Senturus Analytics Connector

User Guide

Cognos to Power BI

Contents

Overview

This guide describes how the Senturus Analytics Connector is used from Power BI after it has been configured.

Please refer to the *Senturus Analytics Connector Installation Guide* for instructions on installing and configuring the Analytics Connector.

The Analytics Connector is designed for Power BI and has been tested against different versions, including both Power BI Desktop and Power BI Gateway on Windows starting with the August 2018 release. For optimal stability and performance, it is recommended that you keep your Power BI software fully updated.

Data Connector vs SQL Server Database Connection

Starting from version 5.0, you can connect from Power BI Desktop and Power BI Gateway to Analytics Connector via SQL Server database connection.

If you migrated from version 4.x, you may still use our custom Data Connector to connect, all your old reports and data sources will continue working. And you can still use our custom Data Connector in new reports if you wish.

Comparing to custom Data Connector, using SQL Server database connection gives you following advantages:

1. No need to install Analytics Connector ODBC client.
2. No need to copy custom Data Connector (.mez files).
3. Use Windows authentication from Power BI Desktop to Analytics Connector server to Cognos.
4. More Power BI functions are supported, e.g. query cancellation.

But there is a known issue when using SQL Server database connection to connect to Analytics Connector, you cannot filter decimal columns. This issue will be fixed in later releases.

Our custom Data Connector wraps Analytics Connector ODBC client and enables DirectQuery capabilities.

To use custom Data Connector, follow steps below:

1. Copy the Custom Data Connector “SenturusAnalyticsConnector.mez” from Analytics Connector server (under “<install_root>\Power BI” folder) to you Power BI Desktop computer (under “{User}\Documents\Power BI Desktop\Custom Connectors” folder).

   If your Senturus Analytics Connector server administrators made some changes using Senturus Data Source Configuration tool or Service Configuration tool, for example mapped a new database to a Cognos package, or moved Analytics Connector server to a new host, a new “SenturusAnalyticsConnector.mez” file will be generated and you need to copy it again.

2. Start Power BI Desktop, go to Files -> Options and Settings -> Options, select Security on left panel, and under “Data Extensions” option, check “(Not Recommended) Allow any extensions to load without validation or warning.” You need to restart Power BI Desktop.

Connecting to Analytics Connector Server from Power BI Desktop

You don't connect to Cognos directly from Power BI Desktop or Power BI Gateway. Instead you connect to Analytics Connector server and Analytics Connector server connects to Cognos dispatcher on your behalf. For more information, please refer to "*Senturus Analytics Connector Installation Guide*", section Architecture Overview.

As mentioned before, you can connect to Analytics Connector Server using either Power BI built-in SQL Server database connector, or Senturus custom Data Connector.

Conecting via SQL Server database connector

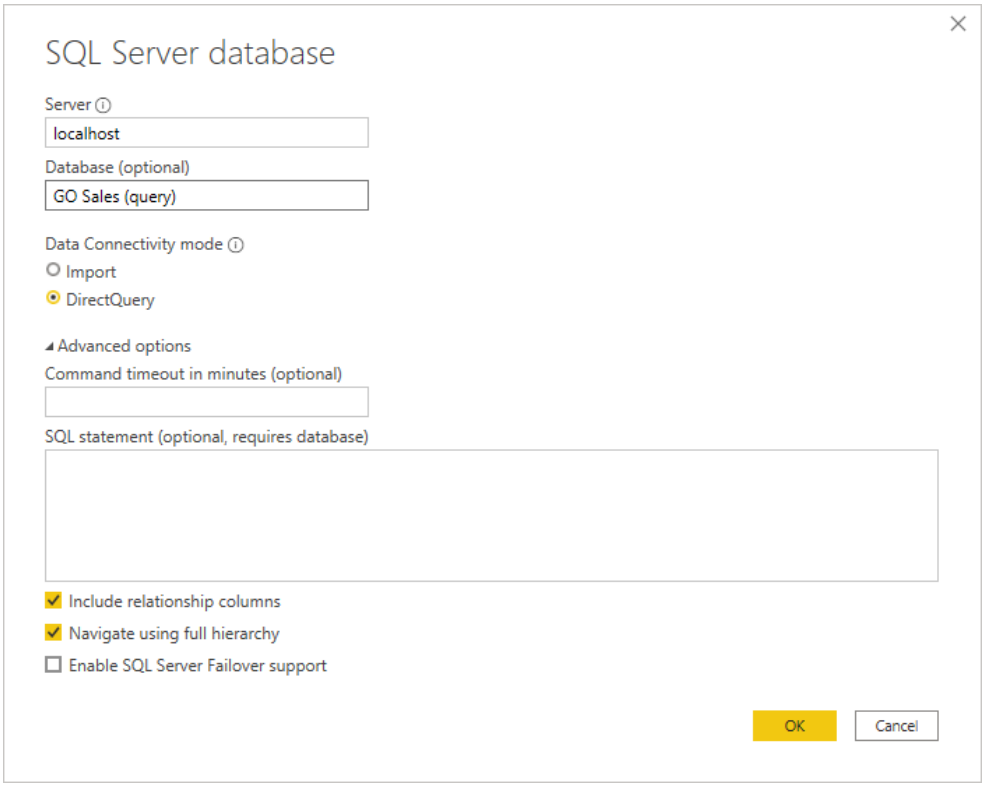
To begin, open Power BI Desktop and select **Get Data -> SQL Server** from the ribbon. "SQL Server database" dialog pops up.

In **Server** input box, Type in hostname of IP of Analytics Connector server.

In **Database (optional)** input box, type in database name given by your Analytics Connector administrator when he/she mapped a Cognos package or data module to it. Please note, Database is mendentary in this case.

Check **DirectQuery** and (optionally) check "**Navigate using full hierarchy**".

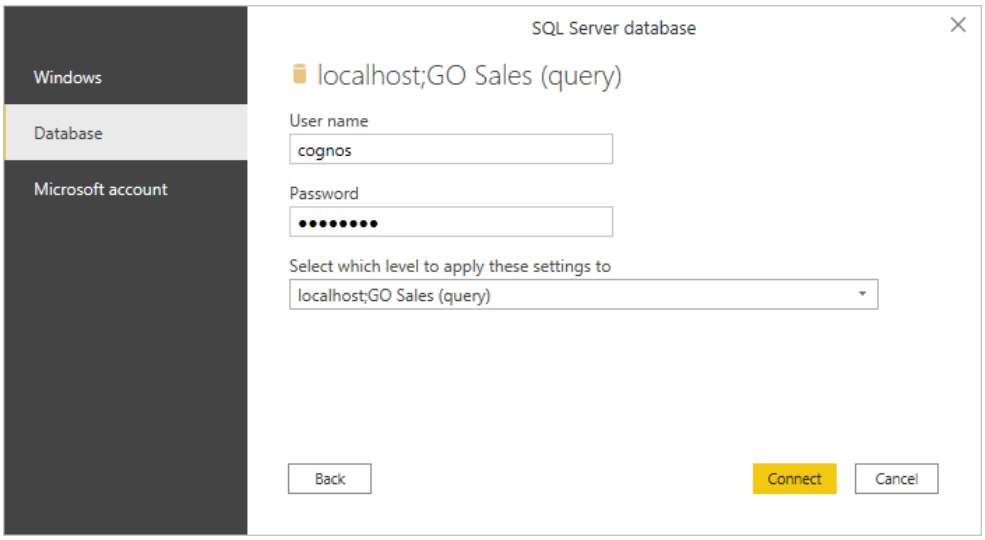
Then click on **OK** button.



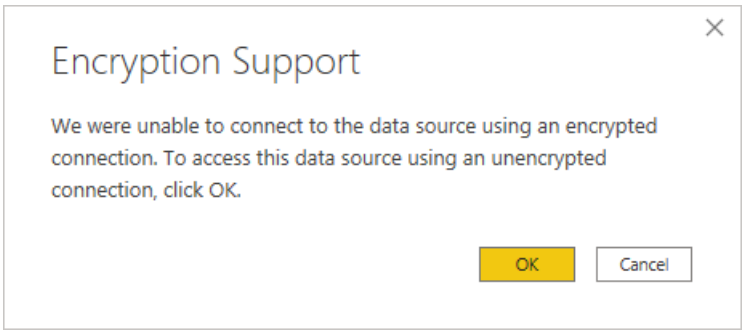
You can use either **Windows** authentication or **Database** authentication (by providing your Cognos login user name and password). But Analytics Connector server does not support **Microsort account** authentication yet.

NOTE: Power BI desktop will securely store these credentials and skip this dialog if you have entered before.

To manage or delete the stored authentication information, go to **File → Options and settings → Data source settings**.



You may see a warning popup, because Analytics Connector, as of version 5.0, does not support SSL yet. Click **OK** to dismiss this message.



If your database is mapped from Cognos package, you will see following objects in Navifator panel:

1. Each namespace is mapped to a schema. Plus some system schemas as you will see when connecting to a real SQL server database.
2. Each query subject is mapped to a table.
3. Each query item will be mapped to a column. Plus a dummy column AAA_LINK in each mapped table. See *Creating Relationships* regarding more informaton.
4. If you have standalone calculations under a namespace, each calculation is mapped to a column under a special table **AAA_CALCULATIONS** (table name is configurable in Data

Source Configuration tool). You can use this special table same as other query subject mapped tables.

