwCqd|kJq2mPyFEHo|uKC54fzxRzO|IpAPY3QOY2D	4	uKC54fzxRzO		

After



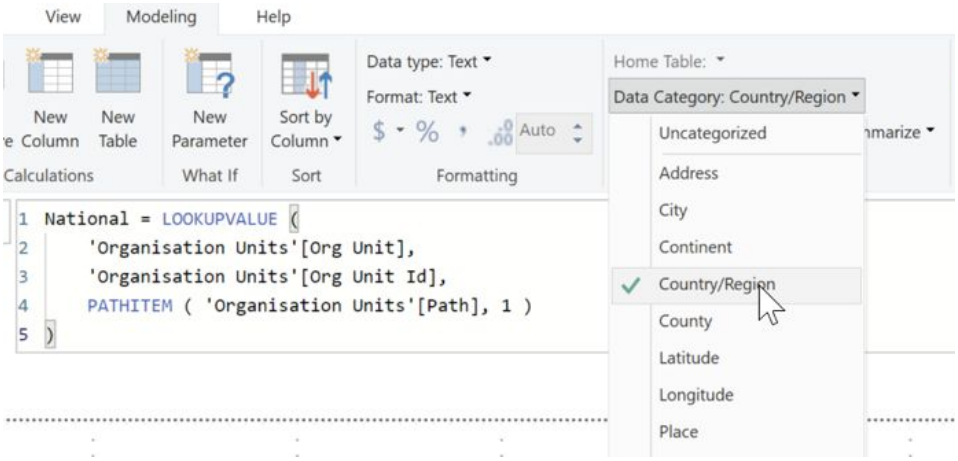
Org Unit Code	Org Unit	Org Unit Id	OU Short Name	Path	Level	Parent Id	Longitude	Latitude	National	District	Chiefdom	Facility
OU_651071	Adonkia CHP	Rp268JB6Ne4	Adonkia CHP	ImspTQPwCqd|at6UHUQatSo|qtr8GGlm4gg|Rp268JB6Ne4	4	qtr8GGlm4gg			Sierra Leone	Western Area	Rural Western Area	Adonkia CHP
OU_278371	Afro Arab Clinic	cDw53Ej8rju	Afro Arab Clinic	ImspTQPwCqd|at6UHUQatSo|qtr8GGlm4gg|cDw53Ej8rju	4	qtr8GGlm4gg			Sierra Leone	Western Area	Rural Western Area	Afro Arab Clinic
OU_678892	Agape CHP	GvFqTavdpGE	Agape CHP	ImspTQPwCqd|O6uvpzGd5pu|U6Kr7Gtpidn|GvFqTavdpGE	4	U6Kr7Gtpidn			Sierra Leone	Bo	Kakua	Agape CHP
OU_226272	Alkalia CHP	dWOAzMcK2Wt	Alkalia CHP	ImspTQPwCqd|qhqAxPSTUXp|J4GiUImJZoE|dWOAzMcK2Wt	4	J4GiUImJZoE			Sierra Leone	Koinadugu	Nieni	Alkalia CHP
OU_758905	Badala MCHP	tlMeFk8C4CG	Badala MCHP	ImspTQPwCqd|qhqAxPSTUXp|Lt8U7GVWvSR|tlMeFk8C4CG	4	Lt8U7GVWvSR			Sierra Leone	Koinadugu	Diang	Badala MCHP
OU_255019	Bai Bureh Memorial Hospital	BH7rDkWjUqc	BaiBurehMemorialHosp	ImspTQPwCqd|TEQlaapDQoK|vn9KJsLyP5f|BH7rDkWjUqc	4	vn9KJsLyP5f			Sierra Leone	Port Loko	Kaffu Bullom	Bai Bureh Memorial Hospital
OU_204923	Baiwala CHP	HFyjUvMjQ8H	Baiwala CHP	ImspTQPwCqd|jUb8gELQApl|lYIM1MXbSYS|HFyjUvMjQ8H	4	lYIM1MXbSYS			Sierra Leone	Kailahun	Dea	Baiwala CHP
OU_222633	Bambara Kaima CHP	LOpWauwwghf	Bambara Kaima CHP	ImspTQPwCqd|kJq2mPyFEHo|KXSqt7jv6DU|LOpWauwwghf	4	KXSqt7jv6DU			Sierra Leone	Kenema	Gorama Mende	Bambara Kaima CHP
OU_222723	Bambawolo CHP	TNbHYOuQi8s	Bambawolo CHP	ImspTQPwCqd|kJq2mPyFEHo|KIUCimTXf8Q|TNbHYOuQi8s	4	KIUCimTXf8Q			Sierra Leone	Kenema	Nongowa	Bambawolo CHP
OU_758923	Bambukoro MCHP	wYLjA4vN6Y9	Bambukoro MCHP	ImspTQPwCqd|qhqAxPSTUXp|VGAFxBXz16y|wYLjA4vN6Y9	4	VGAFxBXz16y			Sierra Leone	Koinadugu	Sengbeh	Bambukoro MCHP
OU_278384	Banana Island MCHP	jjtzkzrmG7s	Banana Island MCHP	ImspTQPwCqd|at6UHUQatSo|qtr8GGlm4gg|jjtzkzrmG7s	4	qtr8GGlm4gg			Sierra Leone	Western Area	Rural Western Area	Banana Island MCHP
OU_1029	Bandajuma MCHP	rx9ubw0UCqj	Bandajuma MCHP	ImspTQPwCqd|O6uvpzGd5pu|U6Kr7Gtpidn|rx9ubw0UCqj	4	U6Kr7Gtpidn			Sierra Leone	Bo	Kakua	Bandajuma MCHP
OU_204896	Bandajuma Sinneh MCHP	OZ1olxsTyNa	BandajumaSinneh MCHP	ImspTQPwCqd|jUb8gELQApl|cM2BKSrj9F9|OZ1olxsTyNa	4	cM2BKSrj9F9			Sierra Leone	Kailahun	Luawa	Bandajuma Sinneh MCHP
OU_758919	Bandakarifaia MCHP	qO2JLjYrg91	Bandakarifaia MCHP	ImspTQPwCqd|qhqAxPSTUXp|J4GiUImJZoE|qO2JLjYrg91	4	J4GiUImJZoE			Sierra Leone	Koinadugu	Nieni	Bandakarifaia MCHP

