

***BI Reporting with PI Integrator for  
Business Analytics  
Version 2020 R2***

**OSIsoft, LLC**  
**1600 Alvarado Street**  
**San Leandro, CA 94577**

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, mechanical, photocopying, recording, or otherwise, without the prior written permission of OSIsoft, LLC.

OSIsoft, the OSIsoft logo and logotype, Managed PI, OSIsoft Advanced Services, OSIsoft Cloud Services, OSIsoft Connected Services, OSIsoft EDS, PI ACE, PI Advanced Computing Engine, PI AF SDK, PI API, PI Asset Framework, PI Audit Viewer, PI Builder, PI Cloud Connect, PI Connectors, PI Data Archive, PI DataLink, PI DataLink Server, PI Developers Club, PI Integrator for Business Analytics, PI Interfaces, PI JDBC Driver, PI Manual Logger, PI Notifications, PI ODBC Driver, PI OLEDB Enterprise, PI OLEDB Provider, PI OPC DA Server, PI OPC HDA Server, PI ProcessBook, PI SDK, PI Server, PI Square, PI System, PI System Access, PI Vision, PI Visualization Suite, PI Web API, PI WebParts, PI Web Services, RLINK and RtReports are all trademarks of OSIsoft, LLC.

All other trademarks or trade names used herein are the property of their respective owners.

**U.S. GOVERNMENT RIGHTS**

Use, duplication or disclosure by the US Government is subject to restrictions set forth in the OSIsoft, LLC license agreement and/or as provided in DFARS 227.7202, DFARS 252.227-7013, FAR 12-212, FAR 52.227-19, or their successors, as applicable.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, mechanical, photocopying, recording or otherwise, without the written permission of OSIsoft, LLC.

---

## Software Versions Used in this Document

The list below describes the software versions used in this version of the course.

<b>Software</b>	<b>Version</b>
PI DataLink	2019 SP1
Microsoft Office	2016
PI ODBC Driver	2016 R2
PI SQL Client	2018 R2
PI Integrator for Business Analytics Advanced Edition	2018 R2 SP1
PI OLEDB Enterprise	2019
Microsoft SQL Server	2016
PI Data Archive	2018 SP3
PI Asset Framework	2018 SP3
PI Vision	2019

## Contents

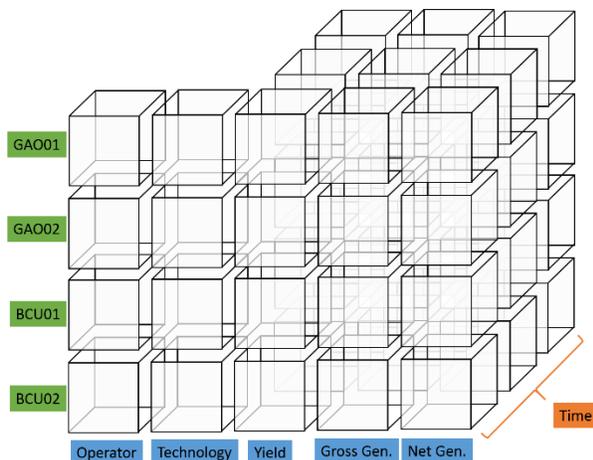
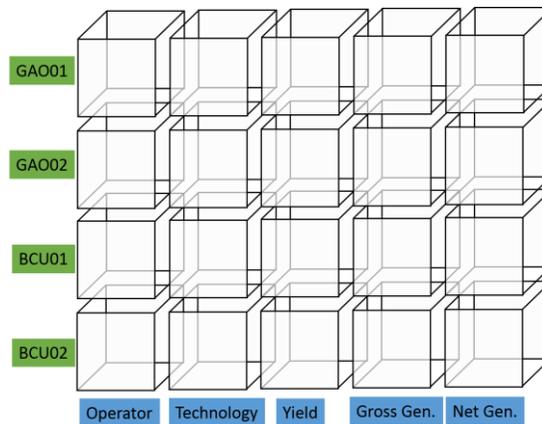
<b>Lesson 1: Business Intelligence and PI Asset Framework</b> .....	<b>3</b>
Introduction to Power BI.....	4
Activity – Inspect a Sample Power BI Report .....	5
<b>Lesson 2: Power BI Reports using PI Integrator for BA</b> .....	<b>14</b>
PI AF Hierarchy and Data Set .....	14
<b>Lesson 3: PI Integrator for Business Analytics</b> .....	<b>16</b>
Product Information and Features.....	16
Architecture Used in Class .....	19
PI Integrator Web UI .....	22
Activity – Create the Transformer Loading View .....	24
<b>Lesson 4: Building the Distribution Network Reports</b> .....	<b>34</b>
Importing Data into Power BI .....	34
Building the Report Visuals.....	37
<b>Lesson 5: Building the Fleet Generation Report</b> .....	<b>59</b>
The Online Fleet Generation Database .....	59
Preparing and Importing the Tables .....	62
Augmenting the Data using DAX .....	76
Configuring the Visualizations.....	79
<b>Final Challenge: Fleet Generation Report with Event Frames</b> .....	<b>83</b>
Activity – Backfill Event Frames in the Online Fleet Generation database .....	83
Activity – Create an Event View with PI Integrator for BA .....	86
Activity – Build the Final Report .....	96

# Lesson 1: Business Intelligence and PI Asset Framework

Business intelligence (BI) tools offer solutions to quickly analyze raw, un-normalized, multidimensional data. Values from the PI Data Archive, external metadata, and calculations from Asset Framework can be transformed by business intelligence tools into actionable analysis and interactive reports in order to gain insight into business and operational processes.

One of the key concepts of the course is how PI AF metadata and hierarchy information inherently provides the relationships and dimensions used to filter and slice data in BI tools.

The Distribution Network and Fleet Generation databases have a comprehensive amount of information including a hierarchy of substations and metadata for each asset. The figure to the right depicts a data cube that captures metadata and real-time data of generating units. This data will be incorporated into a BI cube and used to develop interactive reports that allow us to “slice and dice” our data and bring meaning to our multidimensional data cube.



Inclusion of additional attributes through table lookups and analytics on existing attributes allow for the expansion of additional columns (or dimensions) to the data cube above.

Further, historical data, interpolated or compressed, add an additional dimension of information that bring more meaning in Business Intelligence reports.

In the next several chapters in the course, we will be using a pair of AF databases to expose meaningful data that will help management and engineers make better, more informed decisions.

## Introduction to Power BI

Power BI is a business analytics service and client provided by Microsoft. It provides interactive visualizations with self-service business intelligence capabilities where end users can create reports and dashboards by themselves without having to depend on information technology staff or database administrators.

Some of the benefits of Power BI:

- Less work than Excel for more complex analysis and visuals
- Can solve problems that are simply too large for Excel and PI DataLink (MS Excel is limited to 1 million rows)
- Cheap – [Free download](#) or \$9.99 / month per user for Power BI Pro
- Live reporting and centralized web-based dashboards in Office 365 and Power BI Server
- Slick visuals including 3<sup>rd</sup> Party Visuals in [Microsoft AppSource](#)

## Activity – Inspect a Sample Power BI Report

### Objective:

- Explore a sample Power BI Report.

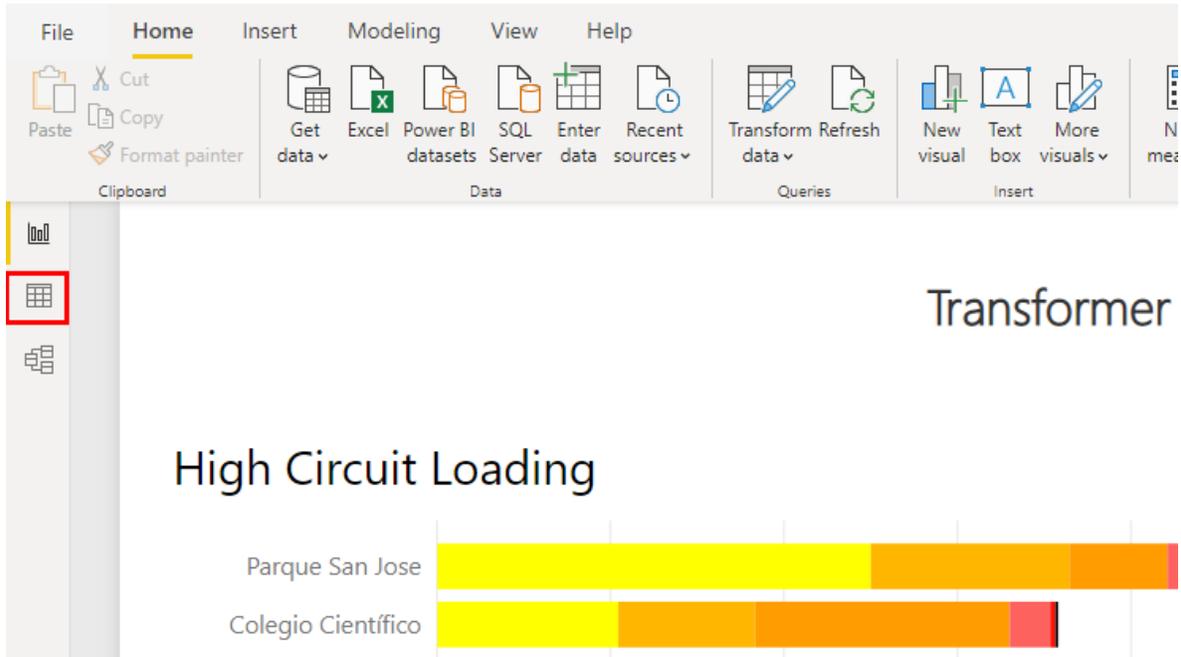
### Approach

We'll start by getting a feel for Power BI using a pre-built report. **Open C:\Class\Part 1 - PI Integrator for BA\Sample Report.pbix**

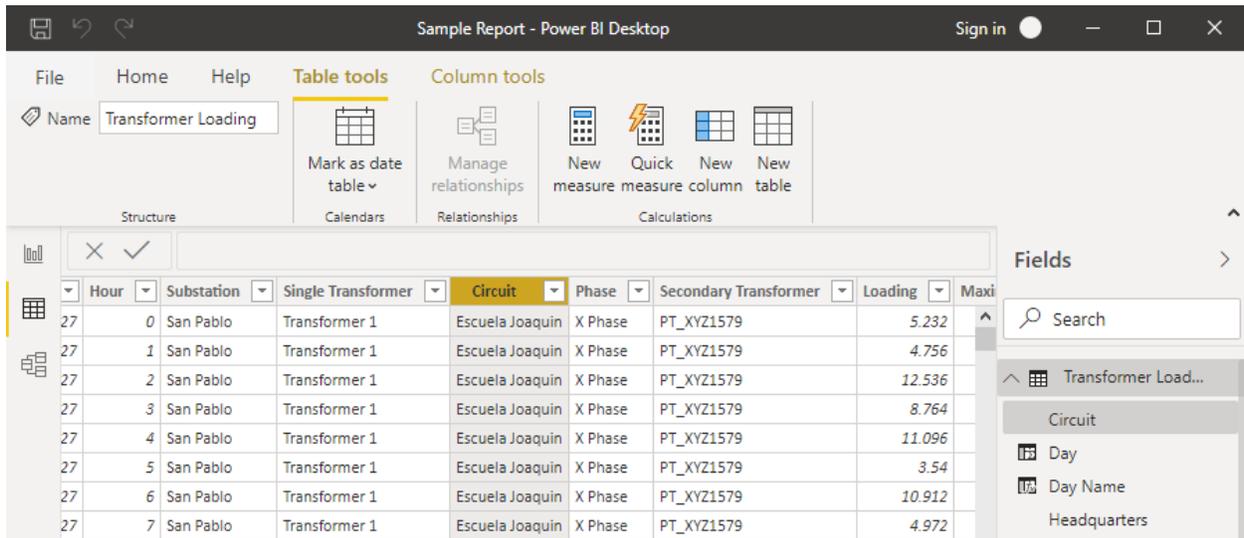


Right now we're looking at the **Report View** where the report and visuals are configured. Start clicking on the visuals and the rest of the report will be filtered to only include the selected items. This is often referred to as **slicing and dicing** the data.

Click on the **Data Tab** to inspect the data set we'll be working with:



Note that all the columns are available in the **Fields List**:

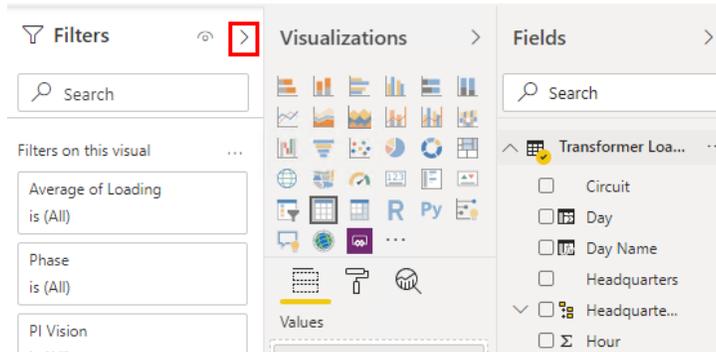


Go back to the Report View, click on the **Pole Transformers Table** visual, and note the **Filters Pane**, **Visualizations Pane**, and **Fields Pane**. These sections are where the bulk of the configuration takes place. Columns from the data set are dragged and dropped from the Fields Pane onto the various sections in the Visualizations Pane. We can see that the Secondary Transformer, Phase, Service Hours, Average of Loading, and PI Vision columns are being displayed in the table.