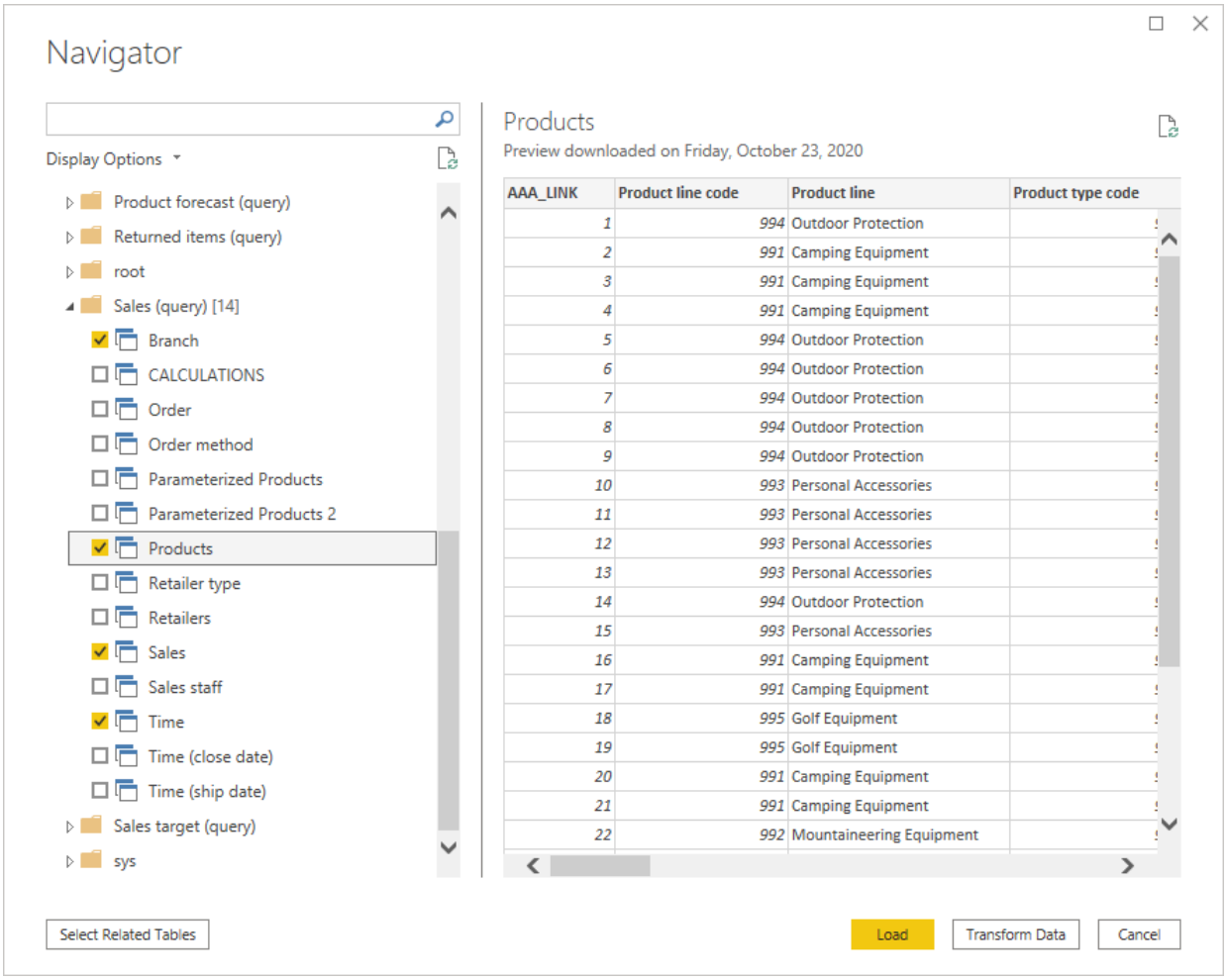
5. If you configured schema to report folder mappings, each report will be mapped to a table under a schema. However, you cannot use report mapped tables as regular tables, please refer to *Working with Reports* for more information.

If your database is mapped from Cognos data module, you will see following objects in Navifator panel:

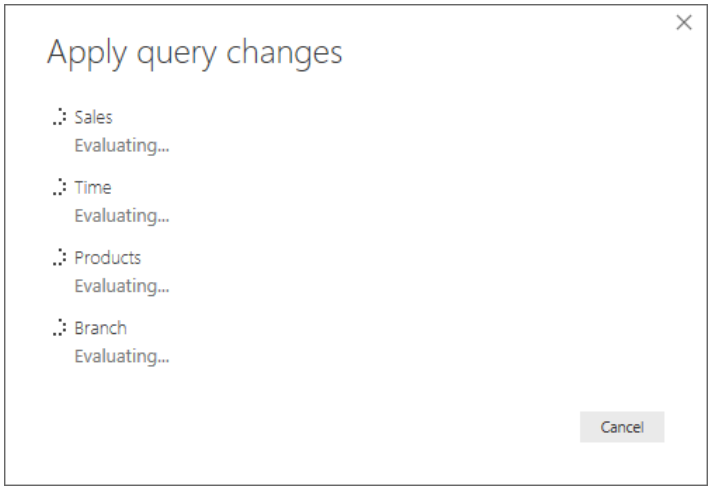
1. Only one schema with the same name as your data module. Plus some system schemas as you will see when connecting to a real SQL server database.
2. Each table in data module is mapped to a SQL Server database table.
3. If you configured schema to report folder mappings, each report will be mapped to a table under a schema. However, you cannot use report mapped tables as regular tables, please refer to Working with Reports for more information.

Select only tables you need to pull columns from and click on **Load** button.

Hint, you don't have to wait table preview to be loaded before clicking on Load button, and you can uncheck **Display Options -> Enable data previews** option to skip previews.



Power BI will then inspect the selected tables and load them into report.



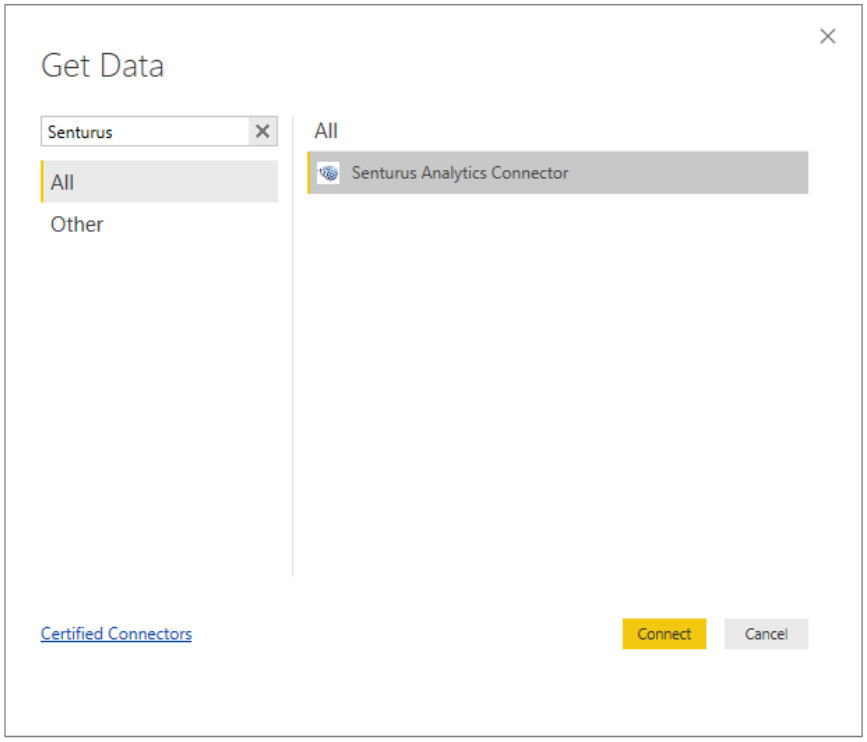
After loading tables into Power BI, you need to create relationships among tables before building report pages. Please refer to *Creating Relationships* for more information.

Connect via Data Connector

To begin, open Power BI Desktop and select **Get Data**.

Find the Senturus Analytics Connector in the list.

Click **Connect**.

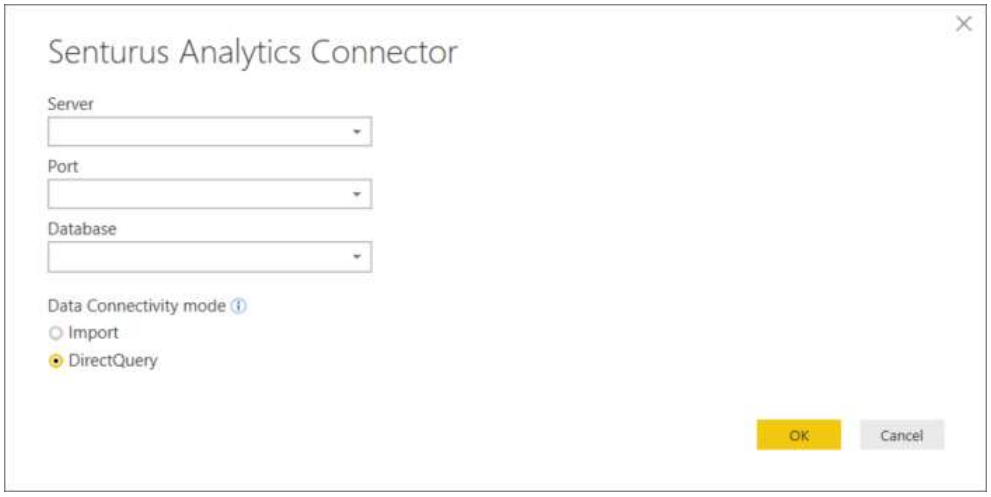


Power BI will display a dialog where you can select the Server, Port, Database, and Data Connectivity modes.