2. Setting the data category to Country, Region, etc. increases the accuracy of the map visuals.

   a. With the column selected, or while you are performing the functions above, go to the Modelling tab.

   b. Under Data Category, select an appropriate category for your organisational units.

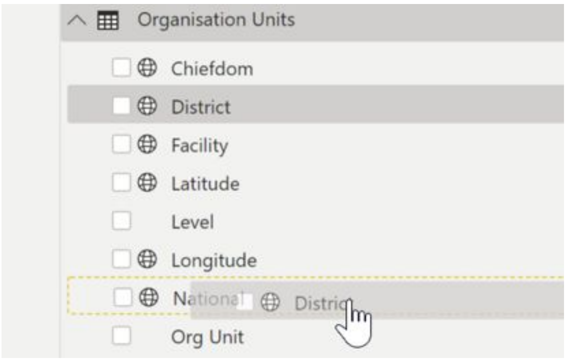
   For example: Categorize the national level to “Country/Region”, district to “Country/Region”, chiefdom to “County”, and facility to “Place”.



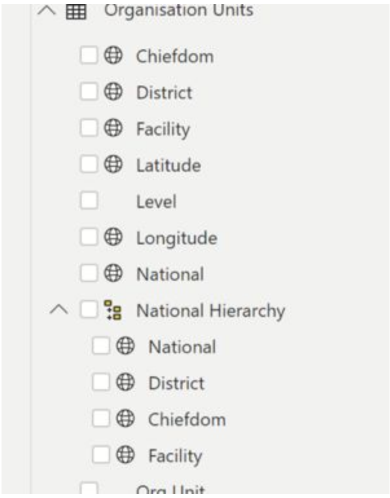
```
National = LOOKUPVALUE (
    'Organisation Units'[Org Unit],
    'Organisation Units'[Org Unit Id],
    PATHITEM ( 'Organisation Units'[Path], 1 )
)
```

3. Create the hierarchy list in your Organisational Units table.

   a. Drag Level2 (e.g. "District") onto the Level1 (e.g. "National") column.

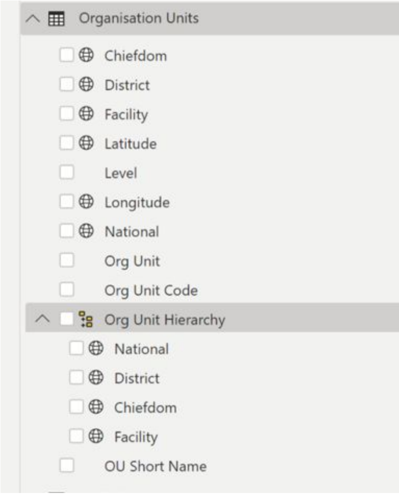


   b. Drag additional Level columns onto the new hierarchy to complete the process.



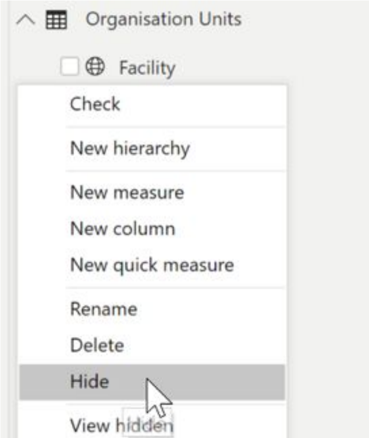
   c. As you may notice above, Power BI automatically named the hierarchy "National Hierarchy". Your hierarchy name may vary, based on what you named your OU

Level1. To change the name of the hierarchy, double click the hierarchy column and rename it e.g. "Org Unit hierarchy"

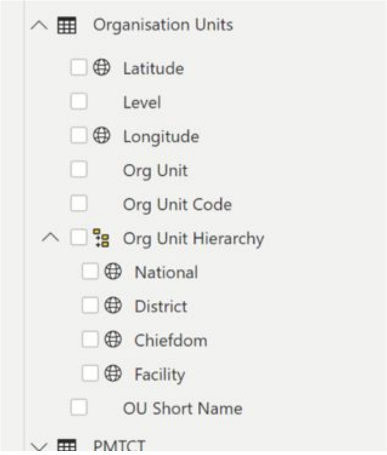


4. Hide the original, non-hierarchical Level* columns if the report developer will not be using them.

    a. Right click or use the ellipsis and select "Hide".