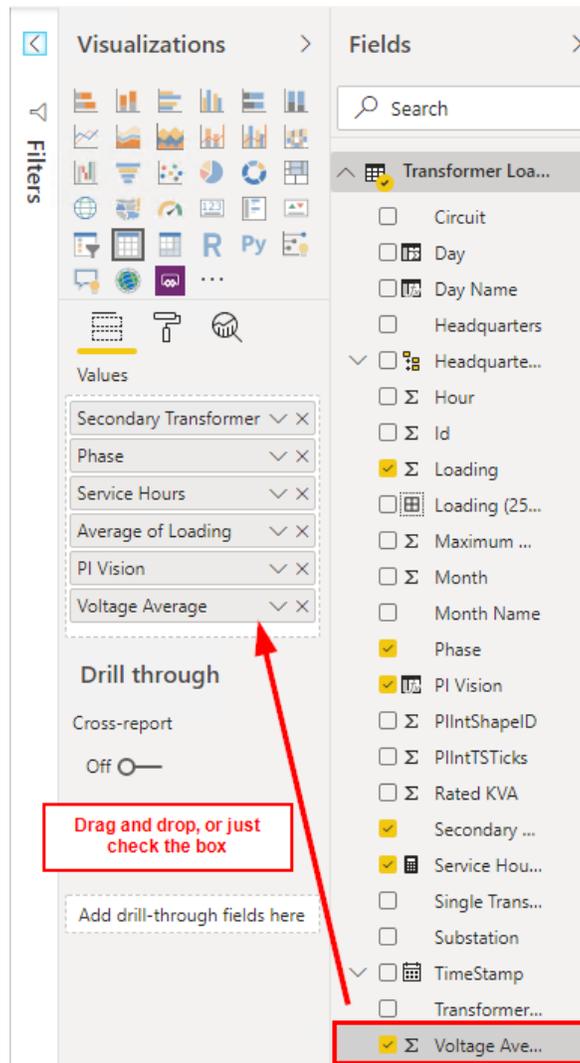
The screenshot displays the configuration interface for a report, divided into three main panes:

- Filters Pane:** Contains a search bar and several filter sections:
  - Filters on this visual:** Includes filters for 'Average of Loading is (All)', 'Phase is (All)', 'PI Vision is (All)', and 'Secondary Transformer top 20 by Average of...'. There is also an 'Add data fields here' button.
  - Filters on this page:** Includes an 'Add data fields here' button.
  - Filters on all pages:** Includes an 'Add data fields here' button.
- Visualizations Pane:** Shows a grid of visualization icons. Below the icons, a 'Values' section lists the selected fields: 'Secondary Transformer', 'Phase', 'Service Hours', 'Average of Loading', and 'PI Vision'. There is also a 'Drill through' section with 'Cross-report' set to 'Off' and 'Keep all filters' set to 'Off', along with an 'Add drill-through fields here' button.
- Fields Pane:** Contains a search bar and a list of available data fields. The 'Transformer Loa...' section is expanded, showing fields such as 'Circuit', 'Day', 'Day Name', 'Headquarters', 'Headquarte...', 'Hour', 'Id', 'Loading' (checked), 'Loading (25...', 'Maximum ...', 'Month', 'Month Name', 'Phase' (checked), 'PI Vision' (checked), 'PIIntShapelD', 'PIIntTSTicks', 'Rated KVA', 'Secondary ...' (checked), 'Service Hou...' (checked), 'Single Trans...', 'Substation', and 'TimeStamp'.

Minimize the Filters Pane for now.



Add the **Voltage Average** column to the table by selecting the table and doing a drag and drop:

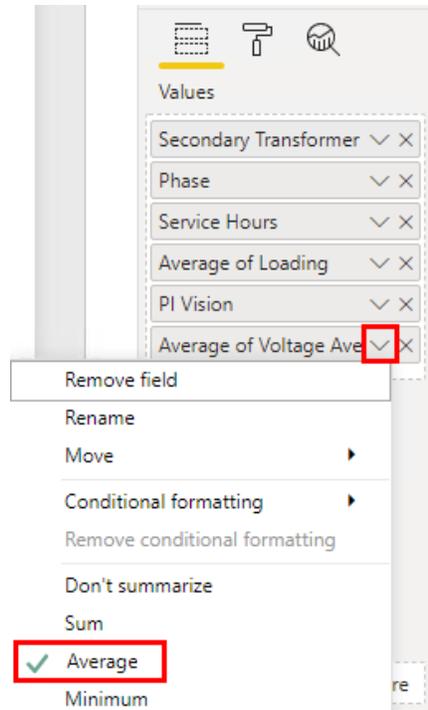


Resize and/or reposition the visual so that the Voltage Average column is visible.

**Pole Transformers**

Secondary Transformer	Phase	Service Hours	Loading	Sum	Voltage Average
PT_XYZ0109	X Phase	2208	130.50		362,473.77
PT_XYZ0911	X Phase	2208	122.20		534,218.20
PT_XYZ0377	Z Phase	2208	117.11		546,361.94
PT_XYZ0096	X Phase	2208	116.99		541,711.65
PT_XYZ0884	Z Phase	2208	103.06		541,607.03
PT_XYZ0566	Y Phase	2208	96.66		531,744.63
PT_XYZ0071	Y Phase	2208	95.68		541,635.89
PT_XYZ0410	Z Phase	2208	92.94		544,598.67
PT_XYZ0644	X Phase	2208	87.80		530,424.16
PT_XYZ1470	X Phase	2208	83.37		527,342.39
PT_XYZ0126	X Phase	2208	81.05		535,169.24
PT_XYZ0589	X Phase	2208	81.05		531,459.91
PT_XYZ0428	Z Phase	2208	81.24		541,167.24
PT_XYZ0254	X Phase	2208	81.08		526,407.05
PT_XYZ0195	Y Phase	2208	80.18		537,319.93
PT_XYZ0210	X Phase	2208	79.63		541,671.41
PT_XYZ0587	X Phase	2208	79.53		531,172.83
PT_XYZ0063	X Phase	2208	78.23		539,603.74
PT_XYZ0065	Y Phase	2208	78.01		534,644.22
PT_XYZ0608	X Phase	2208	76.36		531,702.44

The Voltage Average will be displayed in the table, but by default all the voltages from all the rows will be summed by transformer. It makes more sense to summarize these as an average, so click the drop down and change the summary to **Average**:

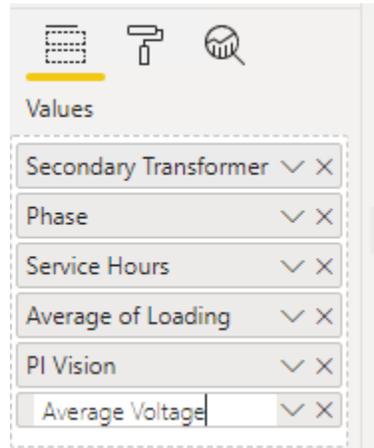


Resize the column directly on the visual similar to Excel:

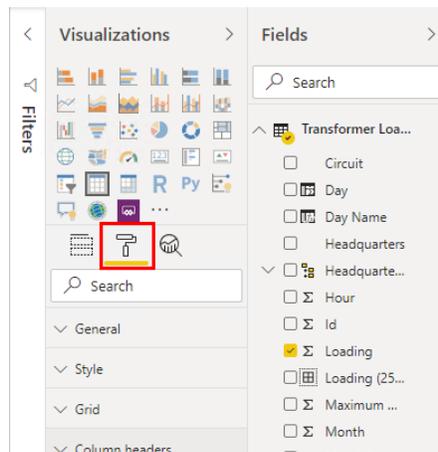
### Pole Transformers

Secondary Transformer	Phase	Service Hours	Average of Loading	PI Vision	Average of Voltage Average
rZ0109	X Phase	2208	130.50		244.16
rZ0911	X Phase	2208	122.20		241.95
rZ0377	Z Phase	2208	117.1		247.45
rZ0096	X Phase	2208	116.99		245.34
rZ0884	Z Phase	2208	103.06		245.29
rZ0566	Y Phase	2208	96.66		240.83
rZ0071	Y Phase	2208	95.68		245.31
rZ0410	Z Phase	2208	92.94		246.65
rZ0644	X Phase	2208	87.80		240.23
rZ1470	X Phase	2208	83.37		238.83
rZ0126	X Phase	2208	82.92		242.38
rZ0589	X Phase	2208	81.63		240.70
rZ0428	Z Phase	2208	81.24		245.09
rZ0254	X Phase	2208	81.08		238.50
rZ0195	Y Phase	2208	80.18		243.36
rZ0210	X Phase	2208	79.63		245.32
rZ0587	X Phase	2208	79.53		240.57
rZ0063	X Phase	2208	78.23		244.39
rZ0065	Y Phase	2208	78.01		242.14
rZ0608	X Phase	2208	76.26		240.81

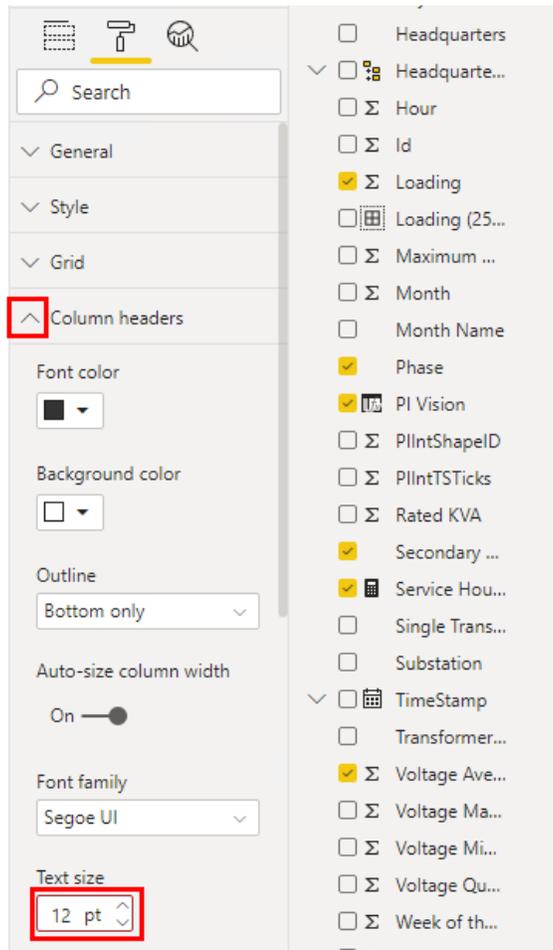
Average of Voltage Average is a pretty weird header name, so double click it and **rename it to Average Voltage**:



Other **Formatting Options** are available by clicking the paint roller icon:



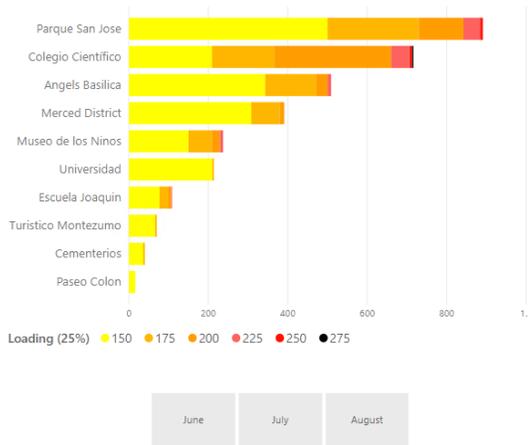
We'll go over more formatting options in the exercises, so for now just bump up the text size of the column headers to 12:



Adjust the column widths and reposition the visuals to make everything fit. Your report should now look something like this.

Transformer High Loading

High Circuit Loading



Pole Transformers

Secondary Transformer	Phase	Service Hours	Average of Loading	PI Vision	Average Voltage
PT_XYZ0109	X Phase	2208	130.50	🔍	164.16
PT_XYZ0911	X Phase	2208	122.20	🔍	241.95
PT_XYZ0377	Z Phase	2208	117.11	🔍	247.45
PT_XYZ0096	X Phase	2208	116.99	🔍	245.34
PT_XYZ0884	Z Phase	2208	103.06	🔍	245.29
PT_XYZ0566	Y Phase	2208	96.66	🔍	240.83
PT_XYZ0071	Y Phase	2208	95.68	🔍	245.31
PT_XYZ0410	Z Phase	2208	92.94	🔍	246.65
PT_XYZ0644	X Phase	2208	87.80	🔍	240.23
PT_XYZ1470	X Phase	2208	83.37	🔍	238.83
PT_XYZ0126	X Phase	2208	82.92	🔍	242.38
PT_XYZ0589	X Phase	2208	81.63	🔍	240.70
PT_XYZ0428	Z Phase	2208	81.24	🔍	245.09
PT_XYZ0254	X Phase	2208	81.08	🔍	238.50
PT_XYZ0195	Y Phase	2208	80.18	🔍	243.36
PT_XYZ0210	X Phase	2208	79.63	🔍	245.32
PT_XYZ0587	X Phase	2208	79.53	🔍	240.57
PT_XYZ0063	X Phase	2208	78.23	🔍	244.39
PT_XYZ0065	Y Phase	2208	78.01	🔍	242.14
PT_XYZ0608	X Phase	2208	76.36	🔍	240.81

We will build a similar report from scratch in a future exercise.

## Lesson 2: Power BI Reports using PI Integrator for BA

We will be working with a data set for a power distribution company, which includes electrical characteristics for over 1500 single-phase transformers. The source data will be published in a data-science ready format using PI Integrator for BA. Once this is done, we'll configure an array of Power BI visuals and integrate the results with PI Asset Framework and PI Vision.

