

Senturus Analytics Connector

User Guide

Cognos to Power BI

Contents

Overview	3
Connecting to Analytics Connector Server	4
Power BI SQL Server Connector vs Custom Connector	4
Connecting via SQL Server Connector	5
Import vs DirectQuery Mode	9
Differences	9
Import Tables	10
Importing Queries	11
Creating Relationships	13
Join DirectQuery Tables	13
Join Import Tables	15
Join DirectQuery and Import Tables	15
Join Tables without Key Columns	16
Cognos Parameterized Tables and Reports	17
Working with Parameterized Tables	17
Working with Reports	21
Working with Power BI Parameters	24
Function Syntax	26
Publish to powerbi.com	28
Legacy Documentation	34
Install the Custom Connector	34
Connect via Custom Connector	35
Contact Us	38

Overview

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

Please refer to the *Senturus Analytics Connector Installation and Configuration 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.

Connecting to Analytics Connector Server

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

You can connect to the Analytics Connector Server using either the Power BI built-in SQL Server database connector or the Senturus custom data connector.

Senturus **highly** recommends adopting the Power BI built-in SQL Server connector. See the [Legacy Documentation](#) section for information on the custom connector.

Power BI SQL Server Connector vs Custom Connector

Starting from version 5.0, you can connect from Power BI Desktop and Power BI Gateway to the Analytics Connector via a 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.

Compared to the custom data connector, the SQL Server connector gives you the following advantages:

1. No need to install Analytics Connector ODBC client¹.
2. No need to copy custom Data Connector (.mez files).
3. Use Windows authentication from Power BI Desktop to Analytics Connector server to Cognos.²
4. More Power BI functions are supported, e.g. [query cancellation](#).

See the [Legacy Documentation](#) section for information on the custom data connector.

¹ You still need to install SQL Server Native Client 11 or above on your client computer.

² Both Power BI Desktop and Cognos server must be secured by the same Active Directory service.

Conecting via SQL Server Connector³

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

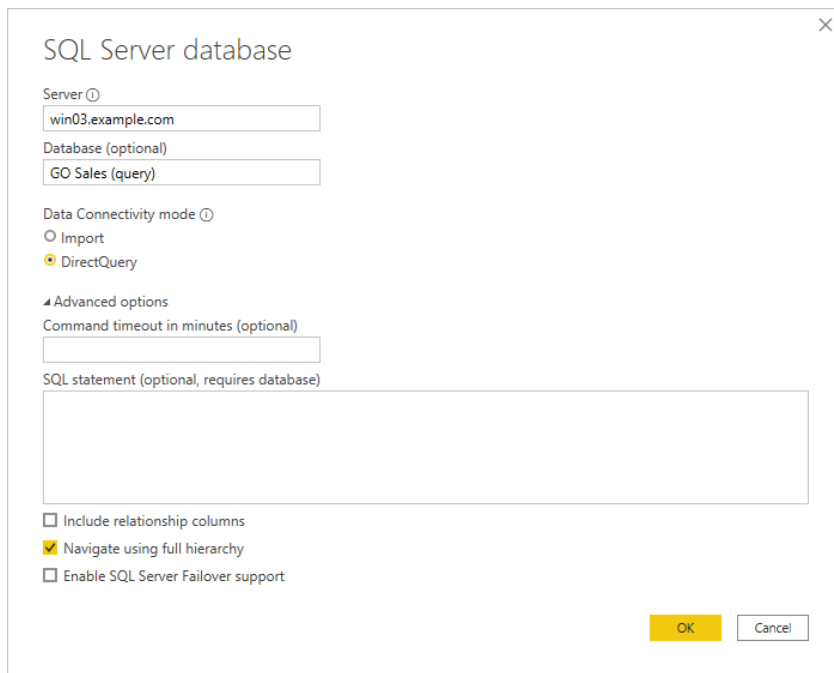
In **Server** input box, Type in hostname of IP of Analytics Connector server.

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.

Check **DirectQuery** and (optionally) uncheck “Include relationship columns” and check **“Navigate using full hierarchy”**.

The Senturus Analytics Connector supports both Import and DirectQuery modes. For the sake of simplicity, we will only use DirectQuery in this section. For more information regarding when and how to use Import vs DirectQuery mode, please refer to the next section.

Then click on **OK** button.



SQL Server database

Server

Database (optional)

Data Connectivity mode Import DirectQuery

Advanced options

Command timeout in minutes (optional)

SQL statement (optional, requires database)

Include relationship columns

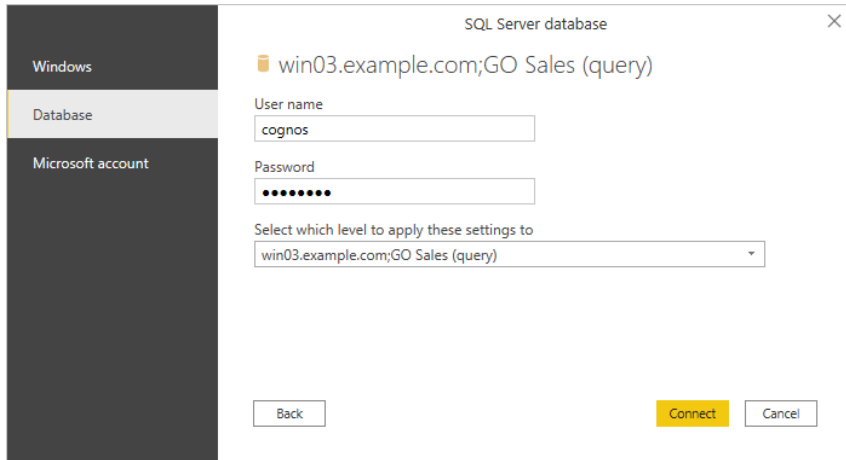
Navigate using full hierarchy

Enable SQL Server Failover support

OK Cancel

³ Make sure you have installed the SQL Server Native Client 11 or above in order to use SQL database connection. The download can be found here: <https://www.microsoft.com/en-us/download/details.aspx?id=50402>

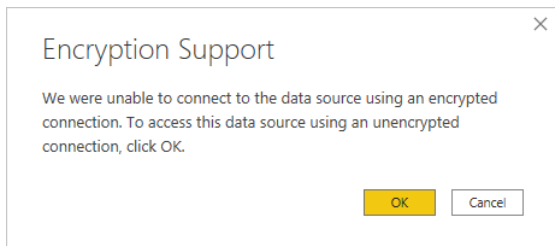
You can use either **Windows** authentication⁴ or **Database** authentication (by providing your Cognos login user name and password).