    b. Your resulting Organization Units table may look similar to this:



Organisation Units

□ Latitude

□ Level

□ Longitude

□ Org Unit

□ Org Unit Code

□ Org Unit Hierarchy

□ National

□ District

□ Chiefdom

□ Facility

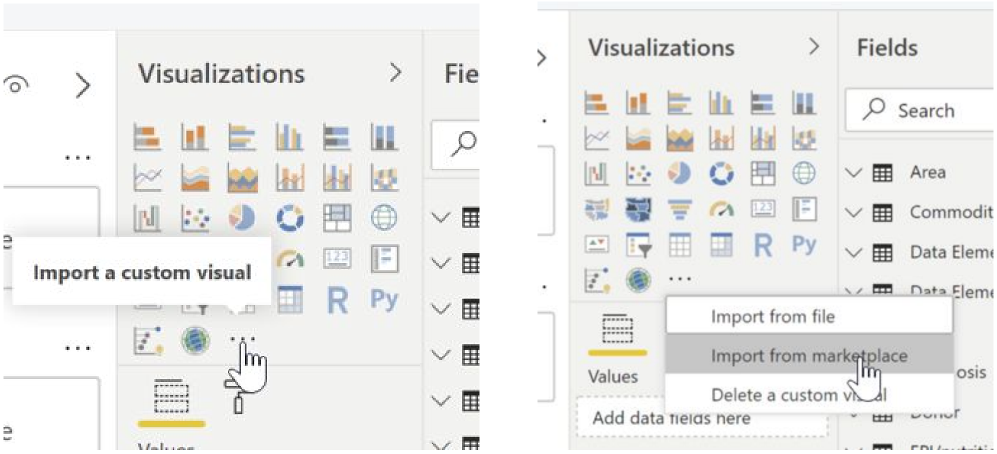
□ OU Short Name

PMTCT

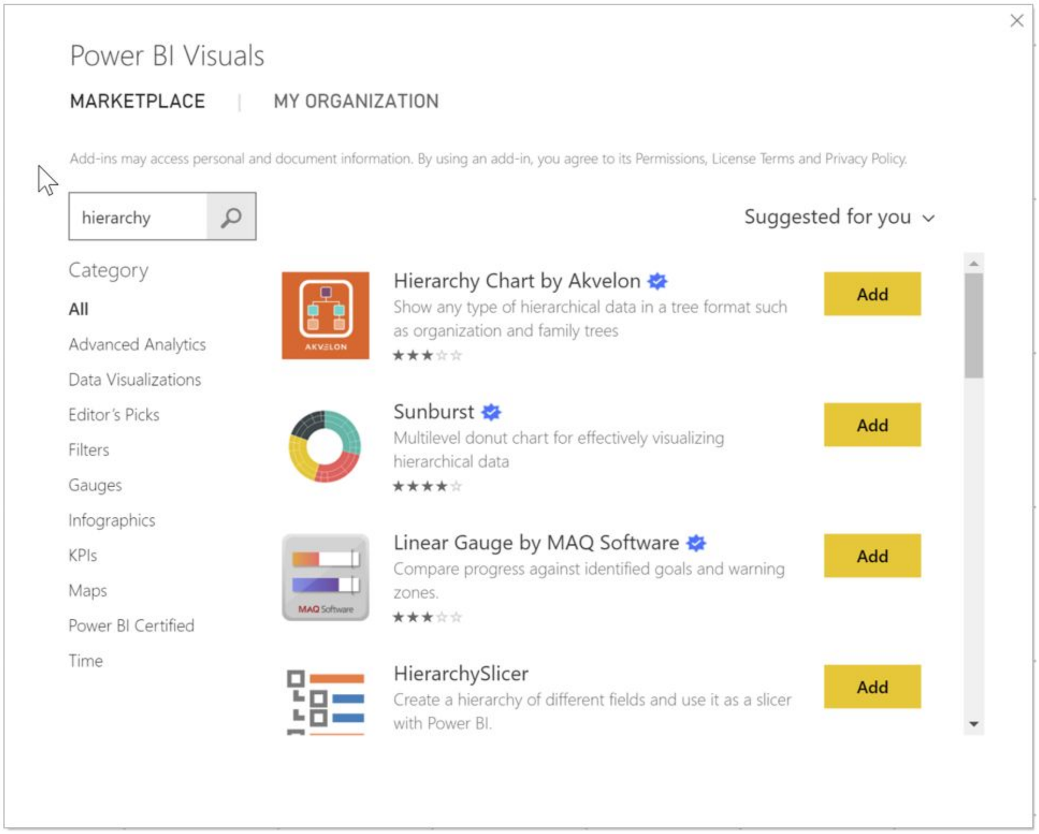
Import the hierarchy slicer

Power BI's hierarchy slicer is an extremely useful visual for data consumers to use, and enables filtering of data by organisational units using a hierarchical structure (similar to DHIS2 analytics). The Hierarchy Slicer is not a native application in Power BI's visual toolbox, but can be easily downloaded and added to your desktop Power BI file (.PBIX).