The transformers we will be analyzing are secondary transformers that deliver power to homes and businesses, which you may have seen on a pad or pole in your own neighborhood. There are thousands of them to keep track of, making this a difficult problem to solve using Excel.

The transformers themselves are not actually instrumented. The power and voltage characteristics we will analyze have actually been computed by rolling up child Meters in PI AF.

### PI AF Hierarchy and Data Set

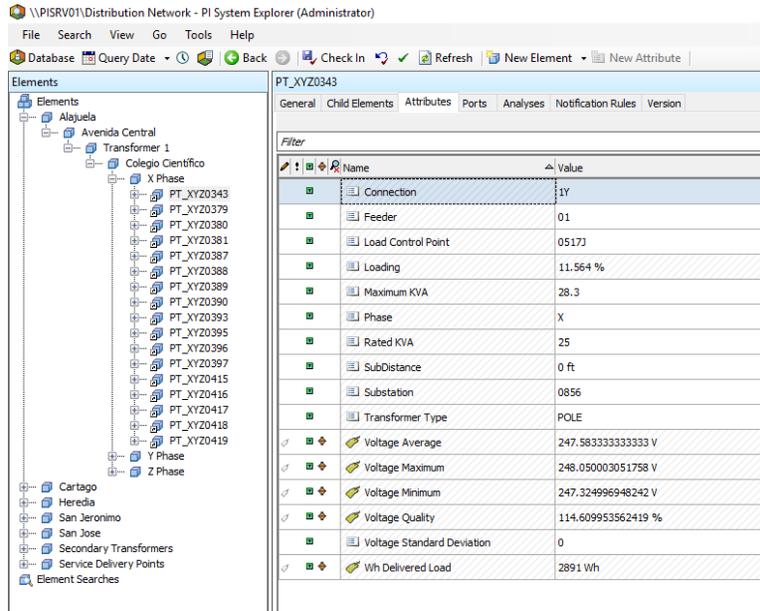
#### Objective:

- Better understand the data set used in the following chapters

We will take a few minutes to understand where the data set came from and relate the sample Power BI report back to the PI System. We are working with a data set for a fictitious power distribution company. They have built a PI AF Hierarchy for their transformers and meters serving a number of geographical areas. In this course, we will focus on analyzing the transformers.

Open PI System Explorer and head to the **Distribution Network AF database**. Drill down to a level with transformers (names starting with PT\_) and inspect the available attributes. We will be using a sub-set of these attributes for all of our analysis, in addition to leveraging the AF hierarchy.

Note that because the Transformers are not instrumented, Voltages, Voltage Quality, and Wh Delivered are rolled up from the child meters using Asset Analytics.



The screenshot shows the PI System Explorer interface. The left pane displays the AF hierarchy for the 'Distribution Network - PI System Explorer (Administrator)'. The selected element is 'PT\_XYZ0343'. The right pane shows the 'Attributes' tab for this element, displaying a table of attributes and their values.

Name	Value
Connection	1Y
Feeder	01
Load Control Point	0517J
Loading	11.564 %
Maximum KVA	28.3
Phase	X
Rated KVA	25
SubDistance	0 ft
Substation	0856
Transformer Type	POLE
Voltage Average	247.583333333333 V
Voltage Maximum	248.050003051758 V
Voltage Minimum	247.324996948242 V
Voltage Quality	114.609953562419 %
Voltage Standard Deviation	0
Wh Delivered Load	2891 Wh

Data from this PI AF hierarchy will be published for use in a Power BI report in a later exercise.

## Lesson 3: PI Integrator for Business Analytics

In simple terms, PI Integrator for Business Analytics reads data from PI Asset Framework and writes the data to a variety of 3<sup>rd</sup>-party platforms and databases referred to as Targets. The most often used target is a Microsoft SQL Server database.

Getting the data out of the AF structure and into the client tools requires the use of integration software such as the PI Integrator for Business Analytics or PI System Access software. This chapter will discuss the former method of extracting the data.

The PI Integrators join your Business Intelligence (BI) infrastructure with OSIsoft's PI System, allowing you to combine high-value Operation Technology (OT) data from the PI System with Information Technology (IT) data for reporting, analytics, and application integrations. The integration of data from OT systems, such as automation and control systems and internet-enabled devices, with data from IT systems, such as transactional and business process systems, increases situational awareness, adds transparency into industrial operations and business processes, and makes it possible to anticipate problems and identify opportunities for process improvements.

### Product Information and Features

The section outlines general product, feature, and component information. The latest documentation can be found in the [PI Live Library](#).

#### Advanced vs Standard Edition

PI Integrator for Business Analytics Advanced Edition serves real-time packets of PI System data to streaming platforms such as Apache Kafka. Streaming platforms assist in operationalizing machine learning models and support Kappa and Lamda architectures for data consumption.

In short, Advanced Edition supports streaming views while Standard Edition does not.

#### PI Asset Framework Requirement

PI Integrator for Business Analytics requires a PI Asset Framework (PI AF) model to select PI System data to produce decision-ready data. The data can be cleansed using a variety of filters and enhanced with asset, event, and time context from the PI System.

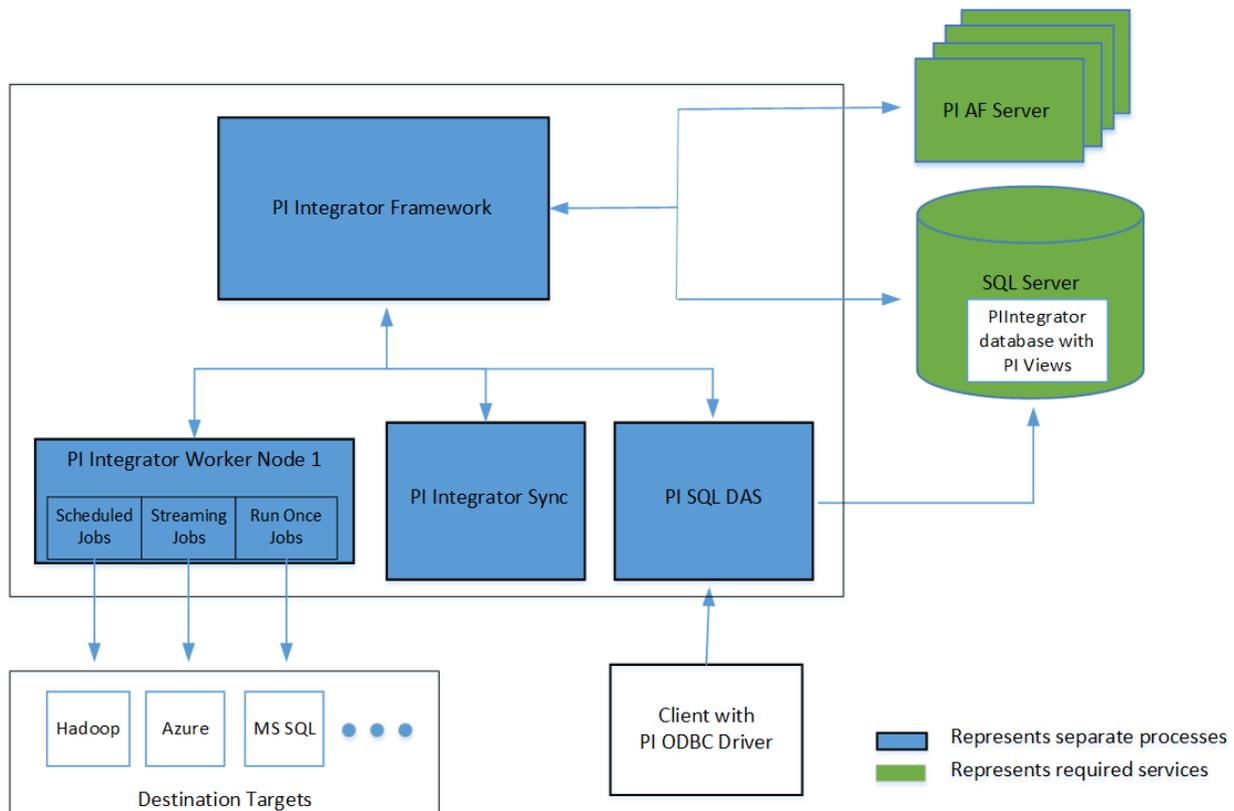
PI Tags cannot be used directly. They must be mapped in PI AF in order to be exposed with PI Integrator for BA.

## Licensing by Streams

PI Integrator for Business Analytics is licensed by stream count. A stream is a unique instance of a PI Tag used in one of the PI Integrator views. If the same tag is used in multiple views it only counts as one stream. Publishing static attributes does not consume streams.

## General Software Architecture

Consult the [PI Live Library](#) for the latest documentation. The system architecture for PI Integrator for BA 2020 from the online documentation is shown below:



The PI Integrator for BA architecture can be confusing because it typically has components on 3 separate servers and that's not even counting Targets:

- All the roles in blue always exist on a single server and represent a set of Windows Services and the Web Page used to configure and administer PI Integrator for BA.
- PI Integrator for BA has backend components in a PI AF Configuration database which are separate from the elements and attributes used as data sources.

- PI Integrator for BA has 3 backend databases on a Microsoft SQL Server: PIIntegratorDB, PIIntegratorLogs, and PIIntegratorStats. These are separate from any configured SQL Server targets.

## Targets

As of the PI Integrator for BA 2020 release, the following Targets are supported by the Advanced Edition:

- Amazon Kinesis Data Streams
- Amazon Redshift
- Amazon S3
- Apache Hive
- Apache Kafka
- Azure Data Lake Storage Gen 1
- Azure Event Hubs
- Azure IoT Hub
- Azure SQL Database
- Azure SQL Data Warehouse
- Google BigQuery
- Google Cloud Storage
- Google Cloud Pub/Sub
- Hadoop Distributed File System (HDFS)
- Microsoft SQL Server
- Oracle
- SAP® HANA® ODBC
- Text file
- PI View (configured out of the box)

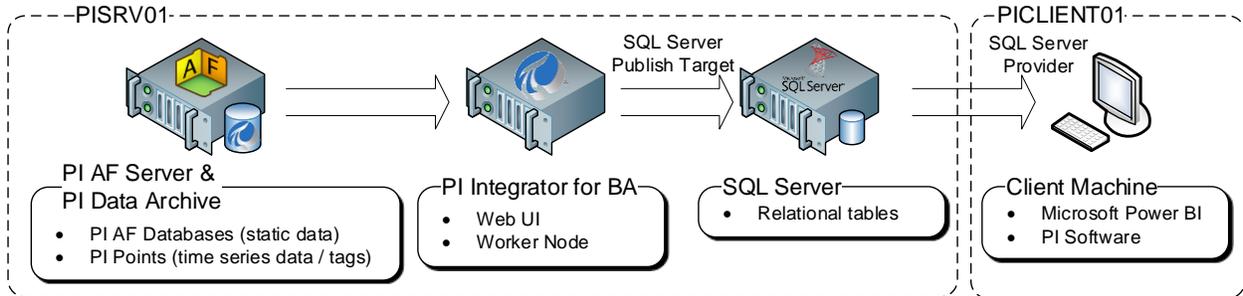
In this course we will only configure PI Integrator for BA to publish data to the Microsoft SQL Server target.

## The PI View Target

The PI View target is the only target that is configured out of the box. Technically the view data is stored in a SQL Server since it exists in the PIIntegratorDB database, but the supported access method is to use the PI ODBC Driver on the client querying the PI View. PI Views mainly exist for convenience and having a readily available Target when the installation completes. It is intended for testing and development sandbox scenarios only and should not be used if a Microsoft SQL Server target is available.

## Architecture Used in Class

In this course, all server roles including PI Data Archive, PI AF Server, SQL Server, and PI Integrator for BA are all installed on PISRV01. In a production grade architecture, each role would typically have its own dedicated server.



The PI Integrator for BA Web Page and Windows Services on PISRV01 are:

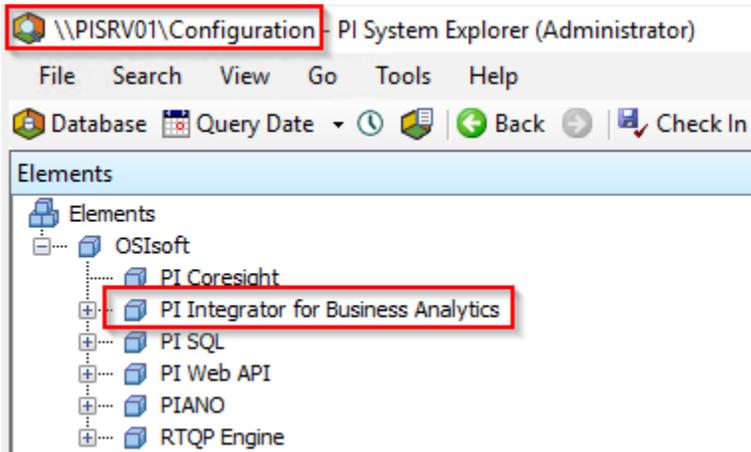
The screenshot shows the PI Integrator for BA web page and the Windows Services console.

**Web Page:** The browser shows the URL `pisrv01.pischool.int:444`. The page displays "My Views" with a table of views. The table has columns: Name, Run Status, Type, Run Mode, Start Time, End Time, and Last Run Time. One view is listed: "Distribution N..." with a status of "Published" (indicated by a red dot), Type "Asset", Run Mode "Once", Start Time "01-Jun-17", End Time "31-Aug-17 23:...", and Last Run Time "Apr 30, 2019 ...".