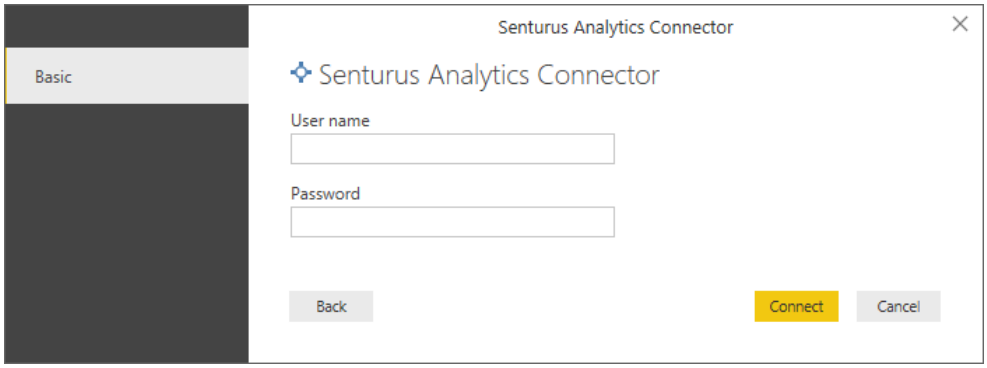
The Server, Port and Database lists are pulled from the configuration information in the DataSourceConfig.xml file under the server's Analytics Connector installation folder.

For the **Data Connectivity** mode, select *DirectQuery*.

Click **OK**.



An authentication dialog will appear. Type in your Cognos credentials, then click **Connect**.



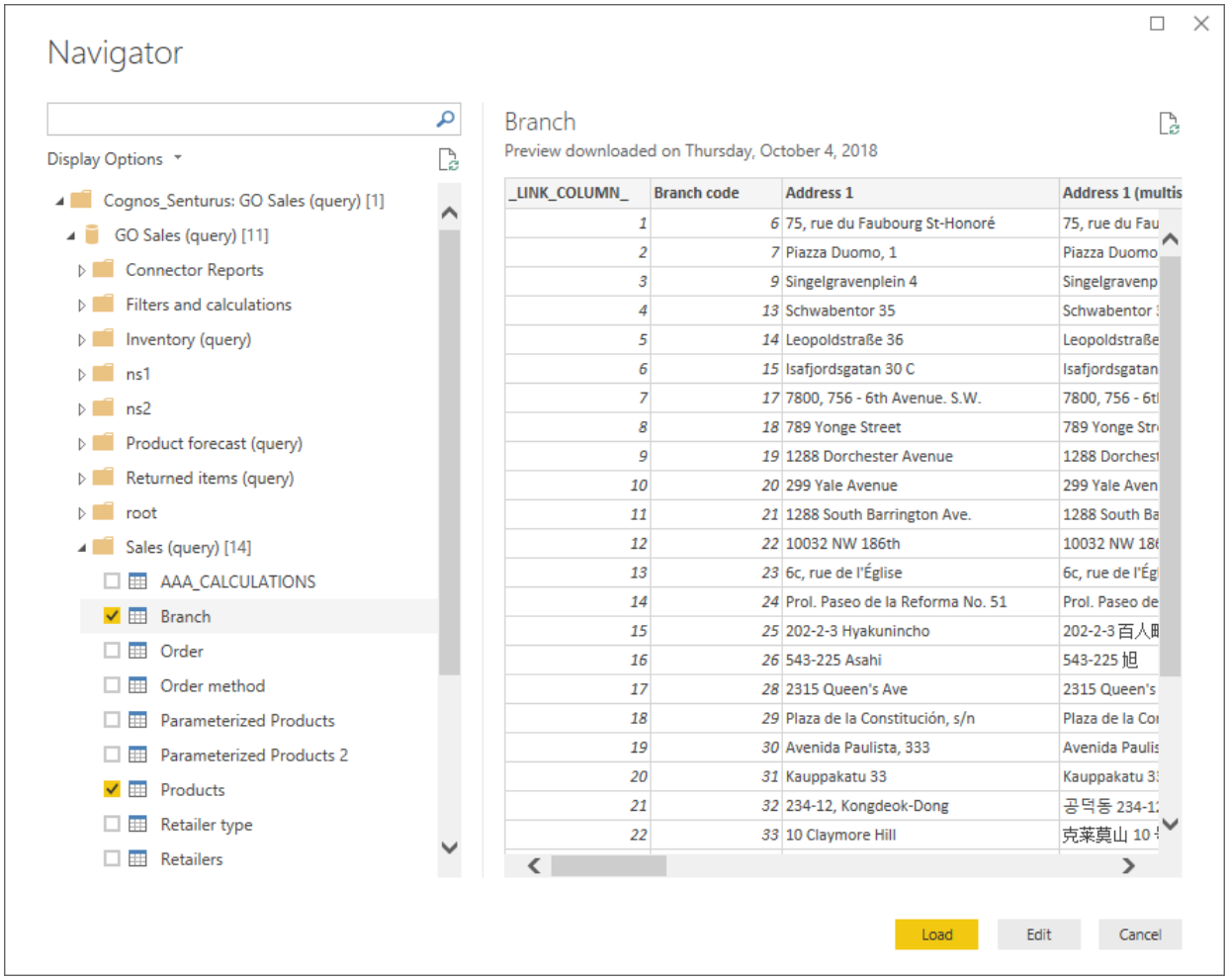
NOTE: Power BI desktop will securely store these credentials and skip this dialog if you select the same Server and Database in a future connection.

To manage or delete the stored authentication information, go to **File → Options and settings → Data source settings**.

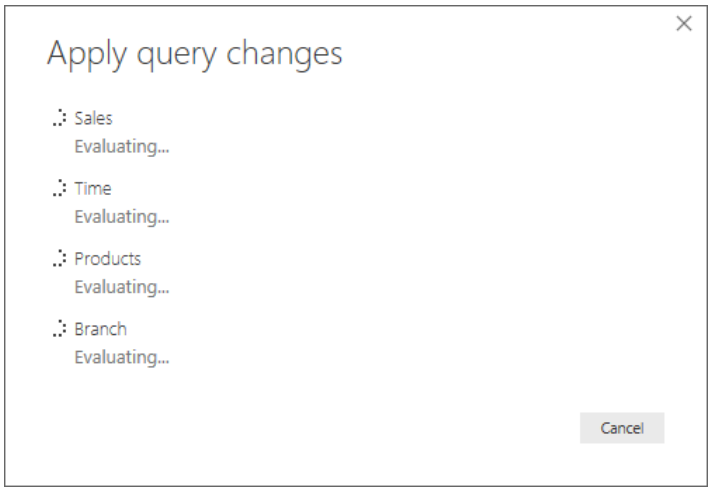
Once connected to the Senturus Analytics Connector data source, the **Navigator** window will display.

Select the tables you wish to include.

Click **Load**.



Power BI will then inspect the selected tables for column information.



When complete, navigate to the Relationships view and creaste relationships among tables.

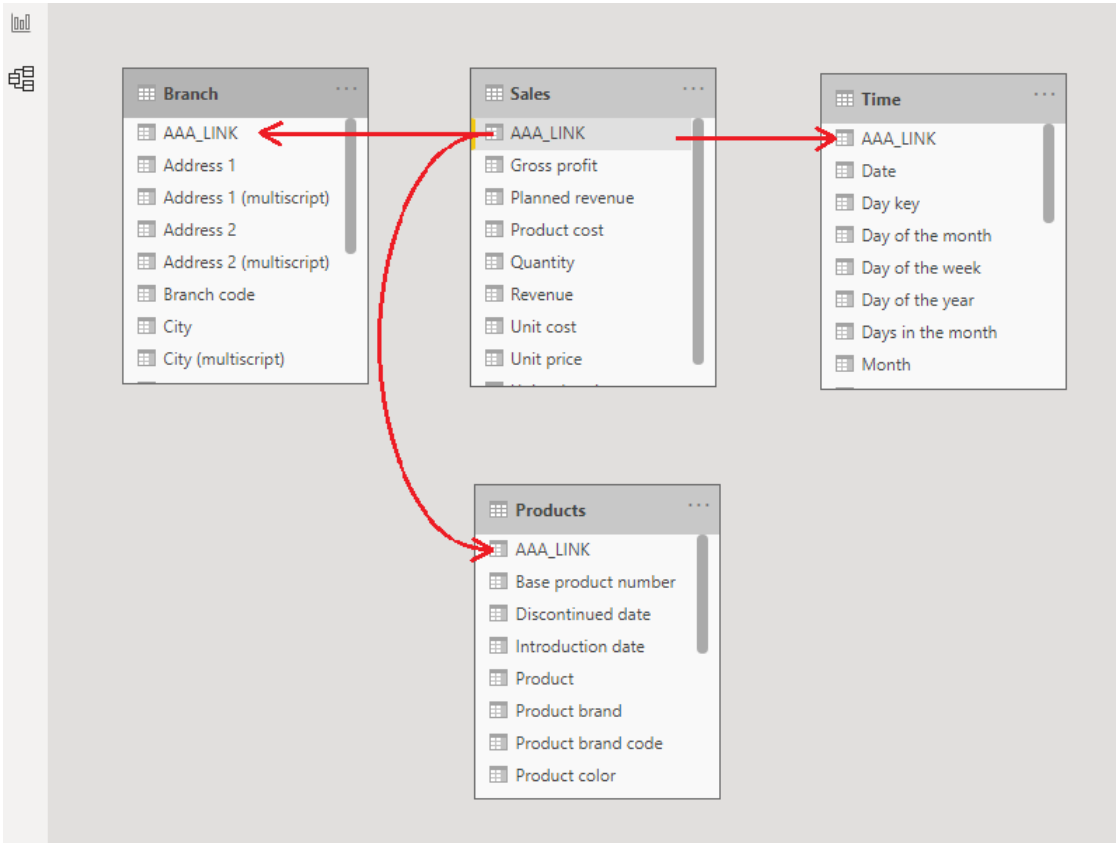
Creating Relationships

Power BI requires relationships between the tables, but Cognos typically does not present primary key and foreign keys at the business user view.