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 if SSL is not enabled on the Analytics Connector server. Click **OK** to dismiss this message.



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

1. Each namespace is mapped to a schema. You will see some system schemas when connecting to a real SQL server database.

⁴ You can only use Windows authentication when your Administrators have successfully configured Windows Authentication support on Senturus Analytics Connector server.

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 information.
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 the 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 a Cognos data module, you will see following objects in the Navigator panel:

1. Only one schema with the same name as your data module. You will see some system schemas when connecting to a real SQL server database.
2. Each table in a 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. Additionally, you can uncheck the **Display Options -> Enable data previews** option to skip previews.*

Navigator

Display Options ▾

- Return
- root
- Sales (query) [15]
 - Branch
 - CALCULATIONS
 - Order
 - Order method
 - Parameterized Products
 - Parameterized Products 2
 - Products
 - Retailer type
 - Retailers
 - Sales
 - Sales staff
 - Time
 - Time (close date)
 - Time (ship date)
 - User
- Sales target (query)
- sys

Products

AAA_LINK	Product line code	Product line	Product type cod
1	994	Outdoor Protection	
2	991	Camping Equipment	
3	993	Personal Accessories	
4	993	Personal Accessories	
5	993	Personal Accessories	
6	993	Personal Accessories	
7	993	Personal Accessories	
8	994	Outdoor Protection	
9	991	Camping Equipment	
10	991	Camping Equipment	
11	994	Outdoor Protection	
12	994	Outdoor Protection	
13	994	Outdoor Protection	
14	994	Outdoor Protection	
15	994	Outdoor Protection	
16	991	Camping Equipment	
17	991	Camping Equipment	
18	991	Camping Equipment	
19	991	Camping Equipment	
20	995	Golf Equipment	
21	994	Outdoor Protection	
22	995	Golf Equipment	
23	993	Personal Accessories	

Select Related Tables Load Transform Data Cancel

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

Apply query changes

- ∴ Sales
Evaluating...
- ∴ Time
Evaluating...
- ∴ Products
Evaluating...
- ∴ Branch
Evaluating...

Cancel

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.

Import vs DirectQuery Mode

Differences

The Senturus Analytics Connector supports both Import and DirectQuery modes. The bottom line is the Senturus Analytics Connector sever has no clue if a query is sent for Import or DirectQuery.

Basic Difference

The basic difference between Import and DirectQuery mode is, Import mode extracts the data from your database and pulls it into the Power BI report, while DirectQuery leaves the data in your database and sends queries (usually grouped and aggregated) to pull the information as needed.

Other things to note. Import gives you the full suite of transformation and data manipulation in the Desktop, but there is a 1 GB limit to the report if you plan on publishing to the PBI Service. DirectQuery leaves the data in your database, so the data will always be live/up-to-date and there is no need to refresh.

Performance

Import is usually faster than DirectQuery, and it reduces your database workload because everything is already imported into a Power BI report but this is not always true. Results vary depending on your live database performance and the size/structure of imported data.

Security

When you use DirectQuery, the Senturus Analytics Connector server logs into Cognos on behalf of you, so all Cognos securities will be enforced, for example package/report access security, and row level security built-in your Cognos model and reports.

When you use Import mode, Cognos securities will be applied only when you import the data. Once data is imported into Power BI, they are disconnected from Analytics Connector/Cognos, so all data will be open to anyone that has access to this Power BI report until you set up row-level security in your Power BI report.

Relationship Setup

Whether your Cognos models (FM packages, data modules, cubes) expose primary and foreign keys also affects your options of which one to use. If there are no join columns between tables (derived Cognos query subjets, hierarchies, etc.), then you cannot join imported tables in Power BI. We will dicuss this in detail later. There is some workaround though, please refer to Join Tables without Key Columns for more information.

Other

The last difference is, Import mode is only available when you are using the SQL Server connector⁵.

Composite Mode

Starting from the December 2020 release, Power BI supports composite models, which means you can mix DirectQuery and Import tables/queries in one report. You still need to make sure PK/FK are available when joining from/to an imported table/query – otherwise, they have to be DirectQuery tables.

Import Tables

We already showed you how to connect tables via DirectQuery mode in the previous section. Importing tables follows the exact same steps, except you need to check the “**Import**” option.

SQL Server database

Server ⓘ
localhost

Database (optional)

Data Connectivity mode ⓘ
 Import
 DirectQuery

Advanced options
Command timeout in minutes (optional)

SQL statement (optional, requires database)

Include relationship columns
 Navigate using full hierarchy
 Enable SQL Server Failover support

OK Cancel

⁵ You can setup an ODBC DSN using SQL Server native client 11.0+ or Senturus Analytics Connector ODBC client, then import tables/queries using Power BI ODBC connector. This is how it works in previous release before V5. This will still work, but we recommend you use SQL Server Database connector in Import mode for all new reports.

Importing Queries

You can also specify a query to import instead of individual tables.

Within the **“SQL Server database”** page, type in **“Database”** (it’s now mandatory when importing a query), and put a valid select script in **“SQL statement (optional, requires database)”** box. You can use standard SQL select statements here, including filter, group, aggregates, functions, even sub-queries. We cannot guarantee it will support all select statements, so keep it as simple as possible.

The schema, tables and columns are derived from your Cognos model, as you can see during the importing tables page. Also, please use **AAA_LINK** to join your tables (see screenshot for example).