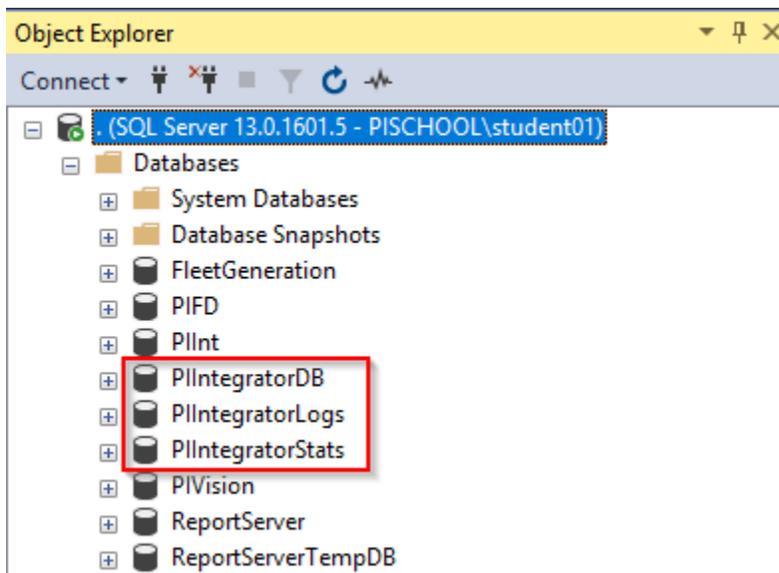
**Windows Services Console:** The list of services includes:

- PI Base Subsystem
- PI Buffer Subsystem
- PI Integrator Framework
- PI Integrator Sync
- PI Integrator Worker Node 1
- PI License Manager
- PI Message Subsystem
- PI Network Manager
- PI Notifications Service
- PI OLEDB Enterprise Agent
- PI Performance Equation Scheduler
- PI Ramp Soak Simulator (rmp\_sk) Interface X64
- PI Random Simulator (random) Interface X64
- PI Recalculator Subsystem
- PI Shutdown Subsystem
- PI Snapshot Subsystem
- PI SQL Data Access Server (PI Integrator Framework)
- PI SQL Data Access Server (RTQP Engine)
- PI SQL Subsystem
- PI Totalizer Subsystem
- PI Update Manager

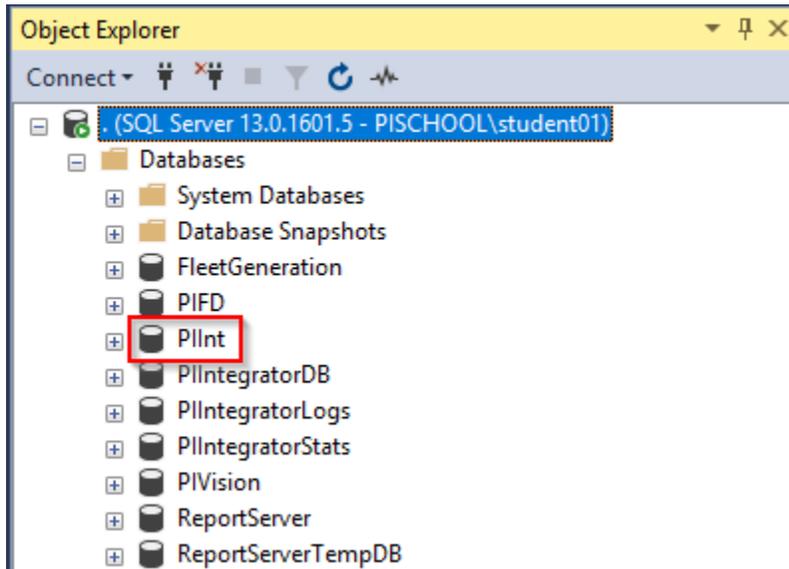
The PI Integrator for BA Configuration database element is:



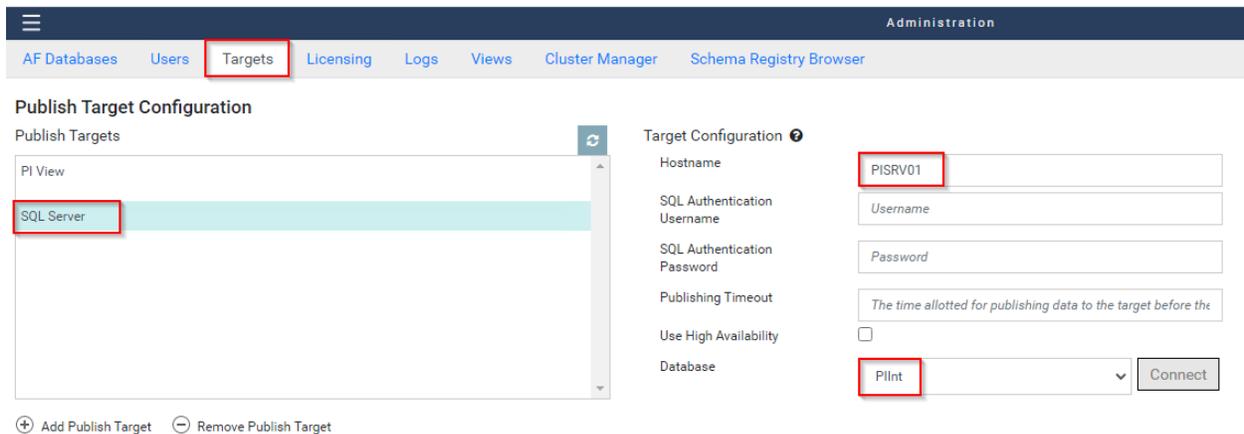
The backend SQL tables are:



The SQL Server Target database is:



The SQL Server target configuration is in the Target section of the Administration page:



In this course we will only configure PI Integrator for BA to publish data to a SQL Server and then use the native SQL Server provider to import the data into Microsoft Power BI.

## PI Integrator Web UI

The PI Integrator for Business Analytics site can be accessed via <https://pisrv01.pischool.int:444> or from the desktop. If prompted for credentials, enter your student account, as this has been given access rights.

Views can be created within the PI Integrator portal that is hosted on PISRV01. A list of previously generated views is present within the portal on the **My Views** page, allowing for previewing and maintenance. These existing views can also be cloned and modified, allowing different views to be created and utilized within BI client tools.

The following is a breakdown of the **My Views** page layout, and the different operations available.

Note: The information regarding the My Views page layout is available within the PI Integrator for Business Analytics User Guide.

The screenshot shows the 'My Views' page with the following components:

- Top Bar:** Contains navigation icons, the title 'My Views', and a user profile 'PISCHOOLstudent01'.
- Action Buttons:** A row of buttons for '+ Create Asset View', '+ Create Event View', '+ Create Streaming View', 'Modify View', and 'Remove View', each with a red circle (2-6) indicating its location.
- Table:** A table with columns: Name, Run Status, Type, Run Mode, Start Time, End Time, and Last Run Time. The first row is 'Distribution Network Sample' with status 'Published' (marked with a red circle 1).
- Bottom Bar:** Contains tabs for 'Overview', 'Log' (with a red circle 8 and a message counter), 'Security', 'View Configuration', and 'Statistics'.
- Run Status Panel (7):** A panel on the left showing details for the selected view, including View Name, PI AF Database, Publish Target, View Type, Run Mode, Last Run Time, and Your Start Time is.
- Asset Shape Panel:** A tree view on the right showing the hierarchy of the selected asset, including Headquarters, Substation, Single Transformer, Circuit, Phase, and Single Phase Transformer.

The My Views page shows details about your views.

1. All the views to which you have access are listed in the table
2. Click to create an **Asset View** that is based on Elements and Element Templates
3. Click to create an **Event View** that is based on Event Frames and Event Frame Templates
4. Click to create a **Streaming View** for publish targets that support streaming such as Apache Kafka, Azure Event Hub, and Azure IoT Hub.
5. To modify a view, select the view in the table and click **Modify View**.
6. To delete it, click **Remove View**. Deleting a view removes data from the buffer, therefore freeing up space. However, this does not free up the available output streams allowed with your license.
7. For the selected view, the Overview, Log and Security tabs provide additional details about the view.
8. The red message counter icon at top right show that there are warning and error messages recorded by PI Integrator for Business Analytics. Click the icon to open the message list.
9. Click the gear icon at top right to see the version of PI Integrator for Business Analytics and AF you are using.

## Activity – Create the Transformer Loading View

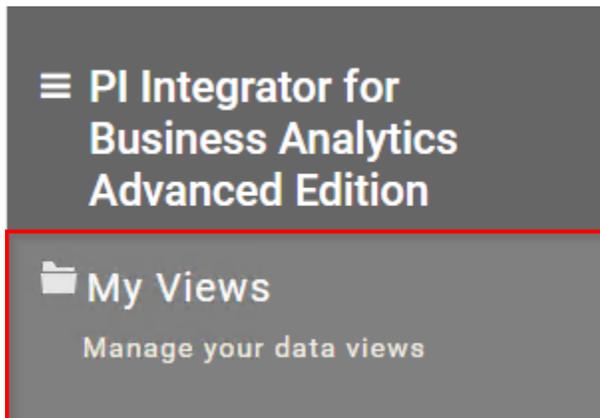
### Objective:

- Use the PI Integrator for Business Analytics to create an Asset View, which will be used in later exercises.

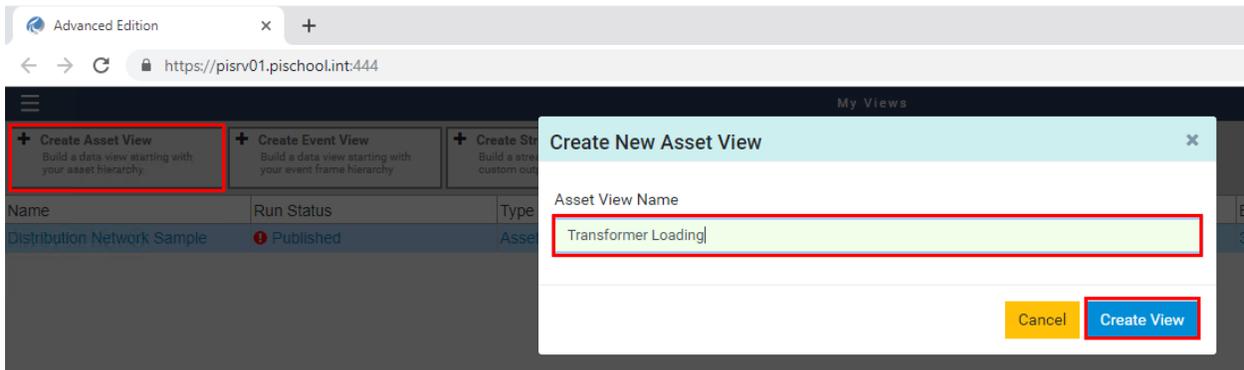
### Approach:

Open Google Chrome and Navigate to the PI Integrator for BA Web UI at <https://pisrv01.pischool.int:444>

Go back to My Views:



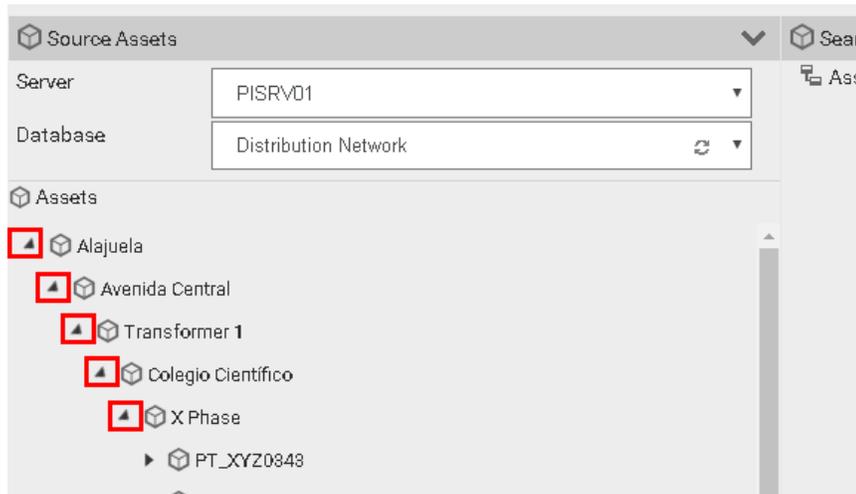
Click **Create Asset View** and name it Transformer Loading, click **Create View**:



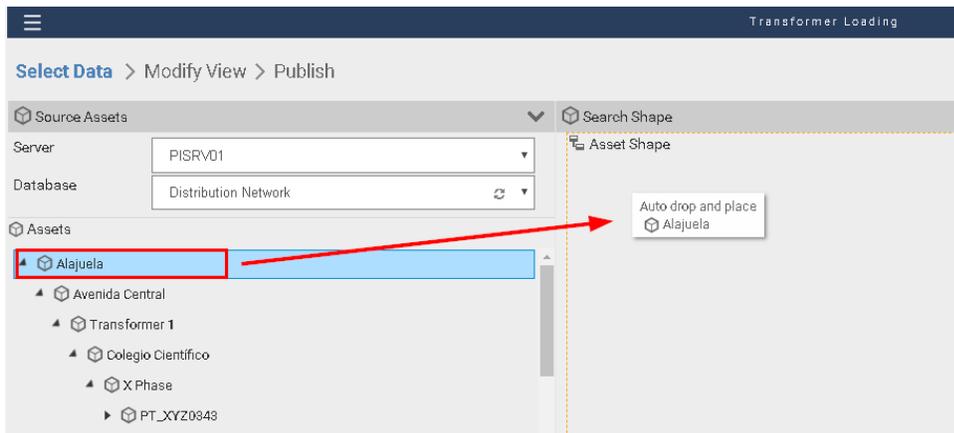
Click **Create a New Shape**



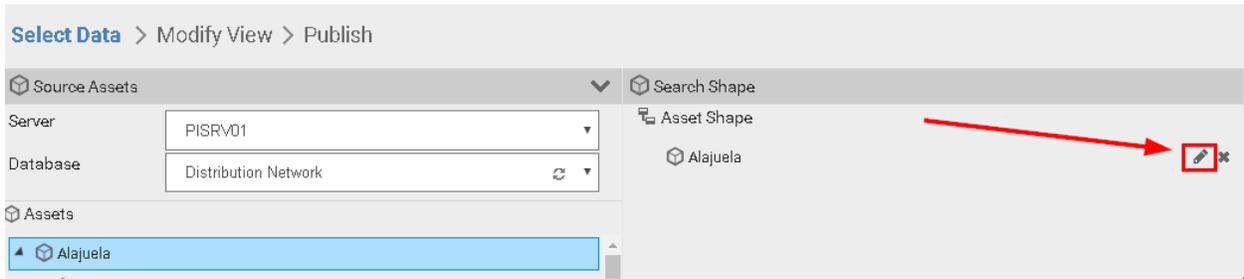
Select Distribution Network as the AF Database, then drill down to PT\_XYZ0343.