The Analytics Connector injects AAA_LINK columns into the tables so you can build relationships between the tables. The Analytics Connector ignores these columns when passing queries to Cognos to execute.

NOTE: It is very important that you create the relationships as described so that Power BI sends the appropriate queries to Cognos!

Start by dragging the *AAA_LINK* column in the fact table to the *AAA_LINK* table in the dimension table.



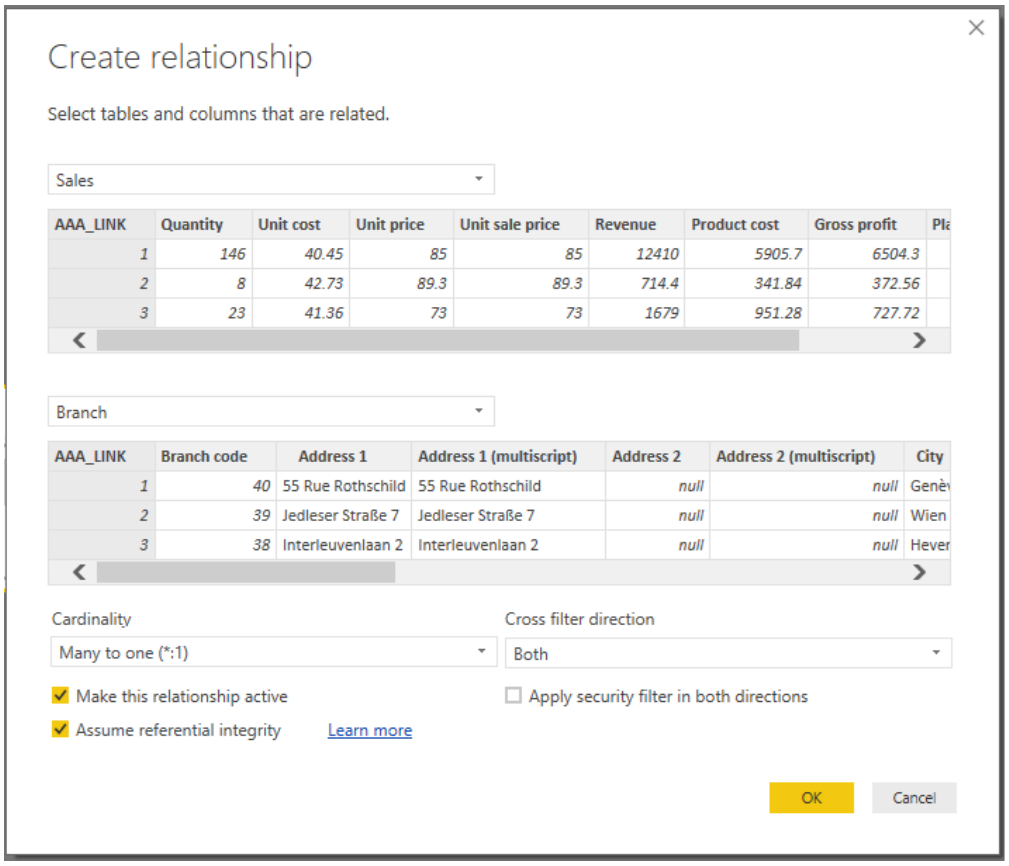
In the **Create Relationship** dialog, make sure the cardinality is set *Many to one (*:1).*

Change the Cross filter direction to *Both*.

Check the box next to *Assume referential integrity*.

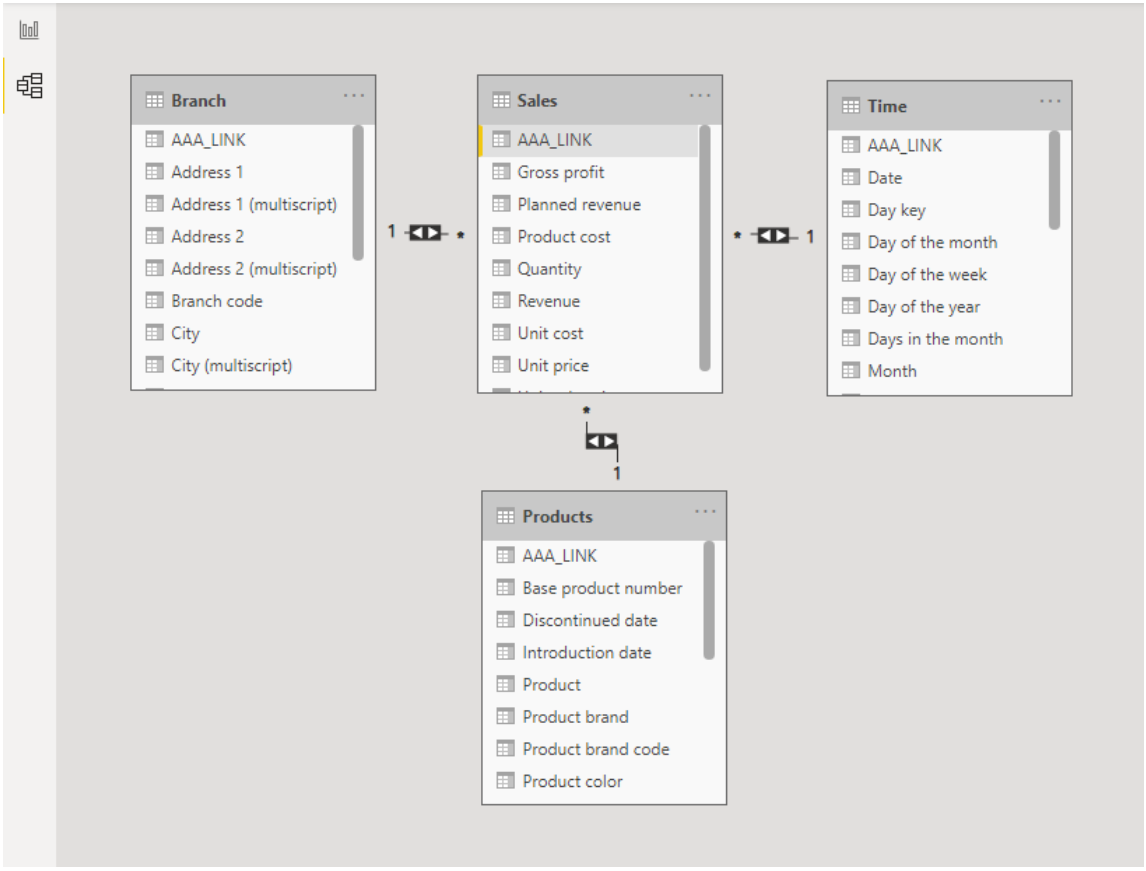
Click **OK**.

Hint: You don't have to wait Power BI to populate preview contents before clicking OK button. And if Power BI pops up an error dialog because it does not finish probing two tables yet, you can simply dismiss and ignore that error, as Power BI will create relationships as you specified anyway.



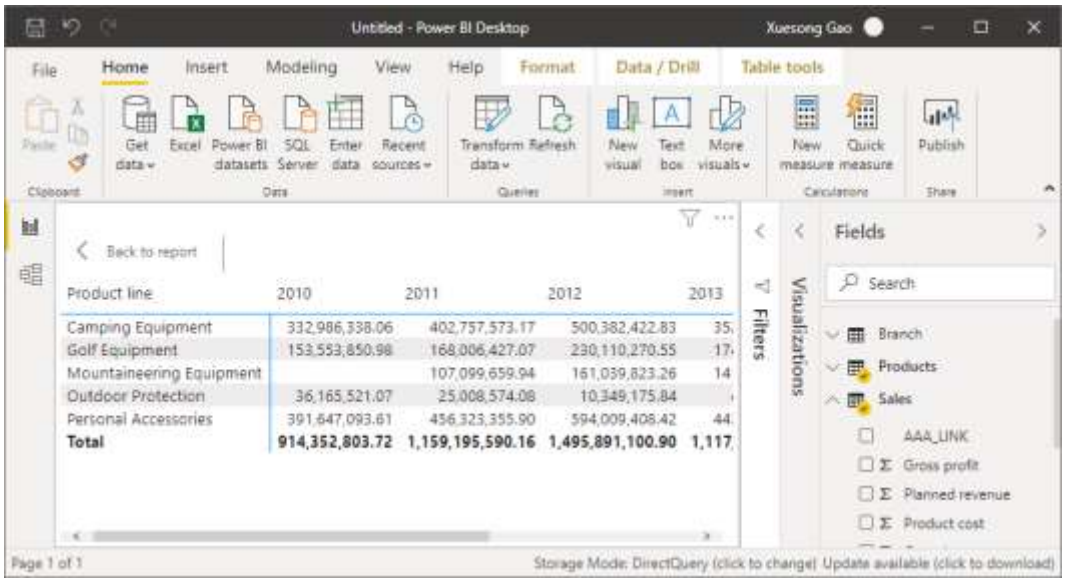
Repeat these steps until the fact table has relationships to all dimension tables.

If you have multiple fact tables, create a relationship to one or more of the dimension tables. The key is that the other fact tables are joined to the model and not isolated. Redudant relationships will be created as inactive relationships and are not necessary.