SQL Server database

Server

Database

Data Connectivity mode Import DirectQuery

Advanced options

Command timeout in minutes (optional)

SQL statement (optional, requires database)

```
select m.*, p."Product line", p."Product type", p."Product", y."Year", y."Quarter", y."Month"
from "great_outdoors_sales_en"."Measures" as m
join "great_outdoors_sales_en"."Products" as p on (m.AAA_LINK = p.AAA_LINK)
join "great_outdoors_sales_en"."Years" as y on (m.AAA_LINK = y.AAA_LINK)
where y."Year" between '2010' and '2013'
```

Include relationship columns
 Navigate using full hierarchy
 Enable SQL Server Failover support

OK Cancel

After clicking OK, this query will be sent to the Analytics Connector server and a preview page will display the result once data is returned. Click **“Load”** to import it into Power BI report or **“Transform Data”** to do more transformation before loading.

localhost: Great Outdoors Sales (cube)

AAA_LINK	Revenue	Product cost	Product plan	Branch expense plan	Quantity sold	Unit cost	Gross profit	Profit margin %	Rev
1	189953.61	104562.9	2818.444189	null	15795	6.62	85390.71	0.44953455	
2	240887.99	180255.98	3075.567182	null	27229	6.62	60632.01	0.25170209	
3	193228.47	106337.06	2856.811802	null	16063	6.62	86891.41	0.44968223	
4	176147.19	96764.54	2733.5905	null	14617	6.62	79382.65	0.45066089	
5	194226.84	107402.88	2952.939904	null	16224	6.62	86823.96	0.44702349	
6	291978.89	218930.02	4563.642187	null	33071	6.62	73048.87	0.25018545	
7	205243.79	113155.66	2749.58619	null	17093	6.62	92088.13	0.44867682	
8	206435.99	113857.38	2864.817838	null	17199	6.62	92578.61	0.44846158	
9	192593.94	105946.48	2704.112935	null	16004	6.62	86647.46	0.44989713	
10	225718.45	125018.7	3106.758609	null	18885	6.62	100699.75	0.4461299	
11	208005.52	114459.8	2861.678251	null	17290	6.62	93545.72	0.4497271	
12	187243.54	103020.44	2572.886108	null	15562	6.62	84223.1	0.4498051	
13	305582.64	154801.4553	5186.517656	null	34614	4.47222093	151275.36	0.49503912	
14	237334.98	129368.04	3165.185872	null	19542	6.62	107966.94	0.45491373	
15	257161.83	130341.0198	3716.975098	null	29109	4.47768799	127279.05	0.49493757	
16	226707.92	123767.52	3316.171447	null	18696	6.62	102940.4	0.45406618	
17	225011.43	122754.66	2811.693524	null	18543	6.62	102256.77	0.45445145	
18	224694.79	122827.48	3028.743411	null	18554	6.62	101867.31	0.45335858	
19	232952.46	126859.06	3223.113959	null	19163	6.62	106093.4	0.4554294	
20	217680.62	118471.52	2766.177037	null	17896	6.62	99209.1	0.45575532	

i The data in the preview has been truncated due to size limits.

Load Transform Data Cancel

Once the query is imported, you click on the “Data” tab to view the data and make more transformation as needed.

Untitled - Power BI Desktop

File Home Help Table tools Column tools

Name: AAA_LINK Format: Whole number Summarization: Sum Data type: Whole number Data category: Uncategorized

AAA_LINK	Revenue	Product cost	Product plan	Branch expense plan	Quantity sold	Unit cost	Gross profit
4340	122888.25	49425.32	17632.39257359		25217	1.96	
4341	131721.45	53021.92	18731.37141865		27052	1.96	
4342	139216.45	56506.8	19186.60748764		28830	1.96	
4343	135192.55	54499.76	18462.21307812		27806	1.96	
4344	120143.55	48504.12	16898.69700112		24747	1.96	
4345	43300	16973.6	5913.81305308		8660	1.96	
4346	37545	14717.64	4487.05768334		7509	1.96	
4347	30480	11948.16	4183.9224558		6096	1.96	

Creating Relationships

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

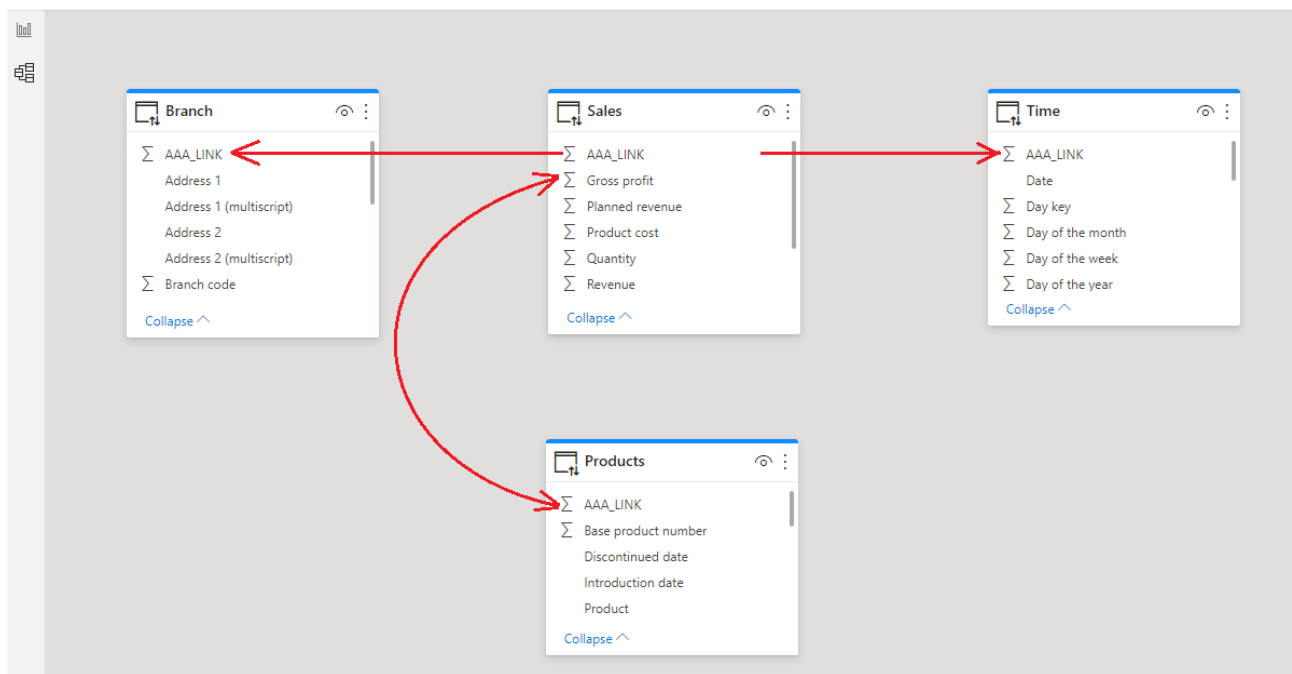
The Analytics Connector injects **AAA_LINK** columns into the tables so you can build relationships between them even when there are no PK/FK columns. The Analytics Connector ignores joins formed by **AAA_LINK** columns when passing queries to Cognos for execution.