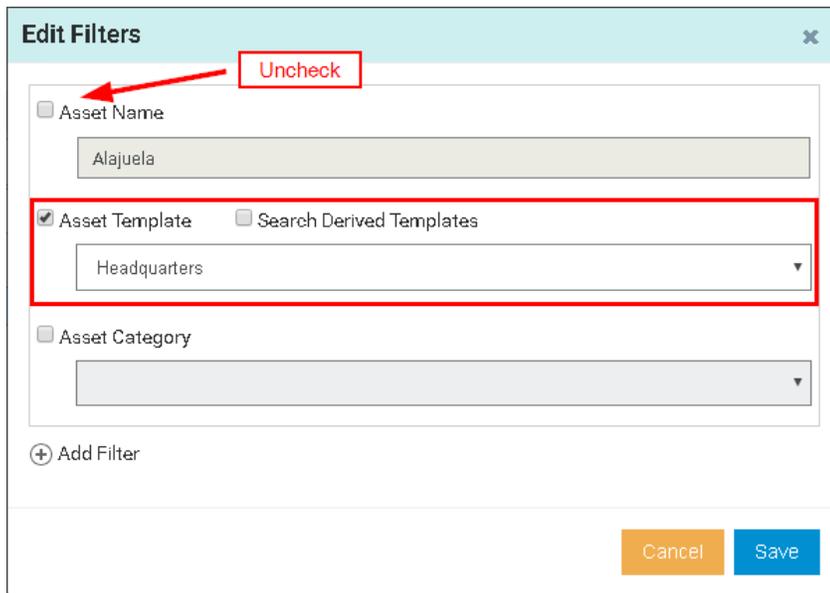
Drag and drop Alajuela to the Shape Builder



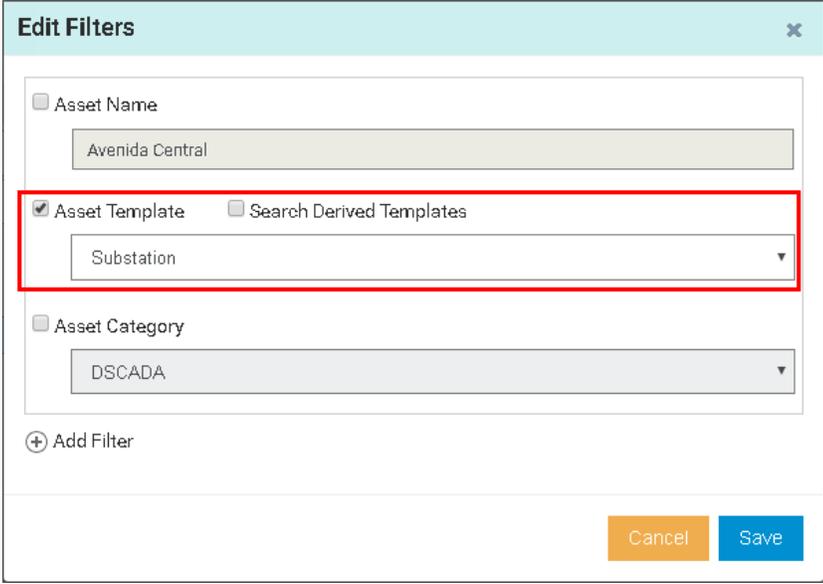
Edit the Filter on Alajuela:



Clear the Asset Name Checkbox, Change it to filter on the Headquarters template, click **Save**:



Drag and drop **Avenida Central** to the Shape configuration, and change it to filter on the **Substation** Template:



The screenshot shows the 'Edit Filters' dialog box with the following configuration:

- Asset Name: Avenida Central
- Asset Template: Substation (highlighted with a red box)
- Search Derived Templates
- Asset Category: DSCADA
- 
- 

Repeat this pattern for **Transformer 1** (Template = Single Transformer), **Colegio Cientifico** (Template = Circuit), **X Phase** (Template = Phase).

Drag and drop **PT\_XYZ0343** and select **Secondary Transformer** as the Template, this time check the box to search derived templates.

**Edit Filters** [X]

Asset Name  
PT\_XYZ0343

Asset Template  Search Derived Templates  
Secondary Transformer

Asset Category  
Single Phase

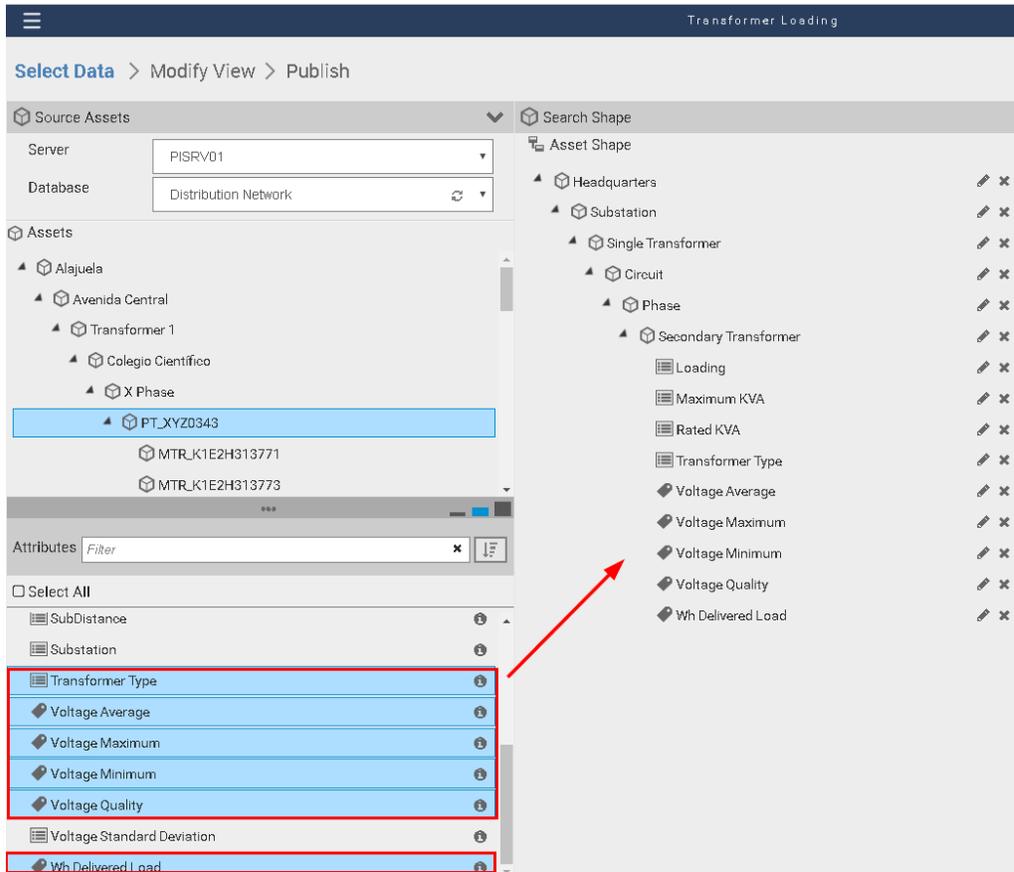
+ Add Filter

Cancel Save

The shape configuration should look like this:

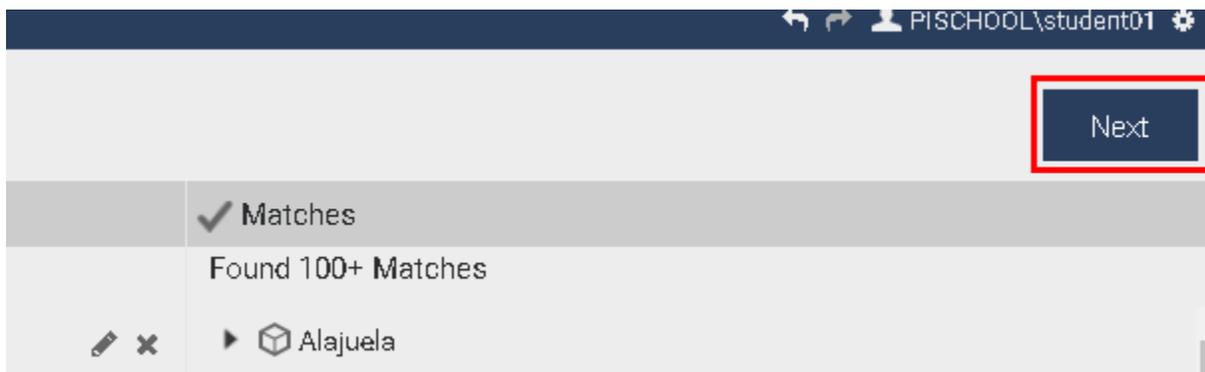


Click **PT\_XYZ0343** then **hold control** and multi-select **Loading, Maximum KVA, Rated KVA, Transformer Type, Voltage Average, Voltage Maximum, Voltage Minimum, Voltage Quality, and Wh Delivered Load**. Drag and drop these selections to the Shape configuration.



Note that all Transformers share these common attributes because they all use the same template.

There should be over 100 matches in the preview, click Next in the top right corner.



We now see a preview of the data using the default Time Range and interpolation mode. Note that each AF attribute is a column in the preview. PI AF templates inherently provide the table format preferred by BI Tools when analyzing a number of similar assets.

Headquarters	TimeStamp	Substation	Single Transformer	Circuit	Phase	Secondary Transformer	Loading	Maximum KVA	Rated KVA	Transformer Type	Voltage Average	Voltage Maximum	Voltage
Alajuela	5/13/2019 6:49	Avenida Centri	Transformer 1	Colegio Cl X Phase		PT_XYZ0381	12.552	31.7	25	POLE	247.196	247.275	247.15
Alajuela	5/13/2019 6:50	Avenida Centri	Transformer 1	Colegio Cl X Phase		PT_XYZ0381	12.552	31.7	25	POLE	247.196	247.275	247.15
Alajuela	5/13/2019 6:51	Avenida Centri	Transformer 1	Colegio Cl X Phase		PT_XYZ0381	12.552	31.7	25	POLE	247.196	247.275	247.15

We want to publish Hourly data for the time period 01-Jun-17 00:00:00 to 31-Aug-17 23:00:00. Modify the Start Time and End Time and click Apply:

Case Transformer	Loading	Maximum KVA	Rated KVA	Transformer Type	Voltage Average	Voltage Maximum	Voltage
0381	11.28	31.7	25	POLE	249.542	249.675	249.425

Click **Edit Value Mode** and change the time step to 1 hour, then **Save Changes**:

**Edit Value Mode**

- Sampled Values
  - Sample values every 1 hours
  - Interpolate
  - Exact
  - Use Key Column: Voltage Average

Buttons: Cancel, Save Changes

The TimeStamp column should now reflect changes to the Start, End, and Value Mode:

Headquarters	TimeStamp	Substation	Single Transformer	Circuit	Phase	Site
Alajuela	6/1/2017 12:00:00 AM	Avenida Central	Transformer 1	Colegio Cientifico	X Phase	P1
Alajuela	6/1/2017 1:00:00 AM	Avenida Central	Transformer 1	Colegio Cientifico	X Phase	P1
Alajuela	6/1/2017 2:00:00 AM	Avenida Central	Transformer 1	Colegio Cientifico	X Phase	P1
Alajuela	6/1/2017 3:00:00 AM	Avenida Central	Transformer 1	Colegio Cientifico	X Phase	P1
Alajuela	6/1/2017 4:00:00 AM	Avenida Central	Transformer 1	Colegio Cientifico	X Phase	P1

Now we'll add some additional time columns that will come in handy later when building the reports. **Click Add Column.** Select the **Time Column** tab. Select Month, Month Name, Week of the Year, and Hour, then click the arrow to bump them over to the right:

**Add Column** ✕

Data Column
Time Column
Static Value

Select Time Column Options for Local

Year(2020)
Month(4)
Month Name(April)
Week of the Year(14)
Day(1)
Day of the Week(Wednesday)
Hour(15)
Minute(19)

←
→

TimeStamp(Local)

Click Display 5 Time Columns:



Now that the time ranges and columns have been specified, click **Next**.