1. Under the Visualizations panel, click the ellipses to import a custom visual, and select "Import from Marketplace".



2. Search for the term 'hierarchy' and select "HierarchySlicer" from the results. Click "Add".



Power BI Visuals

MARKETPLACE | MY ORGANIZATION

Add-ins may access personal and document information. By using an add-in, you agree to its Permissions, License Terms and Privacy Policy.

hierarchy

Suggested for you

Category

- All
- Advanced Analytics
- Data Visualizations
- Editor's Picks
- Filters
- Gauges
- Infographics
- KPIs
- Maps
- Power BI Certified
- Time

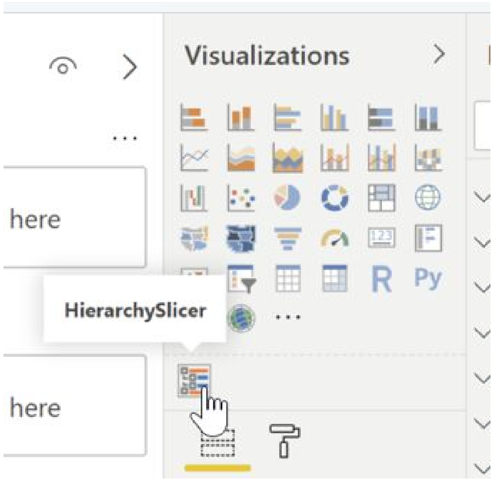
Hierarchy Chart by Akvelon — Show any type of hierarchical data in a tree format such as organization and family trees ★★★☆☆ — Add

Sunburst — Multilevel donut chart for effectively visualizing hierarchical data ★★★★☆ — Add

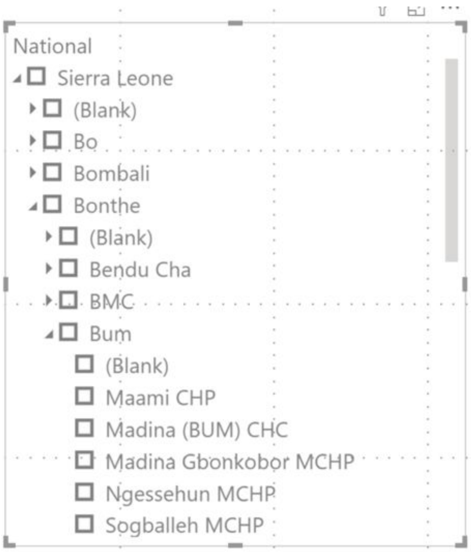
Linear Gauge by MAQ Software — Compare progress against identified goals and warning zones. ★★★☆☆ — Add

HierarchySlicer — Create a hierarchy of different fields and use it as a slicer with Power BI. — Add

3. Find the new Hierarchy Slicer in your visual toolbox.



4. Explore your new OU Hierarchy using the slicer.

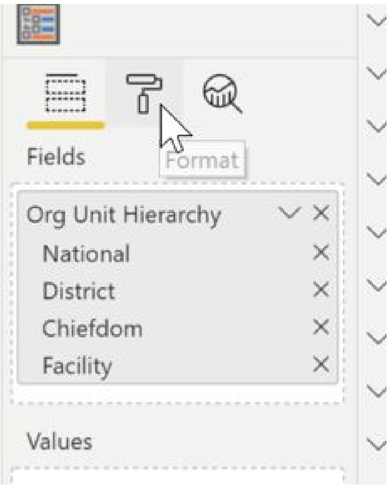


National
- ☐ Sierra Leone
  - ☐ (Blank)
  - ☐ Bo
  - ☐ Bombali
  - ☐ Bonthe
    - ☐ (Blank)
    - ☐ Bendu Cha
    - ☐ BMC
    - ☐ Bum
      - ☐ (Blank)
      - ☐ Maami CHP
      - ☐ Madina (BUM) CHC
      - ☐ Madina Gbonkobor MCHP
      - ☐ Ngessehun MCHP
      - ☐ Sogballeh MCHP

There are a number of steps you can perform to customize the appearance and functionality of the slicer.

a. Format visual properties. With the slicer visual selected, go to the **Visual Properties** tab (marked by a paint roller icon). This tab will enable you to edit the visual properties without changing the underlying metadata and data.

Customization is entirely up to you, and should be based on your needs.



Fields

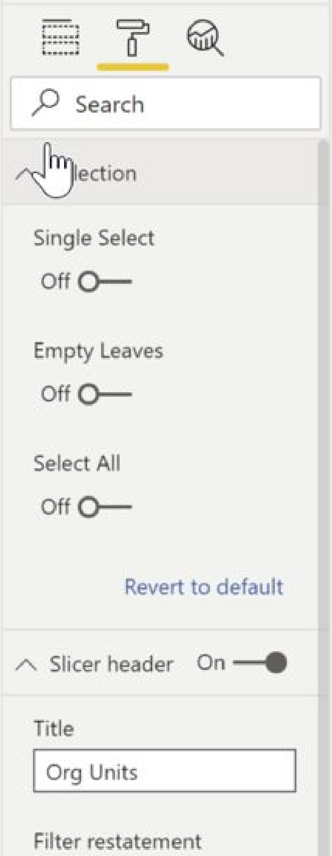
Org Unit Hierarchy
- National
- District
- Chiefdom
- Facility

Values

b. Allow multi-select. Move the Single Select slider to “Off”. This will enable you to control hold and select multiple org units.

c. Remove blanks. Move the Empty Leaves slider to “Off”.

d. Change the name of the visual title to “Org Units”