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

Join DirectQuery Tables

When joining DirectQuery tables, always use **AAA_LINK** columns because you want to let Cognos decide the correct relationship. Even when you do have business PK/FK columns, please still join them by **AAA_LINK** columns as this will guarantee correct results and better performance.

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)*.

Check the box next to *Assume referential integrity*.

Click **OK**.

Hint: You don't have to wait for Power BI to populate preview contents before clicking the OK button. If Power BI shows an error dialog because it hasn't finished probing the tables yet, you can simply dismiss and ignore the error, as Power BI will create relationships as you specified anyway.

Create relationship

Select tables and columns that are related.

Sales

AAA_LINK	Quantity	Unit cost	Unit price	Unit sale price	Revenue	Product cost	Gross profit	Price
1	146	40.45	85	85	12410	5905.7	6504.3	
2	8	42.73	89.3	89.3	714.4	341.84	372.56	
3	23	41.36	73	73	1679	951.28	727.72	

Branch

AAA_LINK	Branch code	Address 1	Address 1 (multiscript)	Address 2	Address 2 (multiscript)	City
1	40	55 Rue Rothschild	55 Rue Rothschild	null	null	Genève
2	39	Jedleser Straße 7	Jedleser Straße 7	null	null	Wien
3	38	Interleuvenlaan 2	Interleuvenlaan 2	null	null	Heverlee

Cardinality: Many to one (*:1)

Cross filter direction: Both

Make this relationship active

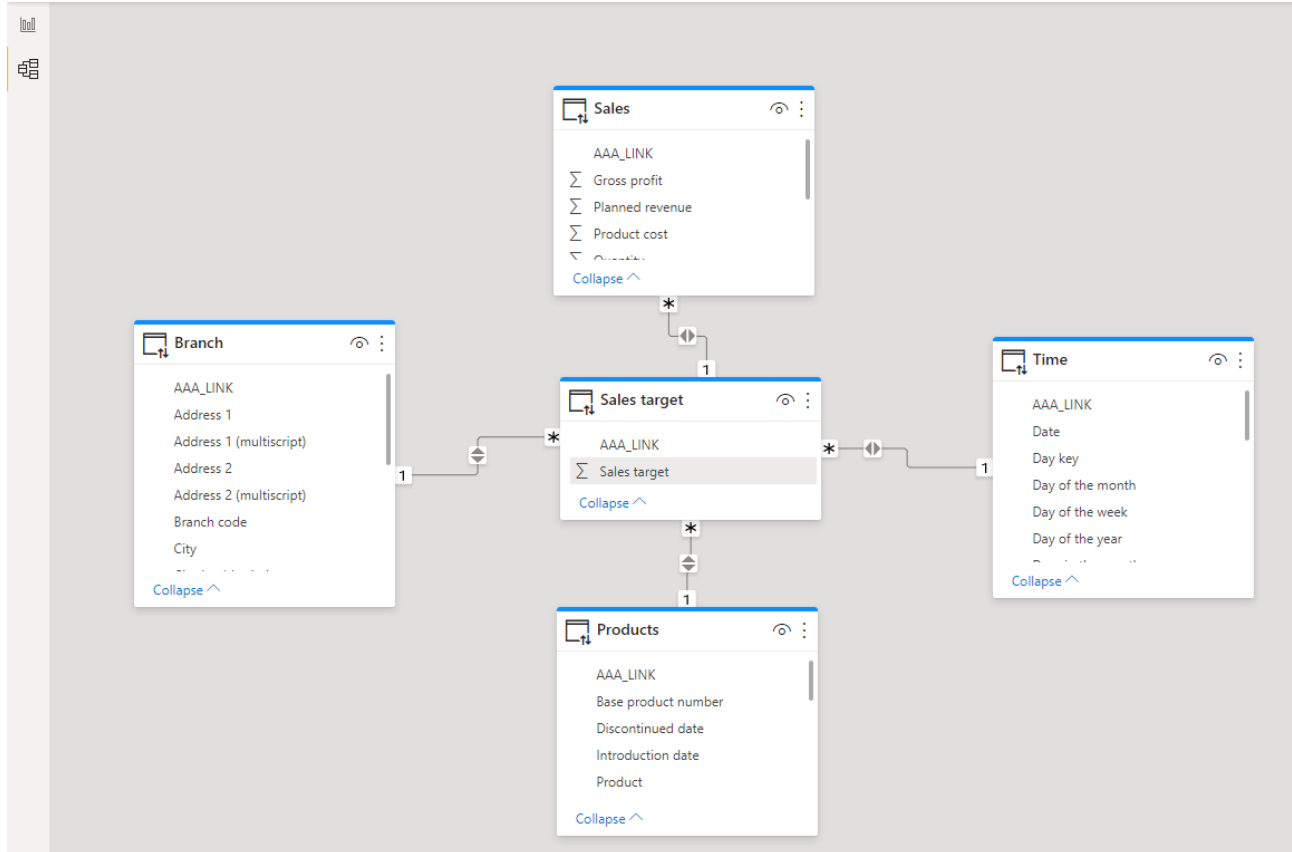
Apply security filter in both directions

Assume referential integrity [Learn more](#)

OK Cancel

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

If you have multiple fact tables, chain these fact tables based on granularity and link the highest level fact table to dimension tables, as shown in the diagram below.



After creating relationships, change to the **Report** view and create your report.

Join Import Tables

When joining import tables, always use your business PK/FK columns. Using **AAA_LINK** to join import tables will yield incorrect results. Why? Because once the table/query data is loaded into Power BI, they are stored as different tables within Power BI. If you join tables by **AAA_LINK** (which are simply sequential numbers in each table/query), the numbers will be wrong.

Join DirectQuery and Import Tables

When joining between DirectQuery and Import tables, always use your business PK/FK columns too. Power BI will query DirectQuery first then join the result to the import table using the relationship you defined, so these relationships must be joined by business PK/FK columns.