Now we can choose what target to publish to. This depends on the platform used to support front-end application, but for our purposes we'll publish to a SQL Server. Select **SQL Server** for the Target Configuration, Leave Run Once checked, and click **Publish**:

The screenshot shows the 'Transformer' software interface. At the top, there is a navigation bar with a menu icon on the left and the word 'Transformer' on the right. Below this is a breadcrumb trail: 'Select Data > Modify View > Publish'. The main content area is divided into two columns. The left column is titled 'Target Configuration' and contains a dropdown menu with 'SQL Server' selected. Below this is the 'Run Mode' section, which has two radio button options: 'Run Once' (which is selected) and 'Run on a Schedule'. The right column is titled 'Summary' and contains a grey box with the following information: 'Shape and Matches' with a bullet point 'There are 100+ Matching Instances'; 'Timeframe and Interval' with three bullet points: 'Your Start Time is 2017-06-01T00:00:00.000Z', 'Your End Time is 2017-08-31T23:00:00.000Z', and 'Your Time Interval gets an interpolated measurement Every 1 hour'. At the bottom of the right column is a blue 'Publish' button.

It will take a few minutes to publish the data.

## Lesson 4: Building the Distribution Network Reports

We will now spend some time configuring a Microsoft Power BI report. The first step is importing the data.

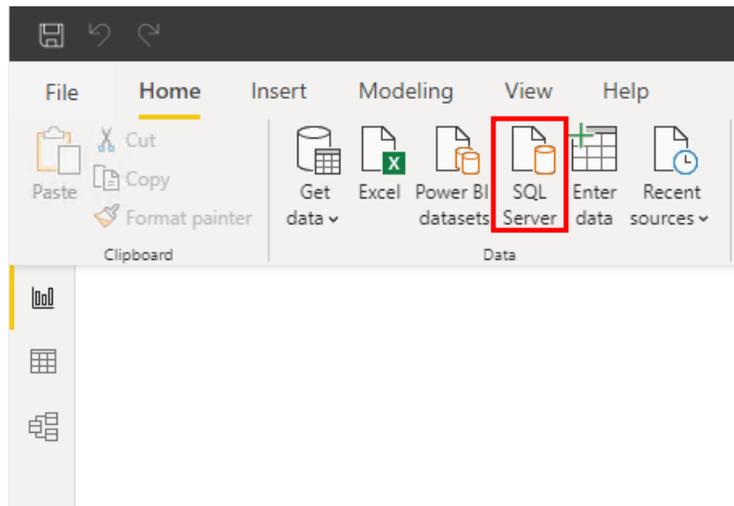
### Importing Data into Power BI

Now that the Transformer Loading table has been published, we will import the SQL table into Power BI.

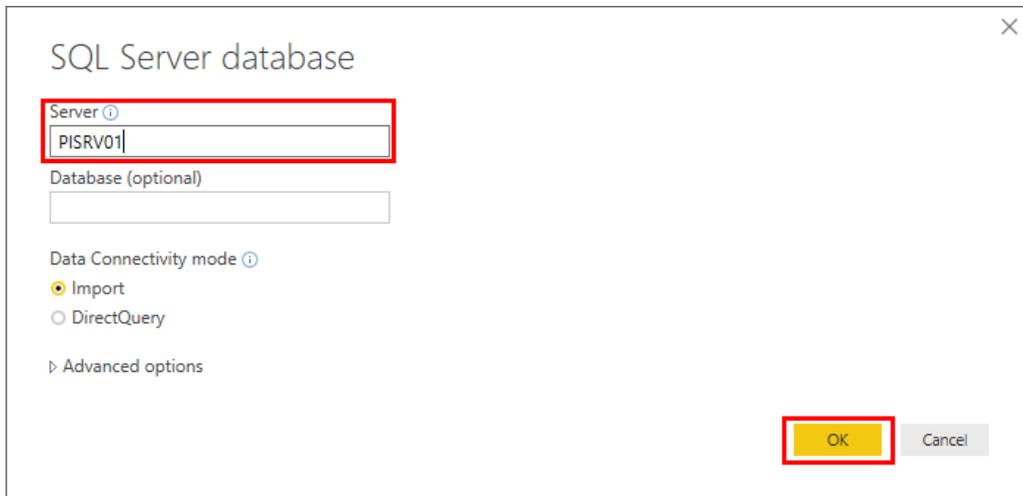
#### Approach:

Open Microsoft Power BI and start a new report.

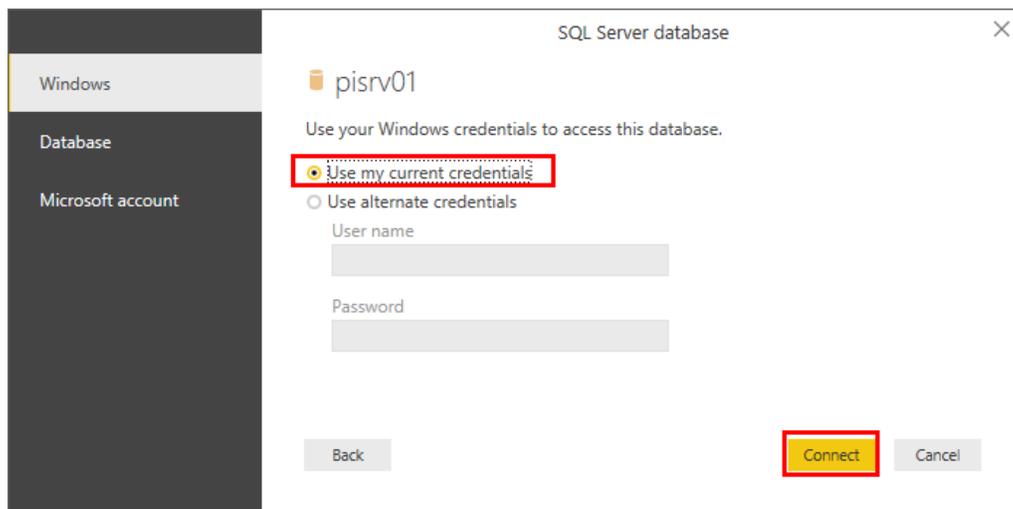
Select **SQL Server** in the Data Group.



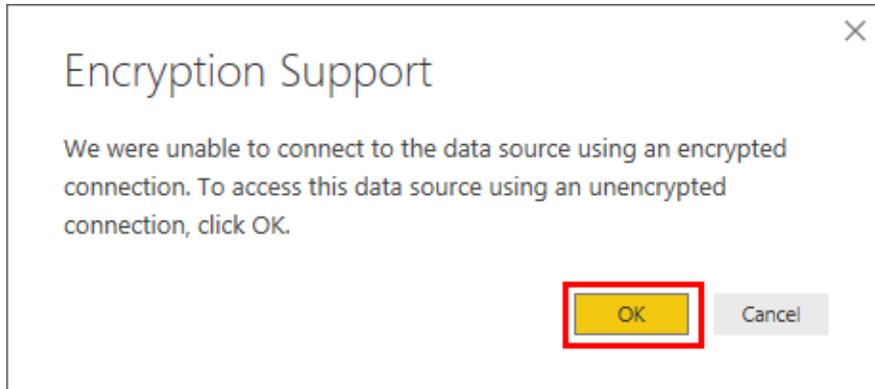
Enter **PISRV01** as the server name and click **OK**.



If Prompted, Leave “use my current credentials” selected and click **Connect**:



There may be a warning that the connection is not encrypted, this can be safely ignored, **click OK**:



Expand the PIInt database and Select the **Transformer Loading** table, click **Load**

Navigator

Display Options

- PISRV01 [9]
  - FleetGeneration
  - PIFD
  - PIInt [1]
    - Transformer Loading**
  - PIIntegratorDB
  - PIIntegratorLogs
  - PIIntegratorStats
  - PIVision
  - ReportServer
  - ReportServerTempDB

Transformer Loading

Id	Headquarters	TimeStamp	Month	Month Name	Week
1	Alajuuela	6/1/2017 12:00:00 AM	6	June	
2	Alajuuela	6/1/2017 1:00:00 AM	6	June	
3	Alajuuela	6/1/2017 2:00:00 AM	6	June	
4	Alajuuela	6/1/2017 3:00:00 AM	6	June	
5	Alajuuela	6/1/2017 4:00:00 AM	6	June	
6	Alajuuela	6/1/2017 5:00:00 AM	6	June	
7	Alajuuela	6/1/2017 6:00:00 AM	6	June	
8	Alajuuela	6/1/2017 7:00:00 AM	6	June	
9	Alajuuela	6/1/2017 8:00:00 AM	6	June	
10	Alajuuela	6/1/2017 9:00:00 AM	6	June	
11	Alajuuela	6/1/2017 10:00:00 AM	6	June	
12	Alajuuela	6/1/2017 11:00:00 AM	6	June	
13	Alajuuela	6/1/2017 12:00:00 PM	6	June	
14	Alajuuela	6/1/2017 1:00:00 PM	6	June	
15	Alajuuela	6/1/2017 2:00:00 PM	6	June	
16	Alajuuela	6/1/2017 3:00:00 PM	6	June	
17	Alajuuela	6/1/2017 4:00:00 PM	6	June	
18	Alajuuela	6/1/2017 5:00:00 PM	6	June	
19	Alajuuela	6/1/2017 6:00:00 PM	6	June	
20	Alajuuela	6/1/2017 7:00:00 PM	6	June	
21	Alajuuela	6/1/2017 8:00:00 PM	6	June	

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

Select Related Tables Load Edit Cancel

Note that about 3.8 million rows have been imported. **This exceeds the 1 million row limit in Microsoft Excel.**

---

## Building the Report Visuals

Now that the Transformer Loading table has been imported, the rest of the chapter will be a walkthrough of configuring various report visuals.

In case there were mistakes or problems with the previous steps, a starter .pbix file has been created with the raw data set already imported with columns that will match the exercises exactly.

Open **C:\Class\Part 1 - PI Integrator for BA\Starter File - Part 1 Distribution Network.pbix** and use this as a starting point for the remaining exercises. Work from the previous exercises can be safely discarded.

### Transformer Loading Analysis

#### Objectives:

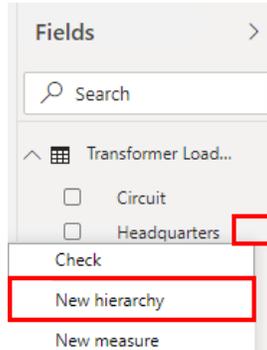
- Configure a **Hierarchy**
- Configure a **Hierarchy Slicer**
- Configure a **Measure** to calculate service hours
- Configure a **Group** to create bins for different load ranges which can then be used for highlighting and filtering
- Configure a **Stacked Bar Chart** to display the service hours spent in each Load Range by circuit
- Configure a **Table** to show the top 20 transformers by average Loading
- Configure a **Slicer** to filter by Month

In the following steps, we will analyze transformer loading characteristics. The goal is to assess the number of service hours spent in various high load conditions to better understand which transformers are at risk of failing and also assess whether a given transformer should be replaced with one that has a higher capacity.

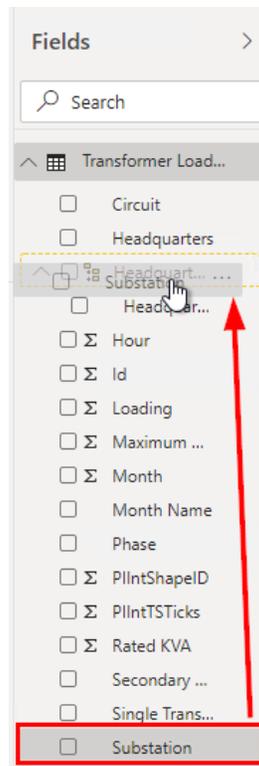
## Approach:

### Configuring the Hierarchy

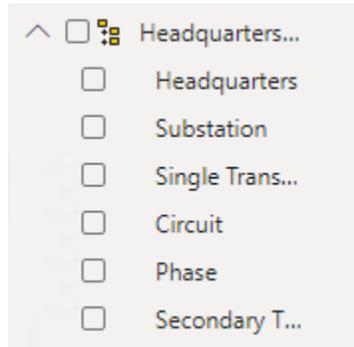
We will now create a hierarchy. In the **Fields List**, click the ellipses next to Headquarters and select **New hierarchy**:



Within the fields list, drag and drop the **Substation** field on top of the new Headquarters hierarchy:



Repeat for **Single Transformer, Circuit, Phase, and Secondary Transformer** and reorder to match the below if necessary.



## Downloading the Hierarchy Slicer

**For this part, there is no need to visit the web site, sign up, or download the file.** We have downloaded the file for use in class so that students do not need to sign up!

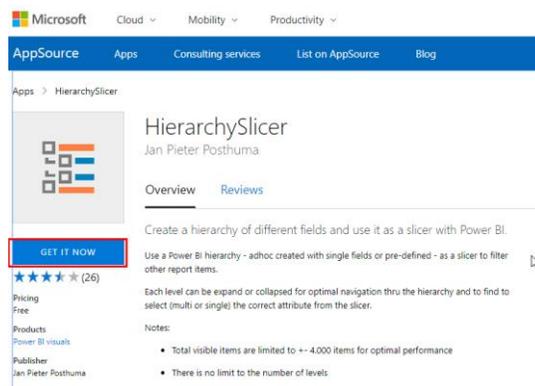
The Hierarchy Slicer is a custom visual that can be used to filter reports and mimic the PI AF hierarchy. This is similar to the PI TreeView from PI WebParts.