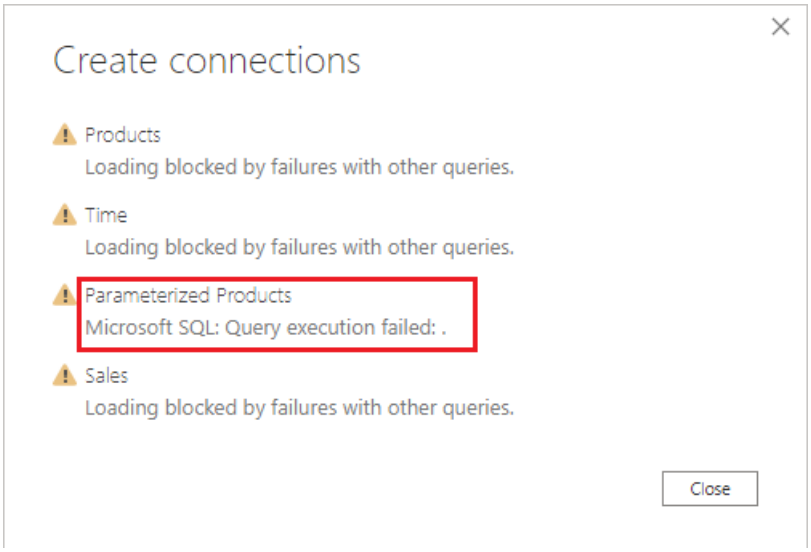
Change to the **Report** view.

Use the columns under the **Fields** tab to create your report.



Cognos Parameterized Tables and Reports

You cannot use parameterized tables (mapped from Cognos parameterized query subjects) or report mapped tables as regular tables. Power BI won't be able to import them if you select any parameterized tables or report tables:



When using parameterized query subjects in a Cognos report, or executing a pre-defined Cognos report with prompts, users will be prompted to enter parameter values. But you cannot provide parameter values in Power BI for regular database tables.

The answer is to write custom SQL and use **`OpenTable`** and **`RunReport`** functions to feed parameter values to Cognos.

Working with Parameterized Tables

The Analytics Connector supports working with parameterized tables in Power BI by using the `OpenTable` function, which lets you pass parameter values to Cognos.

Example:

```
select p."Product line", p."Product type", sum(s.Revenue) as Revenue,
sum(s.Quantity) as Qantity
from
  OpenTable (
    "Sales (query)"."Parameterized Products",
    "p_product line code"="[991, 992, 993, 994, 995]",
    "p_Product Line"="'Golf Equipment'",
```

```
    "p_Date"="{d '2010-01-01'}",
    "p_Datetime"="'2010-01-01 19:00:00'"
  ) as p
  join "Sales (query)"."Sales" as s on (p.AAA_LINK = s.AAA_LINK)
group by p."Product line", p."Product type"
```

Instead of selecting from table directly, you use OpenTable function (a tabular function) to invoke the parameterized table and join it to other tables. The first parameter `"schema name"."table name"` is the Cognos table name. This is followed by zero or more Cognos parameter name-value pairs separated by commas. For more information, please refer to *Function Syntax*.

To begin, open Power BI Desktop and select **Get Data -> SQL Server** from the ribbon. “SQL Server database” dialog pops up.

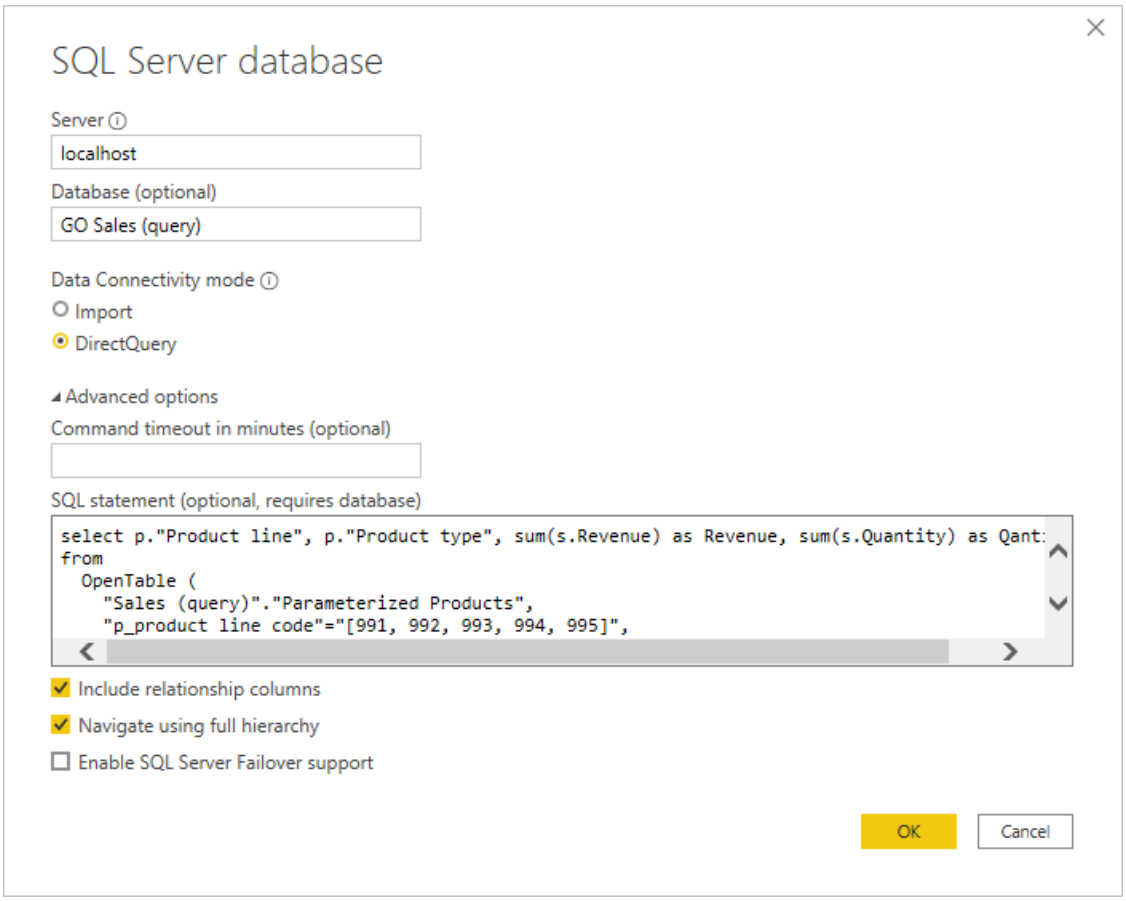
In **Server** input box, Type in hostname of IP of Analytics Connector server.

In **Database (optional)** input box, type in database name given by your Analytics Connector administrator when he/she mapped a Cognos package or data module to it. Please note, Database is mendentary in this case.

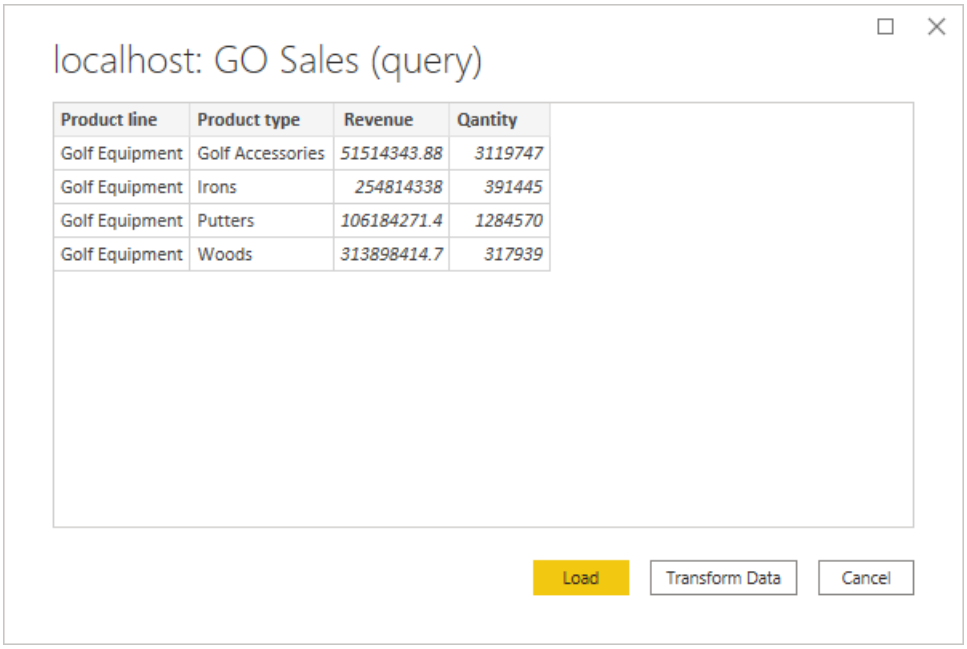
Check **DirectQuery** and (optionally) check “**Navigate using full hierarchy**”.

Expend “**Advanced options**”, put custom SQL in “**SQL Statement (optional, requires database)**” box.