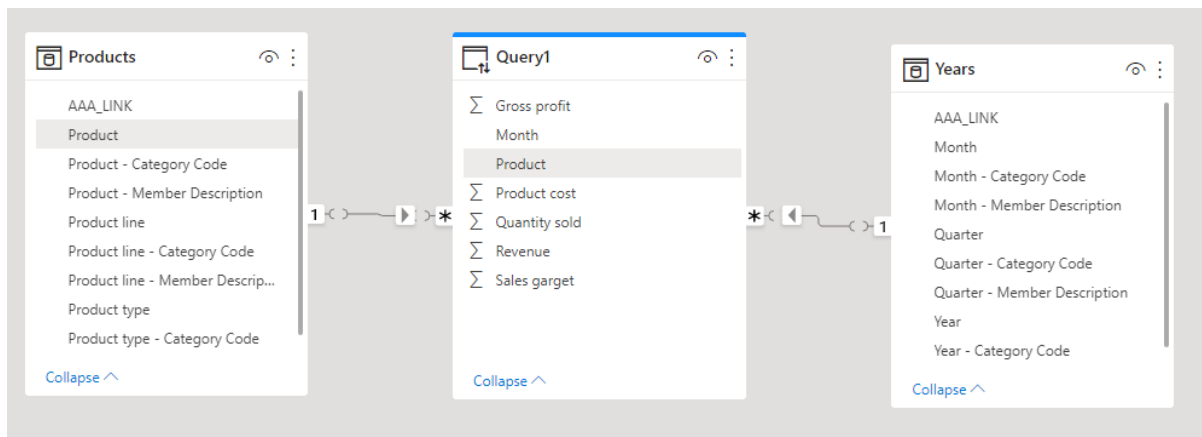
Join Tables without Key Columns

What if your table does not have business PK/FK columns and you still need to join it with an import table, or this table itself is an import table? You can borrow ones from other tables using a query.

You can import a query, and “borrow” FK columns from other tables by using a select statement like below:

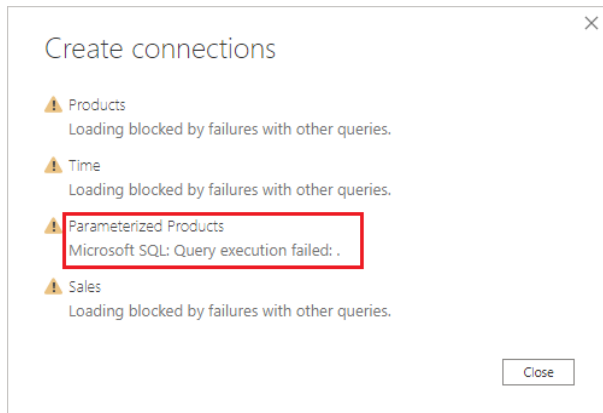
```
select p."Product",
       y."Month",
       sum(m."Quantity sold") as "Quantity sold",
       sum(m."Revenue") as "Revenue",
       sum(m."Product cost") as "Product cost",
       sum(m."Gross profit") as "Gross profit",
       sum(m."Sales target") as "Sales garget"
from "great_outdoors_sales_en"."Measures" as m
     join "great_outdoors_sales_en"."Products" as p on (m.AAA_LINK = p.AAA_LINK)
     join "great_outdoors_sales_en"."Years" as y on (m.AAA_LINK = y.AAA_LINK)
where y."Year" = '2010'
group by p."Product",
         y."Month"
```

Then you can join this query to other import tables like below:



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 will not 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 Quantity
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 a table directly, you use the 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 has mapped a Cognos package or data module to it. Please note, Database is mandatory in this case.

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

Expand "**Advanced options**", put custom SQL in "**SQL Statement (optional, requires database)**" box.

Click **OK**.

SQL Server database

Server

Database (optional)

Data Connectivity mode DirectQuery

Advanced options

Command timeout in minutes (optional)

SQL statement (optional, requires database)

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

Include relationship columns

Navigate using full hierarchy

Enable SQL Server Failover support

OK Cancel

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

localhost: GO Sales (query)

Product line	Product type	Revenue	Qantity
Golf Equipment	Golf Accessories	51514343.88	3119747
Golf Equipment	Irons	254814338	391445
Golf Equipment	Putters	106184271.4	1284570
Golf Equipment	Woods	313898414.7	317939

Load Transform Data Cancel

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

Untitled - Power BI Desktop Xuesong Gao

File Home Insert Modeling View Help Format Data / Drill

Paste Get data Excel Power BI datasets SQL Server Transform data Refresh New visual Text box More visuals Publish

Clipboard Data Queries Insert Calculations Share

Back to report

Product line	Quantity	Revenue
Golf Equipment	5113701	726,411,367.89
Golf Accessories	3119747	51,514,343.88
Irons	391445	254,814,337.99
Putters	1284570	106,184,271.37
Woods	317939	313,898,414.65
Total	5113701	726,411,367.89

Fields

Search

Visualizations

Filters

Query1

- Product line
- Product type
- Σ Quantity
- Σ Revenue

Page 1 of 1 Storage Mode: DirectQuery (click to change)

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.

The Analytics Connector only supports 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**.

SQL Server database

Server

Database (optional)

Data Connectivity mode DirectQuery

Advanced options

Command timeout in minutes (optional)

SQL statement (optional, requires database)

```
Select *
From
  RunReport (
    "Connector Reports"."Parameter Report",
    "p_product line code"=[991,992,993,994,995]",
    "p_Product Line"='Golf Equipment',
    ...
  )
```

Include relationship columns

Navigate using full hierarchy

Enable SQL Server Failover support

OK Cancel

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

localhost: GO Sales (query)

Product line code	Product line	Product type code	Product type	Year	Revenue
995	Golf Equipment	968	Irons	2010	4654281.1
995	Golf Equipment	969	Woods	2010	5138304.03
995	Golf Equipment	970	Putters	2010	2459044
995	Golf Equipment	971	Golf Accessories	2010	864227.83

Load Transform Data Cancel

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

Untitled - Power BI Desktop Xuesong Gao

File Home Insert Modeling View Help Format Data / Drill

Paste Get data Excel Power BI datasets SQL Server Transform data Refresh New visual Text box More visuals Publish

Clipboard Data Queries Insert Calculations Share

Back to report

Product line	2010	Total
Golf Equipment	13,115,856.96	13,115,856.96
Golf Accessories	864,227.83	864,227.83
Irons	4,654,281.10	4,654,281.10
Putters	2,459,044.00	2,459,044.00
Woods	5,138,304.03	5,138,304.03
Total	13,115,856.96	13,115,856.96

Fields

Search

Visualizations

Filters

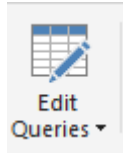
Query1

- Product line
- Σ Product line code
- Product type
- Σ Product type code
- Σ Revenue
- Σ Year

Page 1 of 1 Storage Mode: DirectQuery (click to change)

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.