Search

Selection

Single Select

Off

Empty Leaves

Off

Select All

Off

Revert to default

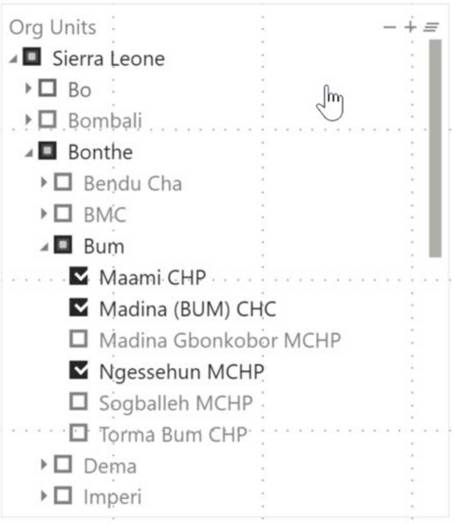
Slicer header On

Title

Org Units

Filter restatement

e. Note the changes to the org unit slicer.

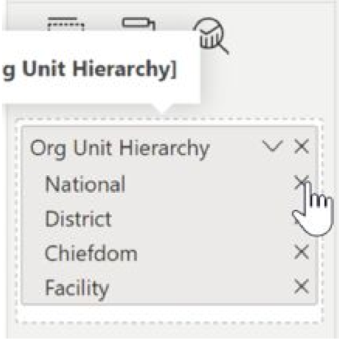


Org Units

- ■ Sierra Leone
  - □ Bo
  - □ Bombali
  - ■ Bonthe
    - □ Bendu Cha
    - □ BMC
    - ■ Bum
      - ☑ Maami CHP
      - ☑ Madina (BUM) CHC
      - □ Madina Gbonkobor MCHP
      - ☑ Ngessehun MCHP
      - □ Sogballeh MCHP
      - □ Torma Bum CHP
    - □ Dema
    - □ Imperi

f. As you will note in the slicer above, the Level1 organisational unit is visible. You may want to delete the top hierarchical level from your visual.

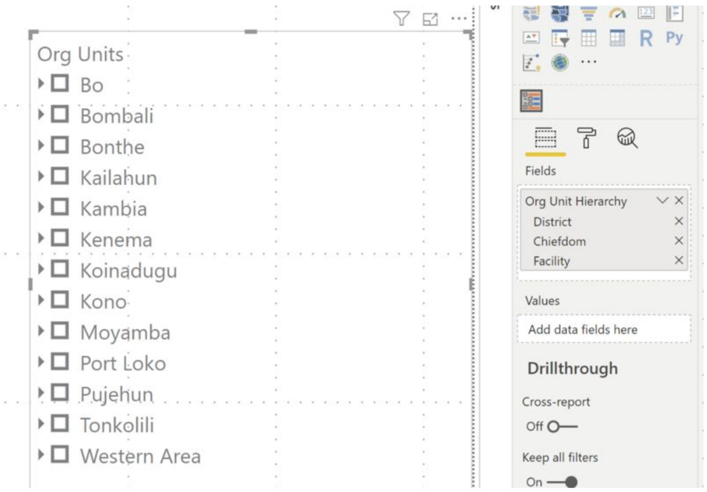
Under the visuals tab, select the “x” under the Org Unit Hierarchy to remove “Level1/Sierra Leone”. Performing this operation will only remove Sierra Leone from the visual - it will not change the metadata or data in the model.



g Unit Hierarchy]

Org Unit Hierarchy
- National
- District
- Chiefdom
- Facility

g. View your customized slicer.



Org Units
- Bo
- Bombali
- Bonthe
- Kailahun
- Kambia
- Kenema
- Koinadugu
- Kono
- Moyamba
- Port Loko
- Pujehun
- Tonkolili
- Western Area

Fields

Org Unit Hierarchy
- District
- Chiefdom
- Facility

Values

Add data fields here

Drillthrough

Cross-report: Off

Keep all filters: On

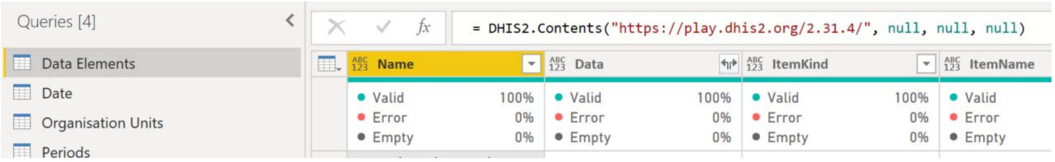
Create date hierarchies

Similar to the organisational unit hierarchy, Power BI will not automatically configure your a date hierarchy. Use of a date hierarchy may be useful for your analysis. Should you desire a date hierarchy, follow the steps above that were used to create your org unit hierarchy.

Explore your dataset

A Power BI template (.pbit) provides all the transformation metadata, relationships, and visuals of a Power BI model, though it omits the actual data. This makes the template compact, secure, and portable, and provides an ideal starting point for developers with similar modeling needs. The template was created by first stepping through all the steps outlined above. Once the data model was complete, several parameters were added to the template to facilitate switching the DHIS2

Url, Start Date, and End Date, and you will be prompted to enter values for these parameters when you first open the template. You can look through each table definition to see how the first step of each replaces the literal value from the connector with a parameter reference. Parameterizing your queries is not necessary, but it may facilitate internal reuse and deployment between dev, stage, and production environments.