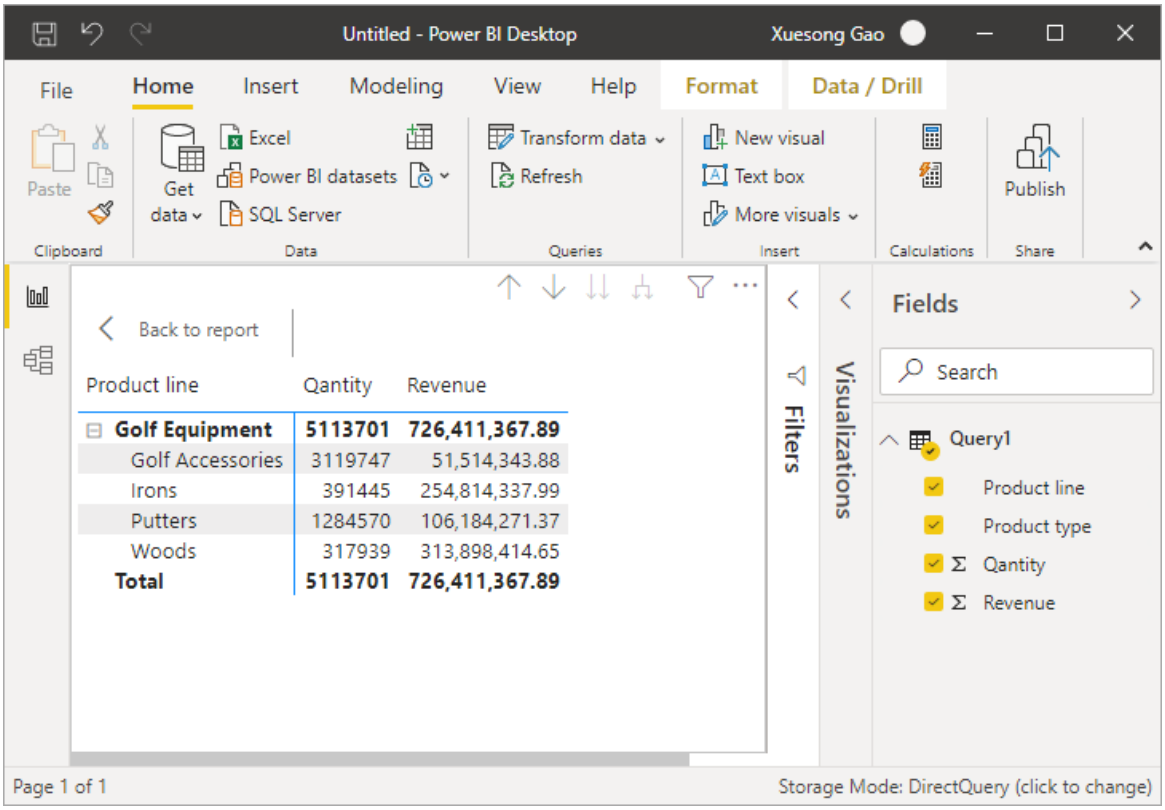
Click **OK**.



A data preview window will display. Click **Load**.



You can now use the columns from the **Fields** tab to build the report.



Working with Reports

The Analytics Connector supports working with Cognos reports in Power BI by using the `RunReport` function, which lets you pass parameter values to a Cognos report.

Analytics Connector only support list style Cognos reports, not crosstabs or charts.

Example:

```
Select *
From
    RunReport (
        "Connector Reports"."Parameter Report",
        "p_product line code"="[991,992,993,994,995]",
        "p_Product Line"="'Golf Equipment'",
        "p_Date"="'2010-01-01'",
        "p_Datetime"="2010-02-01 19:00:00"
    ) as "Report"
```

The first parameter `"schema name"."report name"` is the Cognos report. This is followed by zero or more Cognos parameter name-value pairs separated by commas. For more information of about RunReport function, please refer to *Function Syntax*.

To begin, open Power BI Desktop and select **Get Data -> SQL Server** from the ribbon. “SQL Server database” dialog pops up.

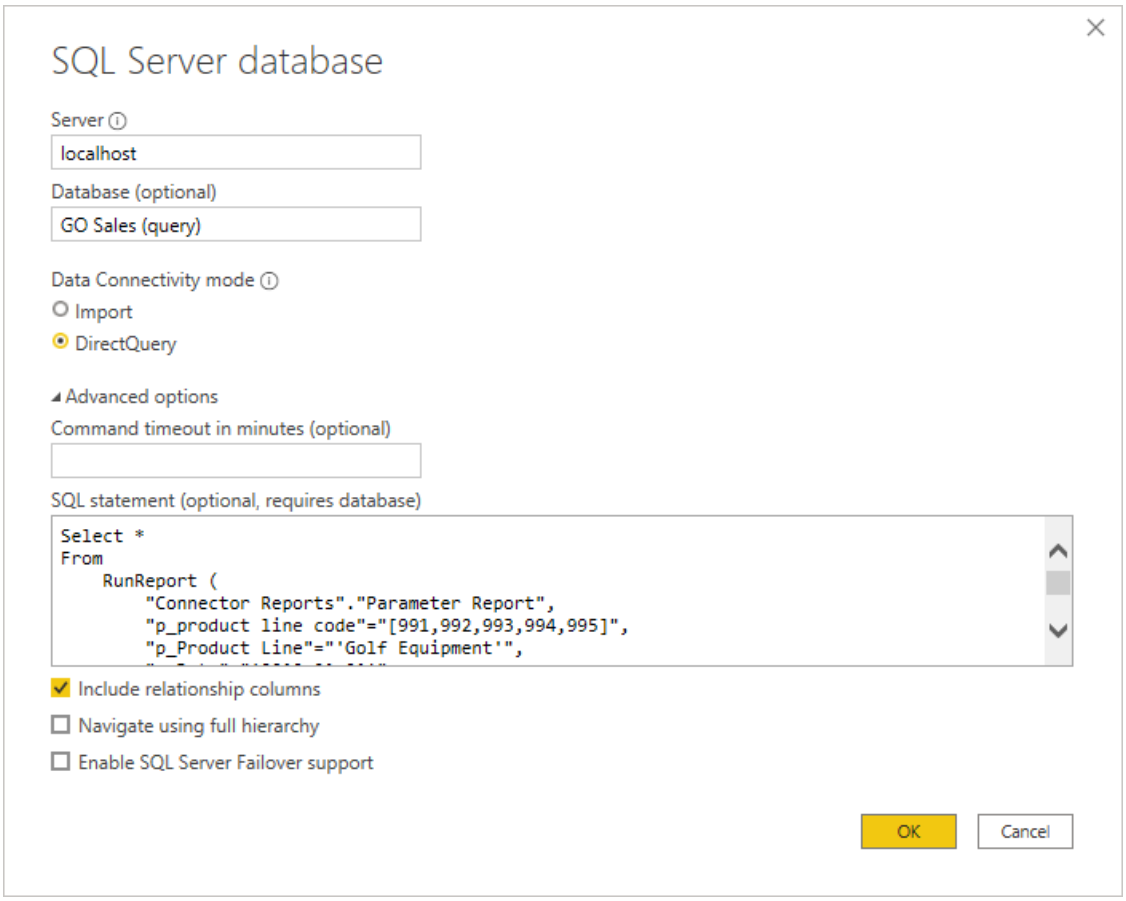
In **Server** input box, Type in hostname of IP of Analytics Connector server.

In **Database (optional)** input box, type in database name given by your Analytics Connector administrator when he/she mapped a Cognos package or data module to it. Please note, Database is mendentary in this case.

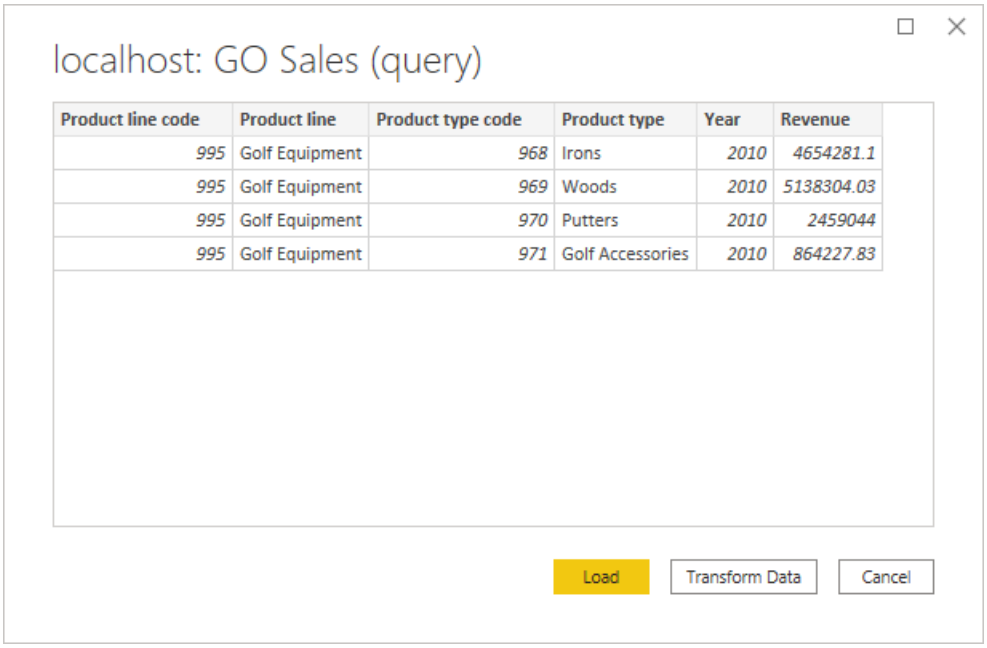
Check **DirectQuery** and (optionally) check “**Navigate using full hierarchy**”.

Expend “**Advanced options**”, put custom SQL in “**SQL Statement (optional, requires database)**” box.