A screenshot of the "Parameters" dialog box in Power BI. The dialog has a title bar "Parameters" and a close button "X". On the left, there is a list of parameters with "Start_Date" selected. A "New" button is visible above the list. On the right, the configuration for the "Start_Date" parameter is shown: Name: Start_Date; Description: Start date to pass to the parameterized table; Required: checked; Type: Date; Suggested Values: Any value; Current Value: 1/1/2010. At the bottom right, there are "OK" and "Cancel" buttons.

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.

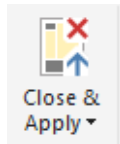
The below example formats a date parameter to text and appends it to the SQL statement.


```

= Odbc.Query("dsn=Senturus Colo .45", "Select * From (#(1f)OpenTable(#(1f) ""Sales
(query)"".""Parameterized Products"", #(1f) ""p_product line code""=""[991, 992, 993, 994, 995]"" , #
(1f) ""p_Product Line""=""Golf Equipment'"" , #(1f) ""p_Date""=""& Date.ToText(Start_Date, ""yyyy-MM-
dd"" & ""'"" , #(1f) ""p_Datetime""=""2010-01-01 19:00:00'""#(1f)#(1f) as ""Parameterized Table'")

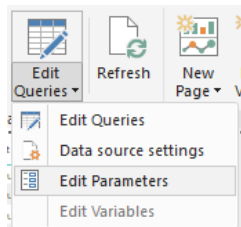
```

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



Click **Close & Apply**.

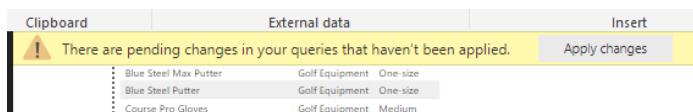
To change the Parameter value(s), select *Edit Parameters* under **Edit Queries**.



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 use the same syntax.

Syntax

Both OpenTable and RunReport function use the 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 across schemas.

Any additional parameters should be defined as 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 the 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.

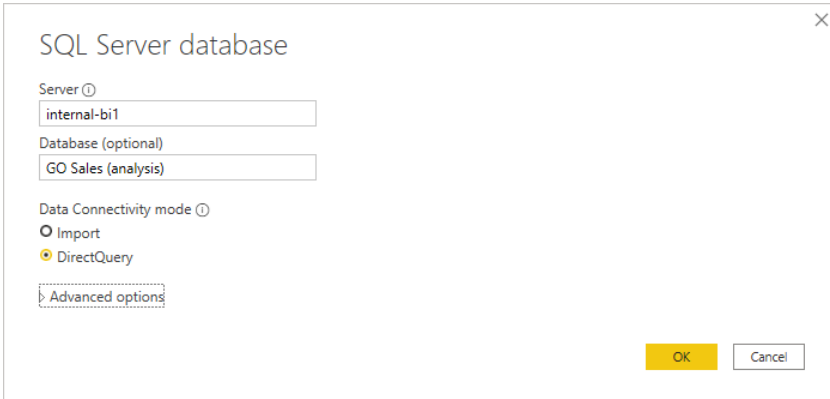
Publish to powerbi.com

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 to your on-premises data, you need to install and configure Power BI Gateway. If you use the custom 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*.

To publish a Power BI report from Desktop to powerbi.com, follow steps below:

1. Connect to the Senturus Analytics Connector server using SQL Server database connection or custom data connector.



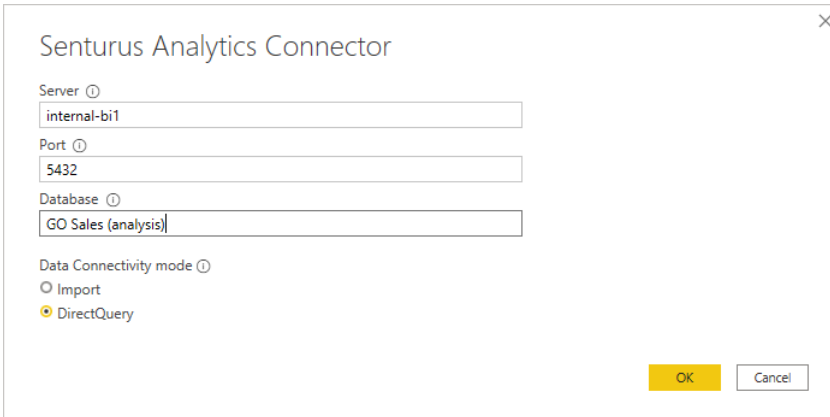
SQL Server database

Server

Database (optional)

Data Connectivity mode Import DirectQuery

Or



Senturus Analytics Connector

Server

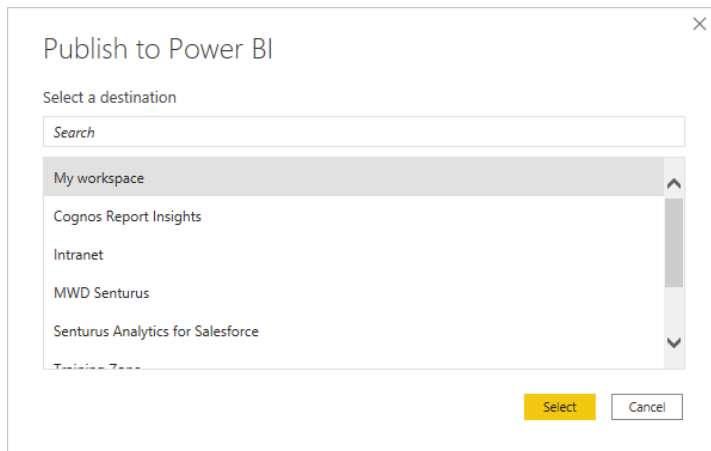
Port

Database

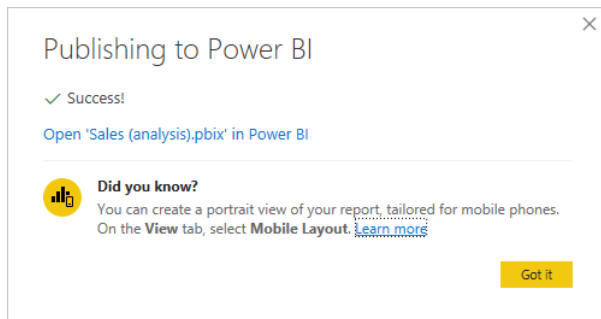
Data Connectivity mode Import DirectQuery