Most custom visuals can be found on Microsoft AppSource. We will briefly go through the procedure of how one would normally obtain a custom visual.

Search for a custom visual on Google or within AppSource and you'll arrive at a page like this:

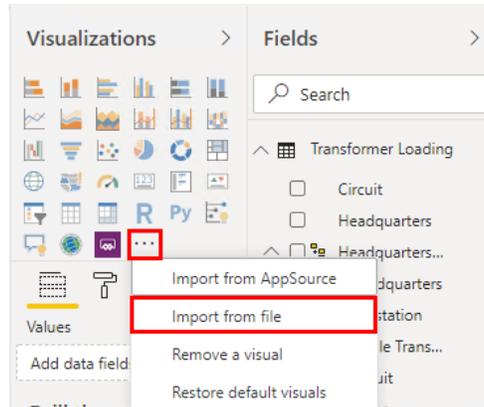
<https://appsource.microsoft.com/en-us/product/power-bi-visuals/WA104380820?tab=Overview>

At which point you would click Get It Now, sign in using your work or school account, and download the .pbviz file.

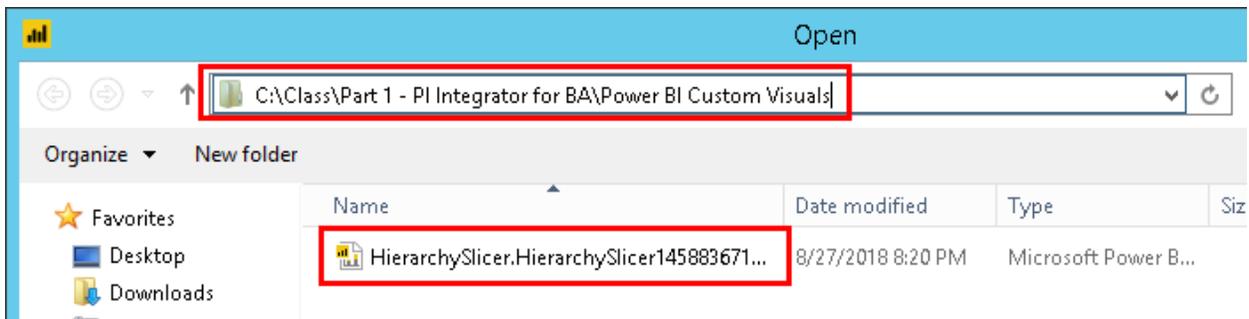


## Importing and Configuring the Hierarchy Slicer

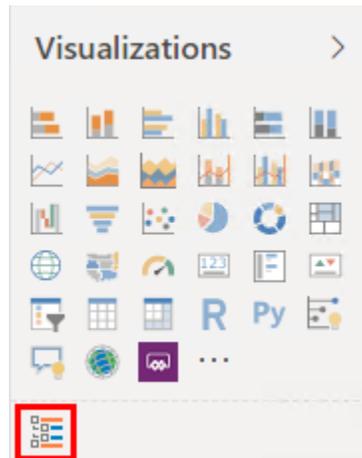
Now it's time to import the custom visual. Open Power BI, click the ellipses within the Visualization Pane, and select Import from file:



Navigate to **C:\Class\Part 1 - PI Integrator for BA\Power BI Custom Visuals** and select the **HierarchySlicer** file.



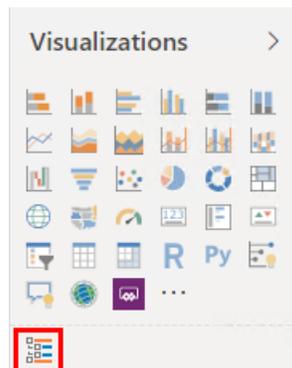
We should now see the Hierarchy Slicer in the list of available visuals:



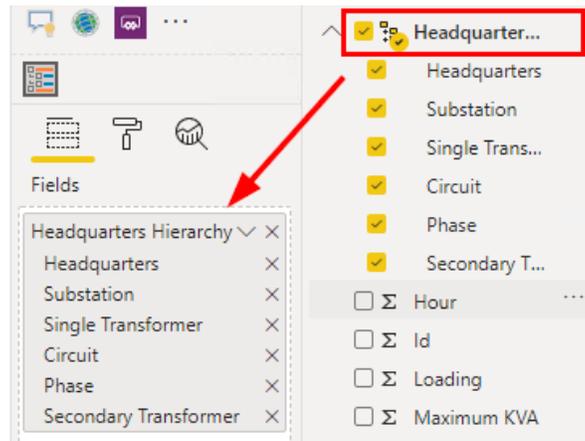
### Mimic PI AF Hierarchy – Hierarchy Slicer

This exercise requires the Hierarchy Slicer custom visual be imported and assumes the Hierarchy has been configured.

We will use a Hierarchy Slicer to leverage the existing PI AF hierarchy for filtering. Add a Hierarchy Slicer by clicking the icon:

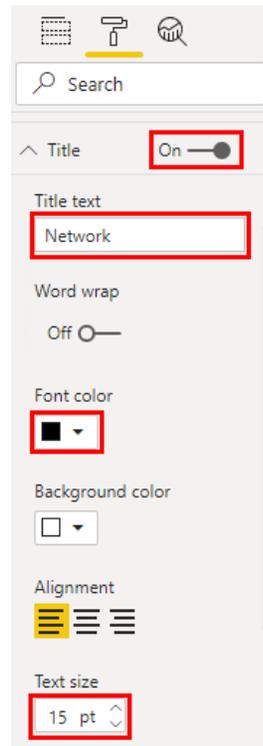


Drag and drop the **Hierarchy** to the visual fields:



Experiment with the Hierarchy Slicer for a bit by drilling down through the levels. Note that checking a box for a parent will also include the children. This is a great way to visualize how filtering works in Power BI.

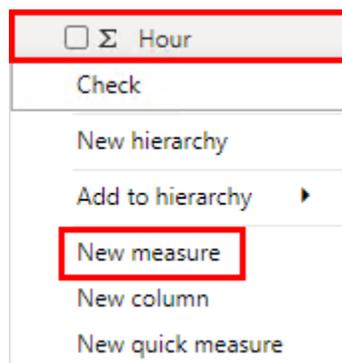
Change the Title of the Hierarchy Slicer to Network in the formatting options. Change the color and increase the text size.



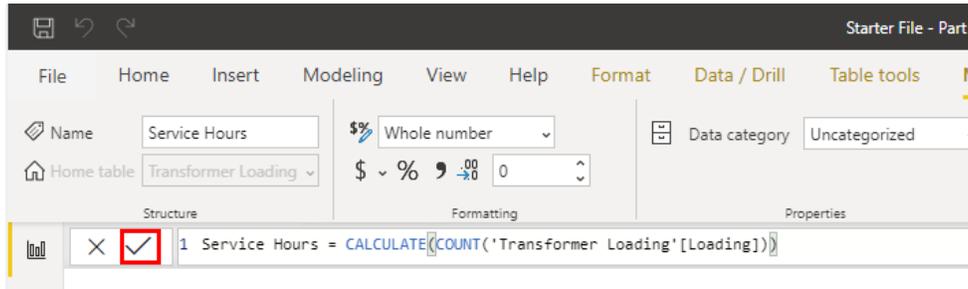
### Service Hours

Now we'll configure a Measure to calculate service hours. Each row in the data set represents 1 hour, so we can simply count the number of rows that have been filtered through user selection. This should make a bit more sense when it all comes together.

Right click **ANY** of the fields from the Fields list and select **New measure**:



Enter the below formula into the configuration box and hit **Enter** or click the **Checkmark**:



The raw text is given below for convenience.

`Service Hours = CALCULATE(COUNT('Transformer Loading'[Loading]))`

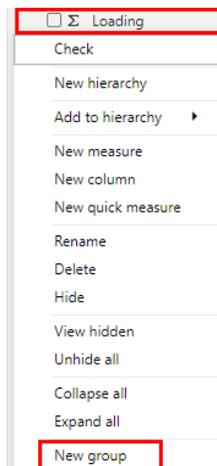
From a configuration perspective, Measures and Calculated Columns are configured similarly so the distinction may not be obvious.

Measures and calculated columns both use DAX expressions. The difference is the context of evaluation. A measure is evaluated on the fly using a subset of data, whereas a calculated column is pre-calculated at the row level within the table to which it belongs. A simple way to put it is that Measures take into account the filtering that has been set by the end user of the report (the stuff they've clicked on), while calculated columns are computed row by row and are not influenced by the report filtering.

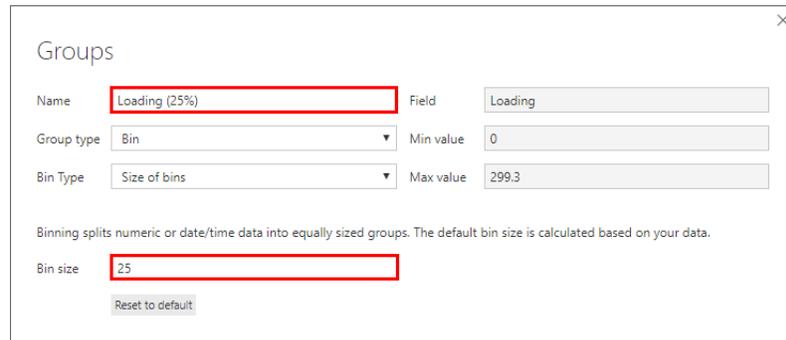
### Loading Groups

Different ranges for Loading will be grouped into bins representing different Load Ranges. It is normal for transformers to be operating at Loads higher than 100% of their rating, but loads in the range of 125% and higher are potentially cause for concern. In order to calculate service hours in the different Load Ranges, a group must be configured in the data set for filtering and counting by the Service Hours Measure.

Right click on **Loading** and select **New group**.



Change the name to **Loading (25%)** and set the bin size to 25, then click **OK**.



Groups

Name  Field

Group type  Min value

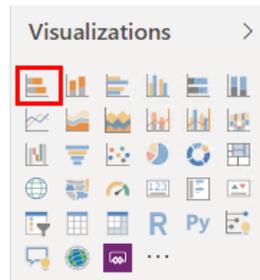
Bin Type  Max value

Binning splits numeric or date/time data into equally sized groups. The default bin size is calculated based on your data.

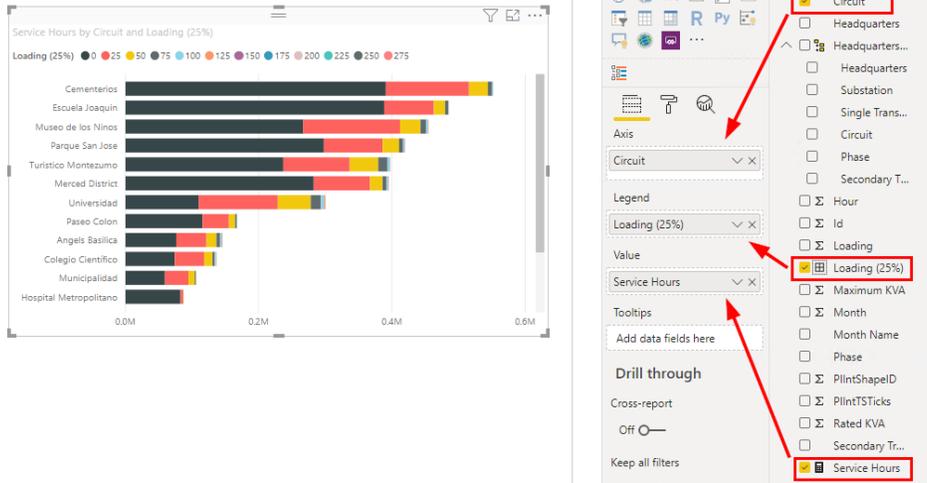
Bin size

## Loading by Circuit – Stacked Bar Chart

Now we can begin to configure the report. **Click some empty space** and then **click the Stacked Bar Chart icon**:

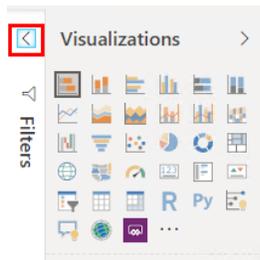


With the Stacked Bar Chart selected, drag and drop Fields from the data set into the field configuration boxes. Use **Circuit** for the Axis, **Loading (25%)** for the Legend, and **Service Hours** for the Value:

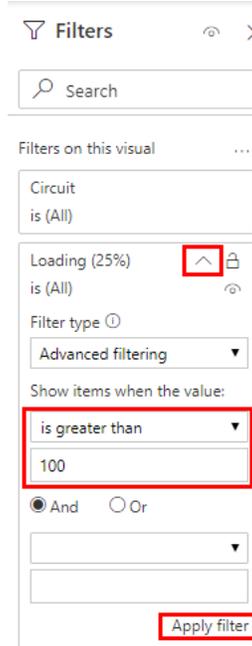


Next we will apply some formatting and filters to make the data set more manageable. We'll change the color scheme and only show Loadings greater than 100%, since loads in the normal range are not of interest to us.