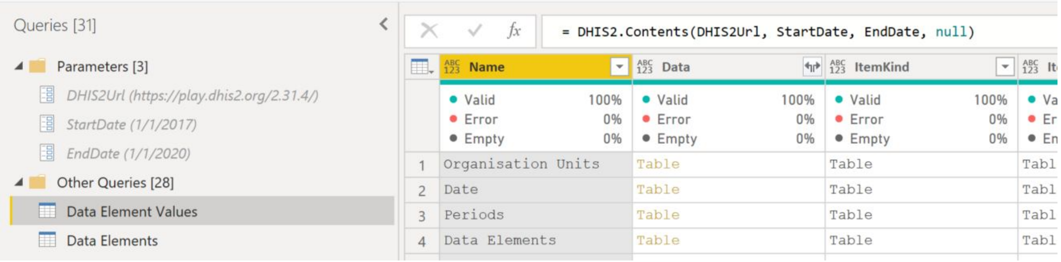
Queries [4]

- Data Elements
- Date
- Organisation Units
- Periods

```
= DHIS2.Contents("https://play.dhis2.org/2.31.4/", null, null, null)
```

Name	Data	ItemKind	ItemName
● Valid 100%	● Valid 100%	● Valid 100%	● Valid
● Error 0%	● Error 0%	● Error 0%	● Error
● Empty 0%	● Empty 0%	● Empty 0%	● Empty

First step of Data Element query showing literal string from DHIS2 connector



Queries [31]

- Parameters [3]
  - DHIS2Url (https://play.dhis2.org/2.31.4/)
  - StartDate (1/1/2017)
  - EndDate (1/1/2020)
- Other Queries [28]
  - Data Element Values
  - Data Elements

```
= DHIS2.Contents(DHIS2Url, StartDate, EndDate, null)
```

	Name	Data	ItemKind	It
	● Valid 100% ● Error 0% ● Empty 0%	● Valid 100% ● Error 0% ● Empty 0%	● Valid 100% ● Error 0% ● Empty 0%	● Va ● Er ● En
1	Organisation Units	Table	Table	Tabl
2	Date	Table	Table	Tabl
3	Periods	Table	Table	Tabl
4	Data Elements	Table	Table	Tabl

First step of Data Element query after swapping string literal for new DHIS2Url, StartDate, and EndDate parameters. The parameter declarations are shown on the top left.

Once you have opened the template, entered parameter values, and loaded data, you may choose to save your work as a Power BI file to include the data (.pbix). Doing so will preserve the data and you won't have to wait through the refresh time when you next work on it (unless you make changes using the Query Editor).

Note, the template was created from an instance of play.dhis2.org, typically https://play.dhis2.org/2.31.4/. It is likely that pointing the Url to an instance other than play.dhis2.org will generate errors as the naming and structure of the dynamic dimensions will not align with the templated data model.

The first tab of the template, All Slicers, contains a slicer for each dynamic dimension, slicers for Data Elements and Organisation Units, and displays the sum of all Data Element Values reflecting the state of all filtered selections. This page is only valuable as a test harness; in practice, one should significantly limit the number of slicers on a page for both performance and usability reasons. The table relationships in the template are almost exclusively 'single' direction. This page is particularly useful in demonstrating the slicer behavior when relationship directions are set to 'both'. Feel free to experiment to best understand what works for you and your report users.

The second tab of the template, Immunization, loosely recreates the top four graphs from the play.dhis2.org Immunization dashboard. These visuals demonstrate how one might assemble simple graphs and maps, but they barely scratch the surface of the capabilities of Power BI. It's important to note these visuals reference measures which may not be optimal, and as stated previously, they do little to demonstrate effective use of Power BI interactive graphics, analytics, and storytelling.

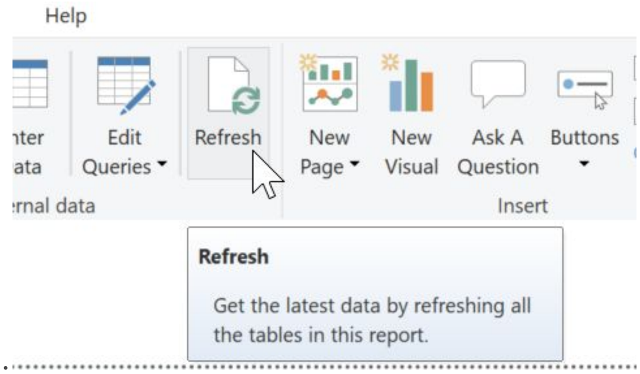
Refreshing data

At the time of writing this user manual, there are currently two processes you may choose from to refresh your DHIS2 data in Power BI. Note that all of the data modelling steps you applied will be retained when you refresh your data. Depending on the expected frequency of refresh, resources, and capacities, you may opt to refresh your data using either process.

In Power BI Desktop

Manually refreshing your data using Power BI Desktop and republishing to the Power BI Service may be time consuming, however, if you require infrequent refresh of your data (e.g. monthly), then this may be a simpler approach compared to using a Gateway.