2. Create your visualization, save Power BI report, click Publish button.

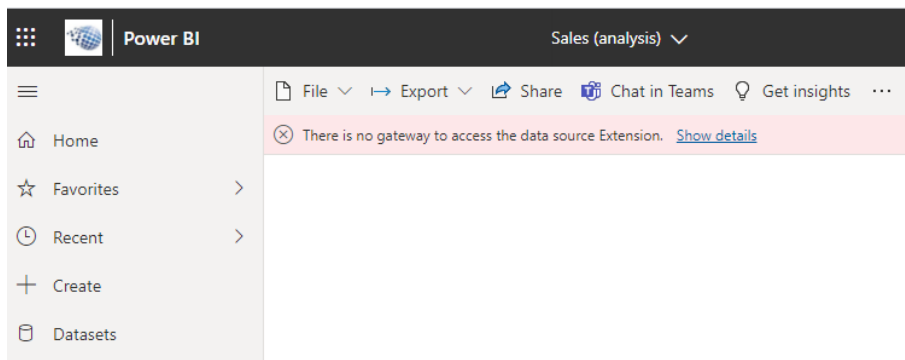
3. Select a destination and click on Select.



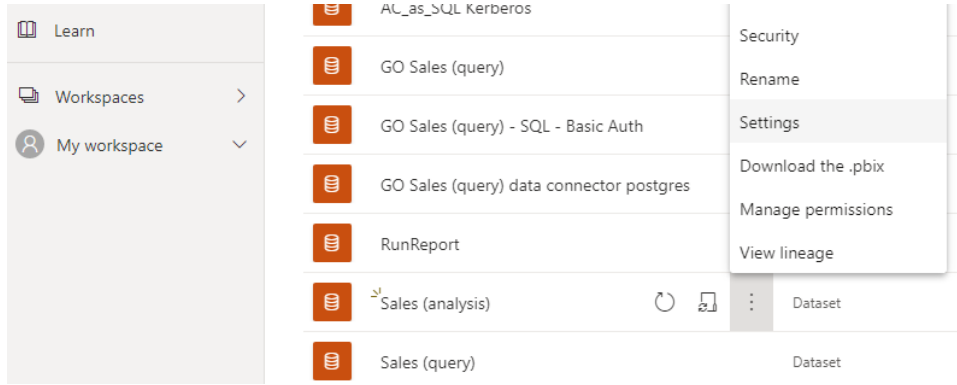
4. Click on “Open ‘your report name.pbix’ Power BI” in the success dialog:



5. You will see an error if you did not setup a gateway using the same connection settings you entered in step 1 before.



6. Go to the workspace, hover over your mouse right to the newly created data set, and click on “More options” button, then “Settings”:



7. Click on the arrow button under the Actions column and click on “Manually add to gateway”. If you added a data source using the same connection settings, you will see a drop-down list of data sources instead of “Manually add to gateway” button. In that case, you can select a data source for your dataset without creating a new data source.

Settings for GO Sales (query)

⚠ One or more cloud data sources for this dataset have been deleted.
[Recreate cloud data sources](#)

[Refresh history](#)

▶ Dataset description

◀ Gateway connection

To use a data gateway, make sure the computer is online and the data source is added in [Manage Gateways](#). If you're using an On-premises data gateway (standard mode), please select the corresponding data sources and then click apply.

Use an On-premises or VNet data gateway

Off

Gateway	Department	Contact information	Status	Actions
bi1.gateway		xgao@senturus.com	⊗ Not configured correctly	▼
Data sources included in this dataset:				
<div style="border: 1px solid gray; padding: 5px;"> ⊗ Extension["extensionDataSourceKind":"SenturusAnalyticsConnector","extensionDataSourcePath":{"server":"internal-bi1.senturus.com","port":"5432","database":"GO Sales (query)"}] Manually add to gateway </div>				

8. Now you can create a data source under a particular gateway.

If you are using the SQL Server connector in your report, you can create data sources just like you do for other SQL server data source except that you have to pick Basic “Authentication method”.

Here is an example:

Data Source Settings Users

Data Source Name

GO Sales (analysis)

Data Source Type

SQL Server

Server

internal-bi

Database

Sales (analysis)

Authentication Method

Basic

The credentials are encrypted using the key stored on-premises on the gateway server. [Learn more](#)

Username

cognosembed@senturus.com

Password

.....

Skip Test Connection

> Advanced settings

Add

Discard

If you are using the custom connector, you need to select “Senturus Analytics Connector” from the Data Source Type list. Here is an example

Data Source Settings Users

Data Source Name

Data Source Type

Server

Port

Database

The credentials are encrypted using the key stored on-premises on the gateway server. [Learn more](#)

Username

Password

Skip Test Connection

> Advanced settings

Please note, you need to type in the exact server, port and database here as you have in Power BI Desktop (case sensitive). Please refer to step 1.

9. After creating data source, go back to data set settings, map the data source and click on Apply.

Settings for Sales (analysis)

⚠ One or more cloud data sources for this dataset have been deleted.
[Recreate cloud data sources](#)

[Refresh history](#)

▶ Dataset description

◀ Gateway connection

To use a data gateway, make sure the computer is online and the data source is added in [Manage Gateways](#). If you're using an On-premises data gateway (standard mode), please select the corresponding data sources and then click apply.

Use an On-premises or VNet data gateway

On

Gateway	Department	Contact information	Status	Actions
bi1.gateway		xgao@senturus.com	Running on INTERNAL-BI1	▼
Data sources included in this dataset:				
<div style="border: 1px solid #ccc; padding: 5px;"><p>Extension{"extensionDataSourceKind":"SenturusAnalyticsConnector","extensionDataSourcePath":{"server":"internal-bi1","port":"5432","database":"GO Sales (analysis)"}}</p><p style="text-align: right;">Maps to:</p><p>GO Sales (analysis) ▼</p></div>				
<input type="button" value="Apply"/>	<input type="button" value="Discard"/>			

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

Legacy Documentation

This section is for the custom connector. While the custom connector is still supported, Senturus **highly** recommends using the built in SQL Server Connector.

Install the Custom Connector

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

To use the custom Data Connector, follow steps below:

1. Install Senturus Analytics Connector ODBC client. The latest ODBC client can be found at [Senturus Analytics Connector - Customer Resources](#).⁶
2. 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.