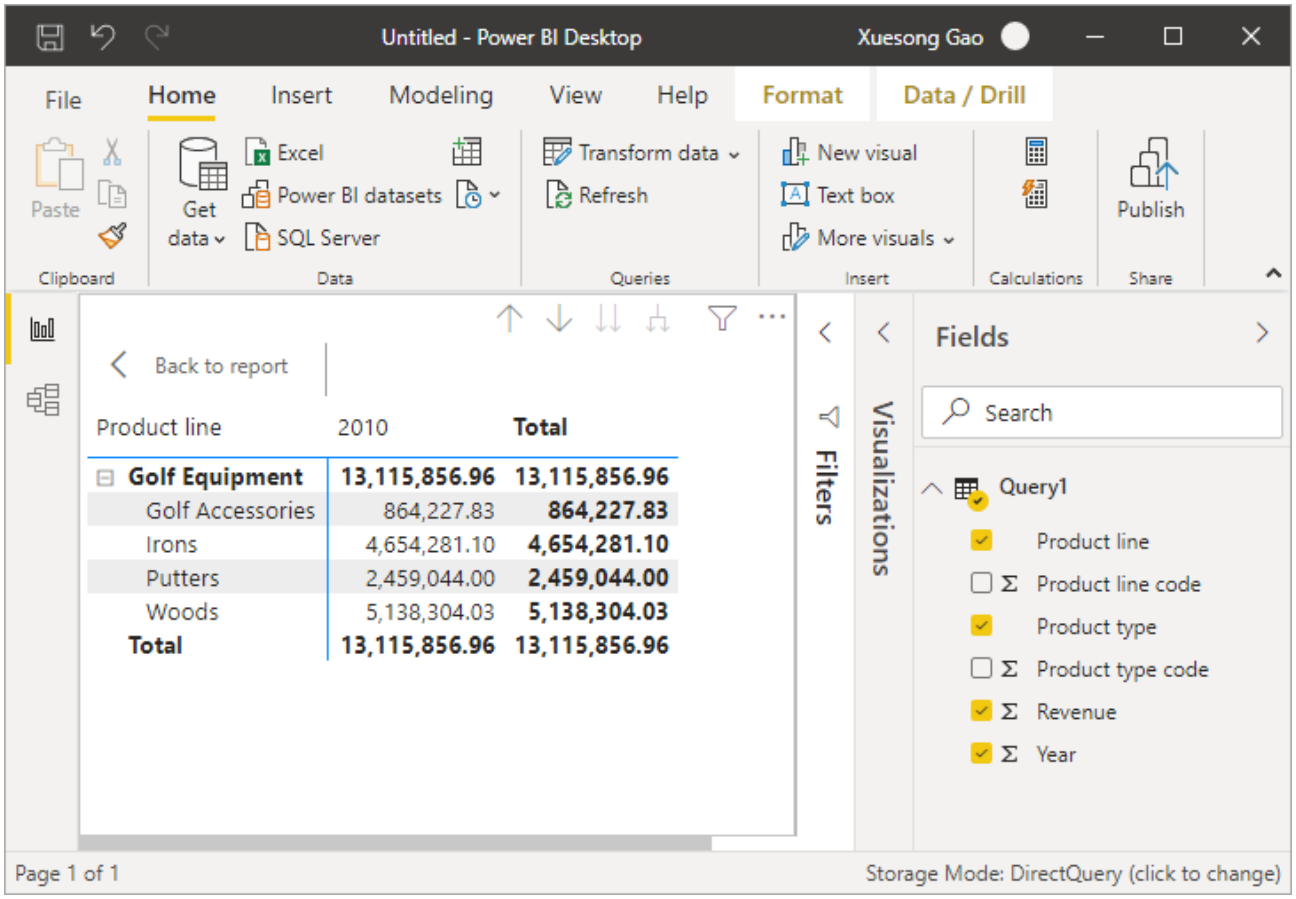
Click **OK**.



A data preview window will display. Click **Load**.

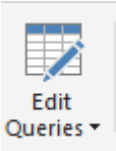


You can now use the columns from the **Fields** tab to build the report.



Working with Power BI Parameters

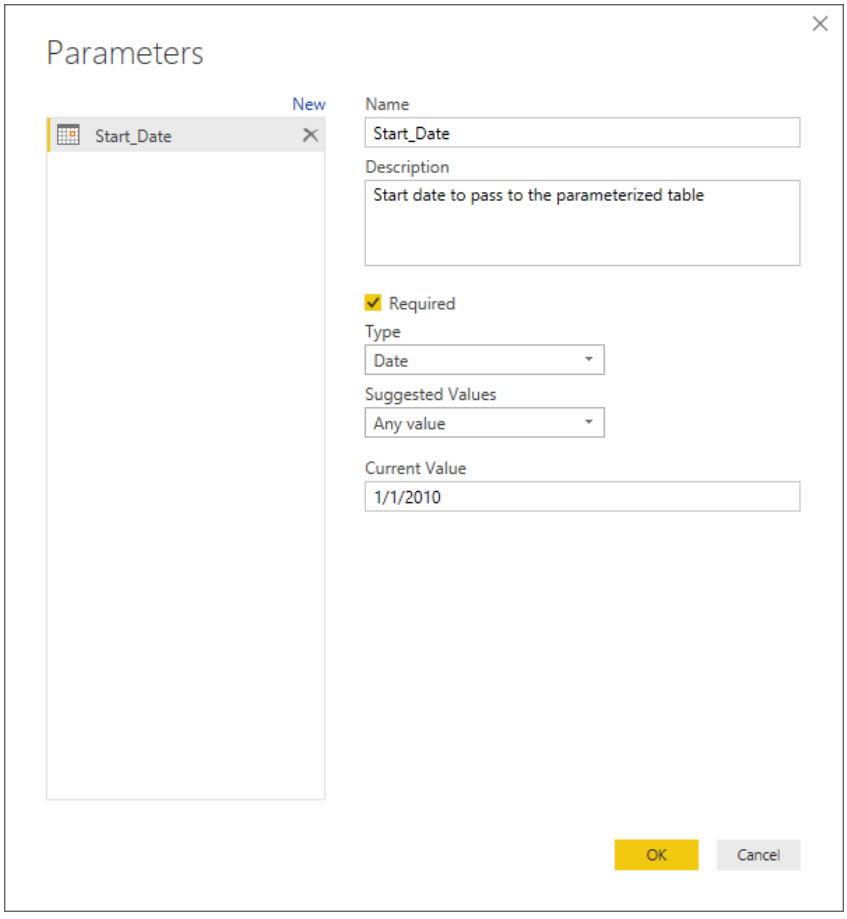
In your Power BI ODBC queries, you can replace the static values with Power BI parameters to create dynamic values that can be easily changed.



After configuring a *Parameterized Table* or *Report* query, click **Edit Queries** to bring up the Power Query Editor.

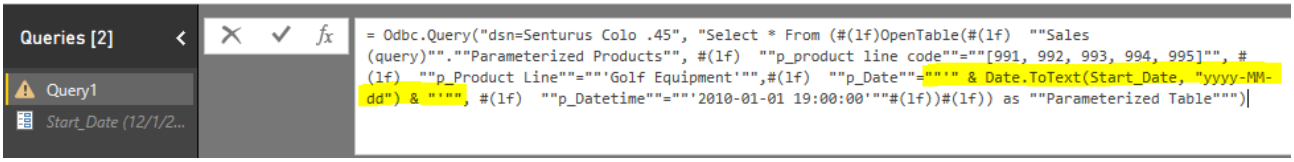
Click **Manage Parameters** to display the **Parameters** dialog.

Create a new parameter.

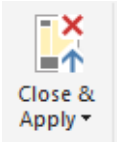


Next, edit the query to include the parameter in the SQL statement. You may need to format non-text data types using M language functions in order to append them to the statement.

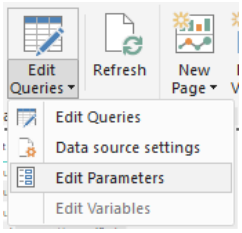
The below example formats a date parameter to text and appends it to the SQL statement.



NOTE: Double quotes must be escaped by using two sets of double quotes.



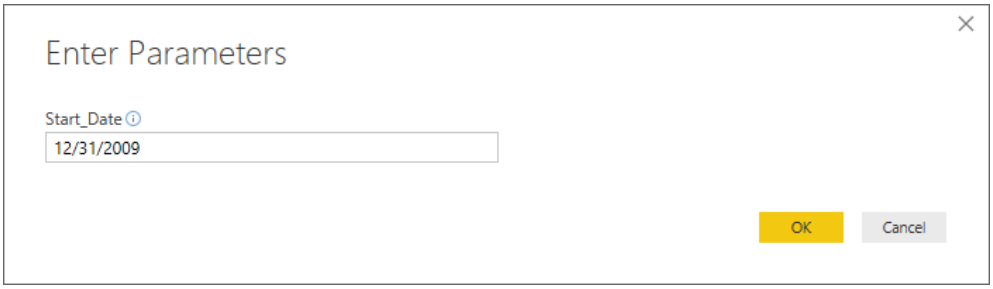
Click **Close & Apply**.



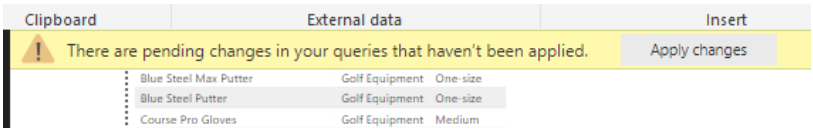
To change the Parameter value(s), select *Edit Parameters* under **Edit Queries**.

The **Enter Parameters** dialog will display with a list of configured parameters and their values.

Fill in new values and click **OK**.



Click **Apply Changes** if the data does not refresh.



Function Syntax

Both OpenTable and RunReport functions adopt same syntax.

Syntax

Both OpenTable and RunReport function adopt following syntax:
OpenTable ("Schema Name"."Table/Report Name"(, "Parameter"="value", …))

The first parameter is a table identifier, using standard SQL syntax. All identifiers are case insensitive and the schema name can be omitted if the table name is unique cross schemas.

Any additional parameters are name-value pairs, separated by commas. Name and values should be surrounded by double quotation marks.

Parameters

Use parameter name in place of "Parameter" for the Cognos connector.

Values

All values should follow SQL/ODBC standard.