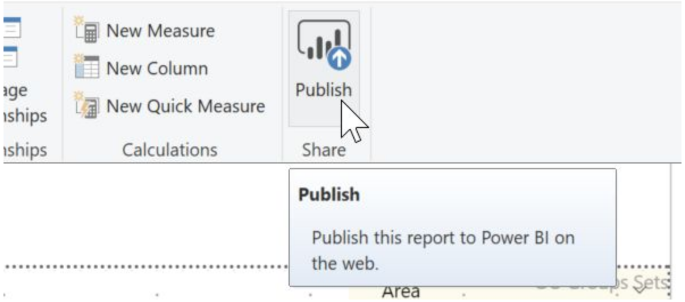
1. In the Home tab, select "Refresh"



Refresh

Get the latest data by refreshing all the tables in this report.

2. Once your data model has been updated with the newest data available in DHIS2, you will need to republish your report(s) to the Power BU Service.



Publish

Publish this report to Power BI on the web.

Use a Gateway

Custom connectors are not available in the Power BI Service (cloud) - this service is maintained only for approved Microsoft connectors[1]. In order to refresh data, the published Power BI content will need to access the connector through a gateway hosted on a computer or VM you maintain. The gateway computer can be on-premise or in the cloud, but it must be able to access the internet. The gateway computer does not require any inbound firewall ports to be opened.

Use of a Gateway may prove more efficient, particularly if the frequency of refresh needed is more frequent (e.g. weekly). The Gateway will enable you to schedule an auto refresh, eliminating the need to manually republish your Power BI content when data change. Documentation is provided below, with more extensive support provided by Microsoft.

1. Install the Gateway.
    a. Download using the link on this page: https://powerbi.microsoft.com/en-us/gateway/

    b. The gateway can be installed in either Personal or Enterprise mode. Refresh should be supported in either mode, but at the time of this writing, only Enterprise mode has been successfully tested, thus only Enterprise installation will be provided. Enterprise runs as a Windows Service in the background, and uses a service account.

    c. The DHIS2 Custom Connector must be present on the gateway computer. The ideal location varies depending on the gateway mode, but the connector must be accessible by the account running the gateway. For Enterprise mode, the default will likely be C:\Windows\ServiceProfiles\PBIEgwService\Documents\Power BI Desktop\Custom Connectors\ . Create this directory and copy the connector file to it.

    d. Enterprise Mode:

        i. Run the installer.

        ii. Choose next until you get to the “Choose the type of gateway you need” page.

        iii. Select the first (default) option and choose ‘Next’.

        iv. Accept the terms of use and choose ‘Install’.

        v. Enter the email address associated with your Power BI license and Sign In.

        vi. You will now register the gateway with your PBI Service. You may install multiple gateways (on different computers) in order to provide high availability to your refresh pipeline. See Microsoft’s help for more

[1] At the time of writing this user manual, the custom connector has not yet been approved by MIcrosoft. As part of the connector’s roadmap, BAO Systems hopes to have this connector approved, thereby reducing some of the current steps to connect and refresh data.

information. For our purposes, we will create a new, single-node cluster. Enter a name for the new gateway, e.g., DHIS2Gateway.

vii. Enter and reenter a recovery key.

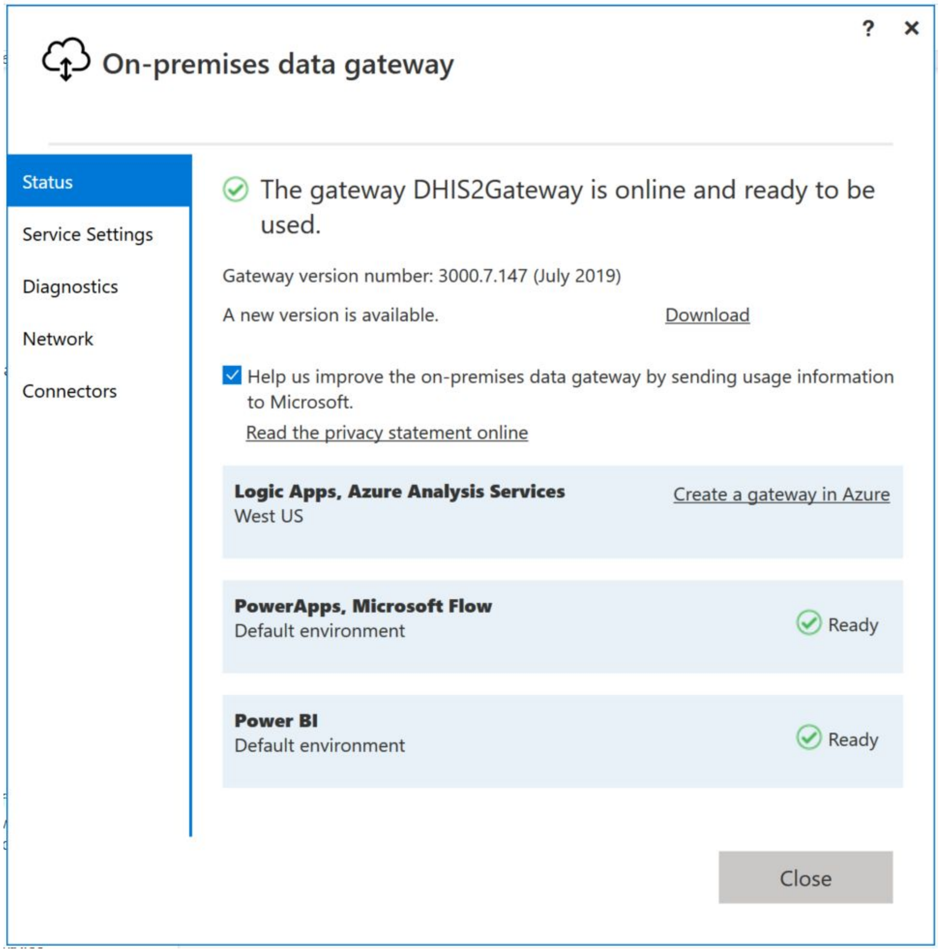
viii. Accept the default region and press Configure.