3. 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.

⁶ ODBC client is automatically installed when you are installing Senturus Analytics Connector server. So, you don't have to install ODBC client separately on the same computer.

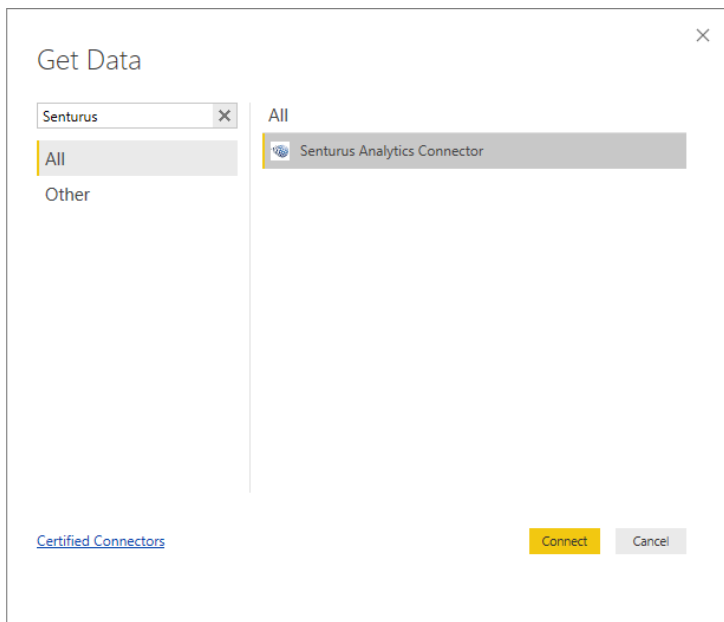
Connect via Custom Connector

Before using this connector, make sure you followed instructions in the [Install the Custom Connector](#) section.

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

Find the Senturus Analytics Connector in the list.

Click **Connect**.

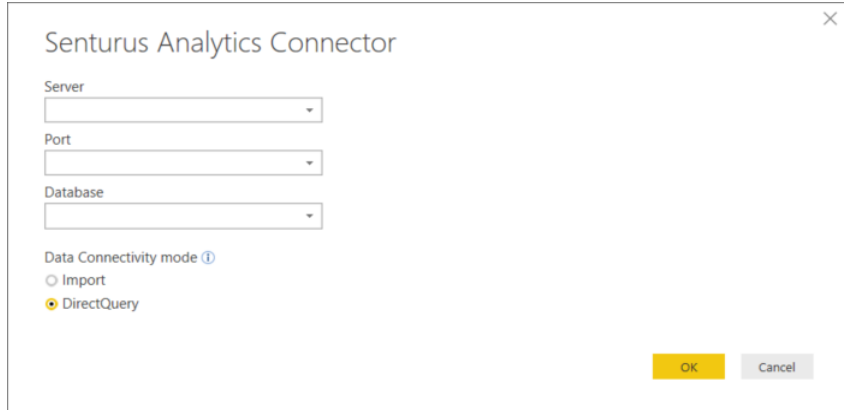


Power BI will display a dialog where you can select the Server, Port, Database, and Data Connectivity modes.

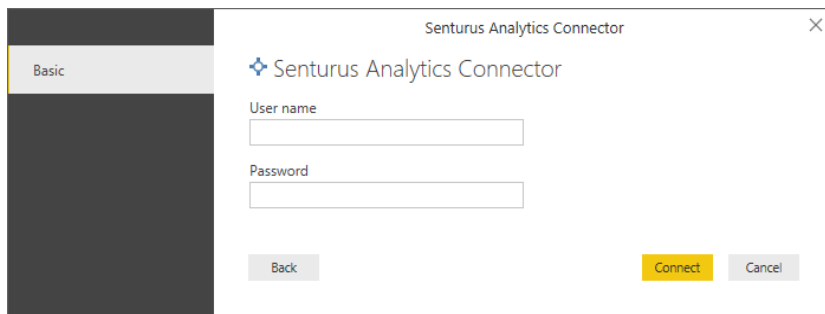
The Server, Port and Database lists are pulled from the configuration information in the DataSourceConfig.xml file under the server's Analytics Connector installation folder.

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**.

Navigator

Display Options

- Cognos_Senturus: GO Sales (query) [1]
 - GO Sales (query) [11]
 - Connector Reports
 - Filters and calculations
 - Inventory (query)
 - ns1
 - ns2
 - Product forecast (query)
 - Returned items (query)
 - root
 - Sales (query) [14]
 - AAA_CALCULATIONS
 - Branch
 - Order
 - Order method
 - Parameterized Products
 - Parameterized Products 2
 - Products
 - Retailer type
 - Retailers

Branch

Preview downloaded on Thursday, October 4, 2018

_LINK_COLUMN_	Branch code	Address 1	Address 1 (multis
1	6	75, rue du Faubourg St-Honoré	75, rue du Fau
2	7	Piazza Duomo, 1	Piazza Duomo
3	9	Singelgravenplein 4	Singelgravenp
4	13	Schwabentor 35	Schwabentor :
5	14	Leopoldstraße 36	Leopoldstraße
6	15	Isafjordsgatan 30 C	Isafjordsgatan
7	17	7800, 756 - 6th Avenue. S.W.	7800, 756 - 6t
8	18	789 Yonge Street	789 Yonge Str
9	19	1288 Dorchester Avenue	1288 Dorches
10	20	299 Yale Avenue	299 Yale Aven
11	21	1288 South Barrington Ave.	1288 South Ba
12	22	10032 NW 186th	10032 NW 186
13	23	6c, rue de l'Église	6c, rue de l'Ég
14	24	Prol. Paseo de la Reforma No. 51	Prol. Paseo de
15	25	202-2-3 Hyakunincho	202-2-3 百人
16	26	543-225 Asahi	543-225 旭
17	28	2315 Queen's Ave	2315 Queen's
18	29	Plaza de la Constitución, s/n	Plaza de la Co
19	30	Avenida Paulista, 333	Avenida Pauli
20	31	Kauppakatu 33	Kauppakatu 3
21	32	234-12, Kongdeok-Dong	공덕동 234-1
22	33	10 Claymore Hill	克莱莫山 10

Load Edit Cancel

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

Apply query changes

- Sales
Evaluating...
- Time
Evaluating...
- Products
Evaluating...
- Branch
Evaluating...

Cancel

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

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.