Only plain string representation is allowed for numeric values. Formatted representations such as **1,200.00** with thousand separators, **(15)** for -15, or **50%** for .5 are not supported.

For string values, single quotation marks are used around the value. For example, 'Golf Equipment' or 'Sam''s Club'.

For date/time/timestamp values, use the ODBC standard format. For example {d '2010-01-01'} or ISO 8601 standard format, 'YYYY-MM-DD' for date, 'hh:mm:ss' for time and 'YYYY-MM-DD hh:mm:ss' for timestamp.

An array of values is passed using brackets. For example:

- "product line code"="[991,992,993,994,995]"

The syntax for a Range parameters is represented by a 2 element array. For example:

- "YearRange"="[,2019]" – All years less than or equal to 2019
- "YearRange"="[2015,]" – All years greater than or equal to 2015
- "YearRange"="[2015,2019]" – Years between 2015 and 2019

Character escaping

Certain characters must be escaped.

For single quotes within a string value, use two single quotes to escape a single quote inside the string.

Double quotes in parameter name and values need to be escaped with two double quotes.

Passing multiple values to a parameter

A Cognos parameter may accept multiple values, for example [Product line code] in (?p_Product line code?). To specify multiple values in your function, put multiple values in a pair of square brackets ([]) and separate each value by a comma (,) just like CSV format.

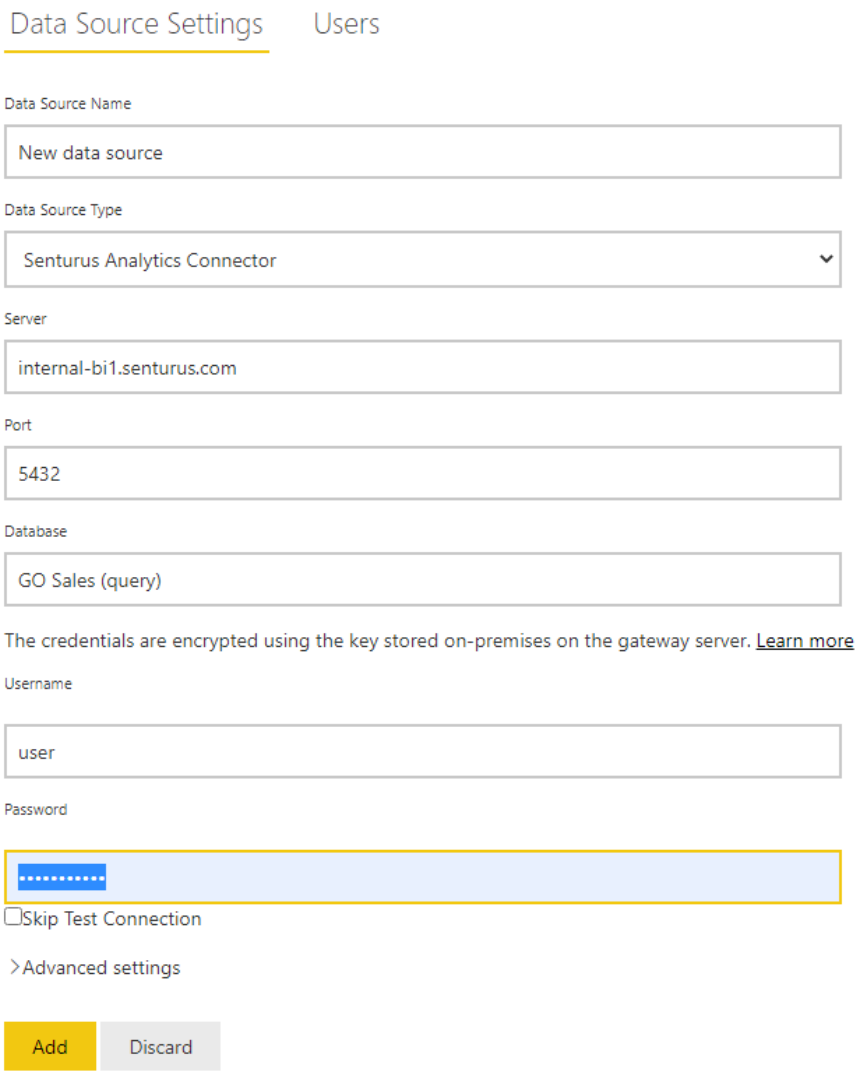
Publish to powerbi.com

After creating a Power BI report, you can publish it to powerbi.com and share it with others. Simply click on Publish button and select a destination you want to publish.

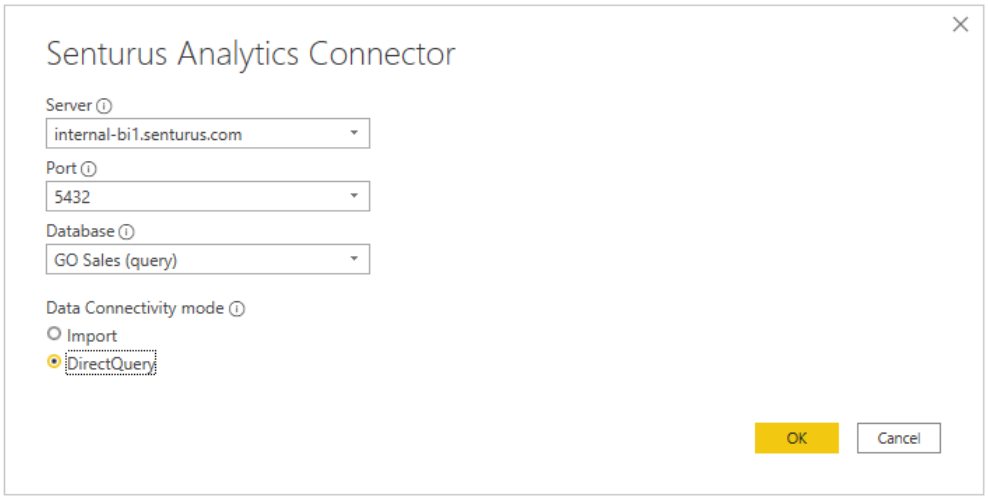
In order to allow powerbi.com access your on-premises data, you need to install and configure Power BI Gateway. If you use custom Data Connector in your Power BI report, you need to install Senturus Analytics Connector client on the computer where Power BI Gateway is running and copy custom connector file to it. For more information, please refer to Senturus Analytics Connector Installation Guide.

Once you published a report onto powerbi.com, you will see a report as well as a dataset (with the same name as your report) in the destination folder. But you cannot open that report on powerbi.com because the dataset is not associated with any data source yet.

First, you need to create a data source.



Please note, you need to type in the exat server, port and database here as you have in Power BI Desktop (case sensitive):



After that, you need to associate the new data source with your report dataset:

Finally, you can open your report and/or create new reports with your dataset.

Technical Reference

Supported ODBC Features

All identifiers (catalog, schema, table and column names) are case insensitive. This ODBC driver supports 15 Cognos data types. Refer to the data type mapping in following table.

Cognos Data Type	ODBC Data Type	Max Length/Precision	Note
int16	smallInt	5	
int32	integer	10	
int64	bigInt	19	
float32	real	7	
float64	double	15	
decimal	decimal	38	
character	char	8000	
characterLength16	varchar	8000	
nChar	nchar	4000	
nVarChar	nvarchar	4000	
date	date	10	
time	time	8	no milliseconds
datetime	timestamp	19	no milliseconds
textBlob	varchar	8000	mapped to varchar
unknown	varchar	8000	mapped to varchar

This driver has been tested against SQL Server, DB2 and Oracle databases (relational and DMR model). It may not support all data types/functions for other databases.

This driver only supports a) packages with one data source or b) packages with multiple data sources. However, all database connections are of the same type (e.g. SQL Server native connection). It may work for other packages, but some functions may fail.

Query features supported by this driver

Numeric, string, date/time, null literals

Cast/convert

Simple calculation (+, -, *, /, and %)

String concatenate\And, or, not logical operators

Comparison (<, >, =, <=, >=, !=, between, not between, in, not in, like, not like)

Is null, is not null

Case (if else) statement

Parenthesis

Sub queries

Functions supported by this driver

Aggregate function: AVERAGE, AVG, COUNT, MAX, MAXIMUM, MIN, MINIMUM, SUM, TOTAL.

Numeric functions: ABS, ACOS, ASIN, ATAN, CEILING, COS, COT, DEGREES, EXP, FLOOR, LOG, LOG10, MOD, POWER, RADIANS, ROUND, SIGN, SIN, SQRT, TAN, , TRUNC, TRUNCATE.

Date and timestamp functions: CURRENT_DATE, CURRENT_TIMESTAMP, DAY, DAYNAME, DAYOFMONTH, DAYOFWEEK, DAYOFYEAR, HOUR, MINUTE, MONTH, MONTHNAME, QUARTER, SECOND, TIMESTAMPADD, TIMESTAMPDIFF, WEEK, YEAR.

Character functions: ASCII, CHAR, CHAR_LENGTH, CHARACTER_LENGTH, CONCAT, CONTAINS, ENDS_WITH, LCASE, LCASE, LEFT, LEN, LENGTH, LOCATE, LTRIM, LTRUNC, REPLACE, RIGHT, RTRIM, RTRUNT, SPACE, STARTS_WITH, STRPOS, SUBSTR, SUBSTRING, TEXTCAT, TO_CHAR, TO_DATE, TO_TIMESTAMP, TRIM, UCASE, UCASE.

Others: CONVERT, IFNULL, COALESCE.

CONTACT US

The latest version of the Analytics Connector along with related documentation and contact information can be found at https://www.senturus.net/connector-download/.

If you have any additional questions, please contact us at CustomerSupport@senturus.com.