ix. Once signed in, the window should show a status “Online and ready to use”.

x. Switch to the Diagnostics tab and execute the “Start new test” link near the bottom. Within a minute, the test should complete successfully.

xi. Go to the Connectors tab and verify the DHIS2 connector is shown in the list. Change the folder if necessary, though keep in mind the connector must be accessible by the Service Account.

xii. The Enterprise Gateway runs in the background as a Windows Service, and should continue running even when the Gateway window has been closed.



✓ The gateway DHIS2Gateway is online and ready to be used.

Gateway version number: 3000.7.147 (July 2019)

A new version is available.

☑ Help us improve the on-premises data gateway by sending usage information to Microsoft.

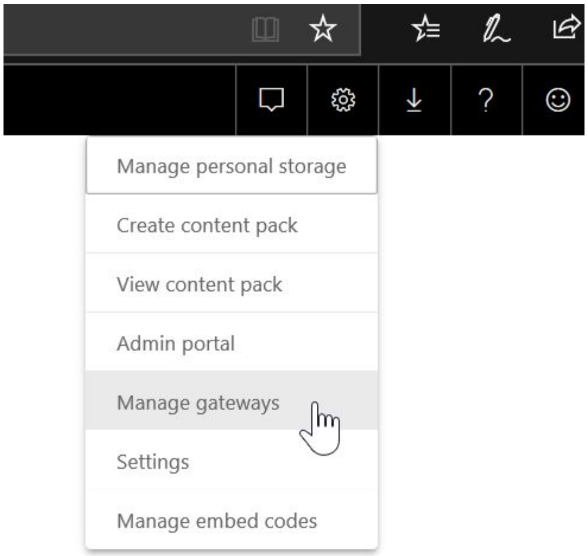
Logic Apps, Azure Analysis Services
West US

PowerApps, Microsoft Flow
Default environment
✓ Ready

Power BI
Default environment
✓ Ready

2. Configuring the Gateway

   a. Log into the PBI Service (https://app.powerbi.com/) and choose “Manage Gateways” from the option menu.



b. Check the second box "Allow user's custom data connectors to refresh…" and select Apply.

c. Click "Add data sources to use the gateway".

d. Name the data source, e.g., DHIS2.

e. Select the Data Source Type dropdown and choose DHIS2 from the list. If DHIS2 is not present, examine your enterprise gateway configuration.

f. Enter the Base Url, Username, and Password for your DHIS2 instance. The Url provided must exactly match the Url you used with your Power BI Desktop data model.

   i. The credentials entered on this screen are connecting to DHIS2 for all published data models which are permitted to use to this gateway data source.

g. Configure the Users tab to manage which report authors are allowed to leverage this gateway data source.

3. Publish your content.
4. Set the refresh schedule.

   a. Go to the Dataset settings page.

   b. Expand Gateway Connection and enable Use a data gateway.

   c. Select the 'arrow' dropdown in the Actions section.

   d. Select your gateway data source (e.g., DHIS2) from the 'Maps to' dropdown.

   e. Press 'Apply' to save your configuration.

f. Expand Data Source Credentials and verify you are using the credentials provided by the administrator.

g. You may now configure a refresh schedule.

Troubleshooting

Frequently encountered issues

The Connector doesn't appear in the PBI Desktop Get Data list

- Double check that your connector is installed per instructions above, paying close attention to spacing and casing
- Restart Power BI, if you had it open prior to installing the connector. In some cases, there may be two versions of Power BI installed on your computer. Often, selecting Power BI from your start menu will bring up the most recent version
- Restart your computer

The Connector doesn't appear in the Gateway Data Source Type list

- Use the diagnostics from the Gateway app to verify successful connectivity
- Ensure the custom connector appears in the Gateway connectors list
- Ensure the Gateway service account has access to the installed connector

Connector takes a long time to load data

- Loading the entire Data Element Value table is expected to take a long time, depending on network speed, latency, DHIS2 available resources, number of logged in users, etc.

Connector can't connect to DHIS2

- Double check the credentials you are entering are valid
- Work with your IT support to verify sufficient network connectivity

Some Data Element Values are missing or vary between refreshes

- Ensure your DHIS2 instance has sufficient memory to process potentially significant API requests

- Verify your DHIS2 instance is responsive

No Data Element Values are present

- No values within provided Start and End Dates
- Credentials used do not have sufficient access

Warning icons displayed in the Get Data dialog

- While the connector dialogs accept a Url and prompt for credentials, these values are not validated until you attempt to access data. If you encounter errors when attempting to expand the Get Data 'Dimensions' folder or when requesting Table Previews, it is likely your DHIS2 instance can't be reached. Verify the Url is correct and ends with a trailing slash. Also be sure your DHIS2 instance is online and accessible using the same credentials entered in Power BI.

BAO Systems Helpdesk