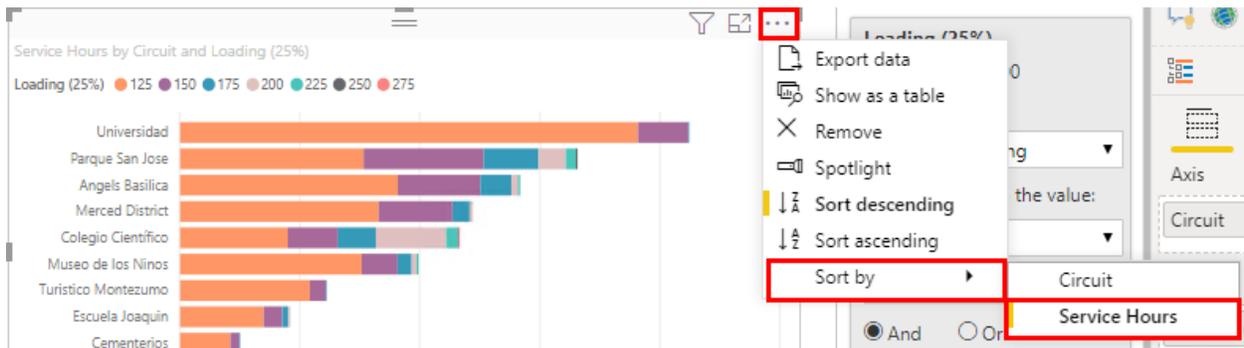
Expand the Filters Pane:



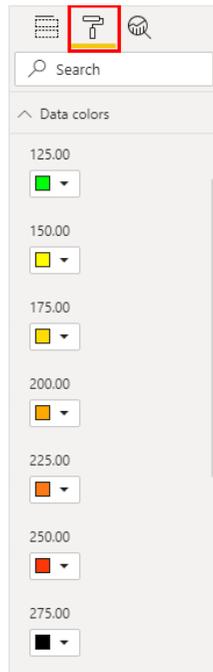
Filter for Loading **greater than 100%**. Be sure to click **Apply Filter**:



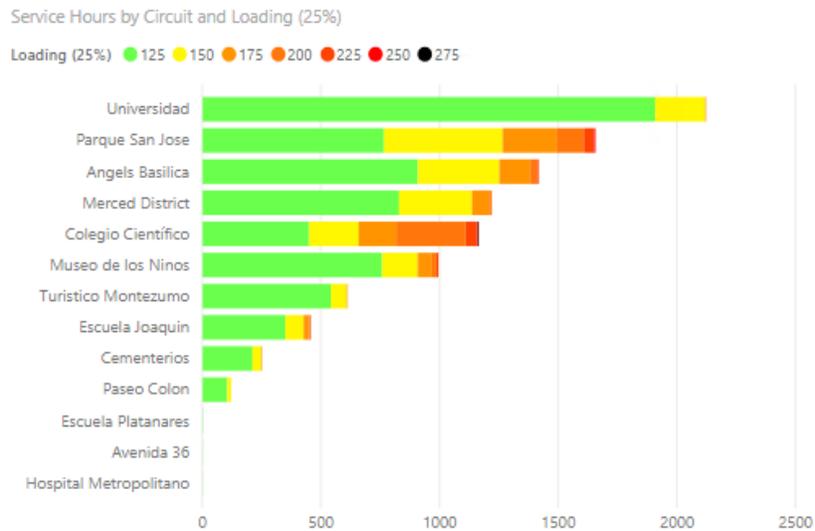
Next go to the Visualization Options and **sort by Service Hours** (done by default in this version of Power BI):



Next change the color scheme. With the Visualization selected, click the Format Icon in the Visualization Pane and adjust the colors to better convey the severity of the loading levels.



The stacked bar chart should now look something like this:

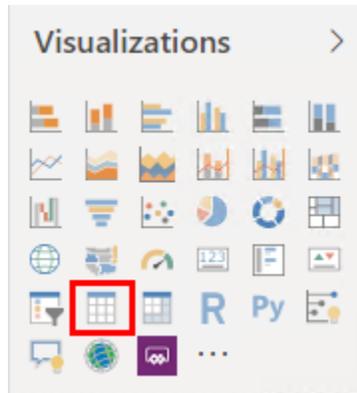


### Service Hours and Average Load by Transformer – Table

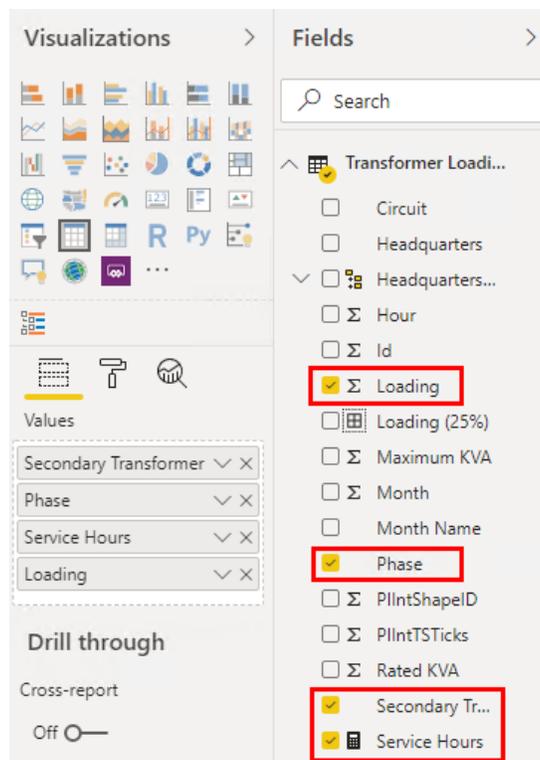
The next visual we will add is a basic table showing the Transformer Name, Phase, Service Hours, and Average Load. We will then filter the table to show only the top 10 transformers by average load. This will give us a quick indicator of which Transformers are consistently overloaded.

Click some blank space on the canvas to deselect any visuals, otherwise you will accidentally convert the Stacked Bar Chart to a Table.

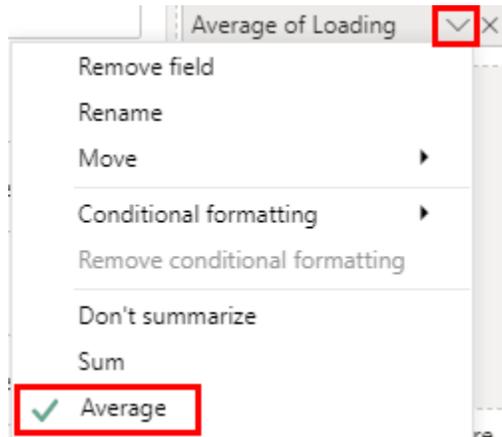
Create a **Table**:



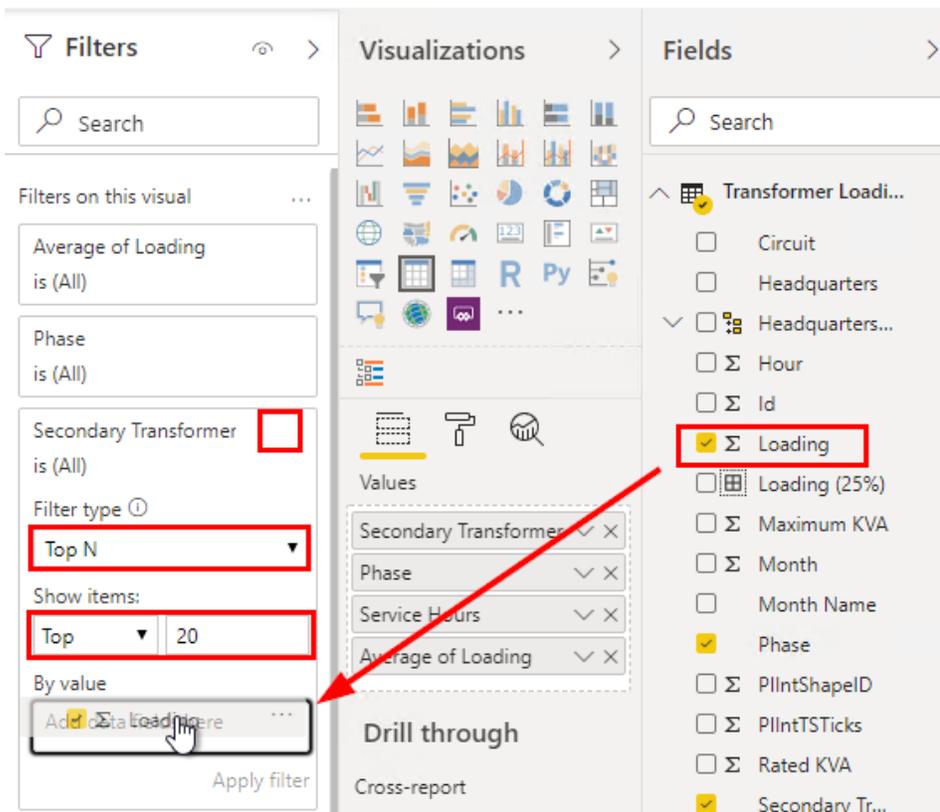
Drag and drop the **Secondary Transformer, Phase, Service Hours, and Loading** Fields into the Values section:



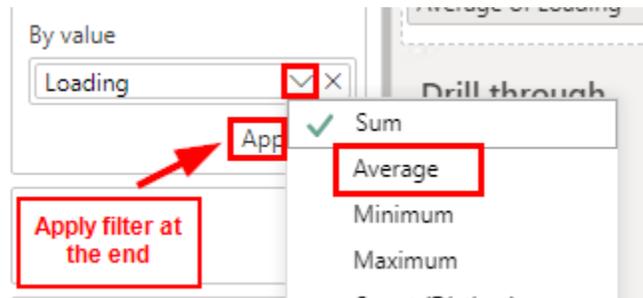
Change the **Loading** Value to summarize by **Average**:



Change the Visual Level Filters to Show the **Top 20** Transformers by **Loading**.

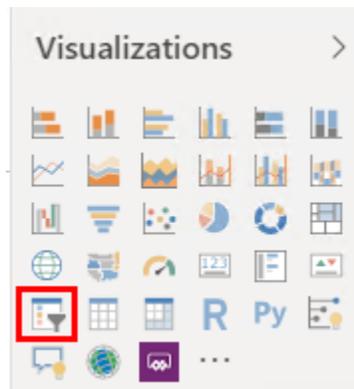


Change to summarize **Loading as Average**, then be sure to click **Apply filter**.



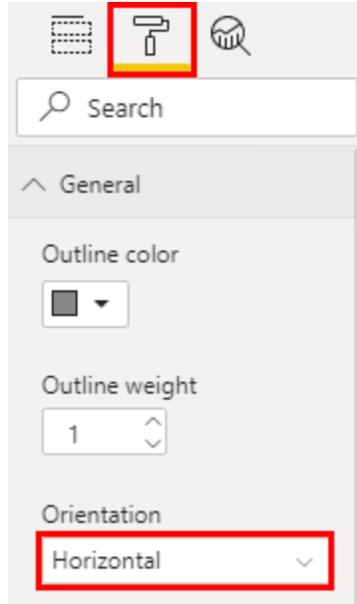
### Filtering by Month – Slicer

We'll now add a basic Slicer to filter by Month. **Click some blank space** and then **add a Slicer**:



Drag **Month Name** to the field list.

Go into the formatting options and change the orientation to **horizontal** to change the look of the Slicer.



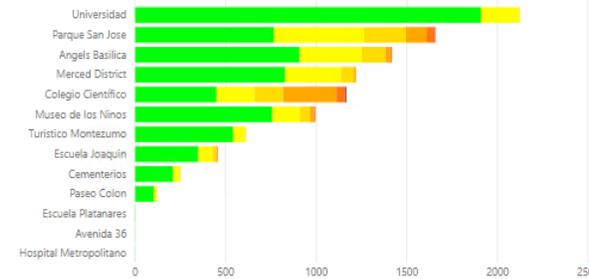
**Reposition & Resize the slicer so all months are in a single row. Reposition & Resize the table and stacked bar chart:**

Network

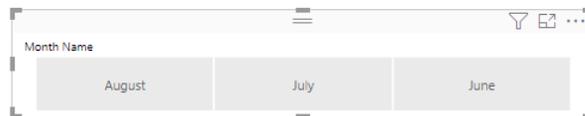
- Headquarters
  - Alajuela
  - Cartago
  - Heredia
  - San Jeronimo
  - San Jose

Service Hours by Circuit and Loading (25%)

Loading (25%) ● 125 ● 150 ● 175 ● 200 ● 225 ● 250 ● 275



Secondary Transformer	Phase	Service Hours	Average of Loading
PT_XVZ0063	X Phase	2208	78.23
PT_XVZ0065	Y Phase	2208	78.01
PT_XVZ0071	Y Phase	2208	95.68
PT_XVZ0096	X Phase	2208	116.99
PT_XVZ0109	X Phase	2208	130.50
PT_XVZ0126	X Phase	2208	82.92
PT_XVZ0195	Y Phase	2208	80.18
PT_XVZ0210	X Phase	2208	79.63
PT_XVZ0254	X Phase	2208	81.08
PT_XVZ0377	Z Phase	2208	117.11
PT_XVZ0410	Z Phase	2208	92.94
PT_XVZ0428	Z Phase	2208	81.24
PT_XVZ0566	Y Phase	2208	96.66
PT_XVZ0587	X Phase	2208	79.53
PT_XVZ0589	X Phase	2208	81.63
PT_XVZ0608	X Phase	2208	76.36
PT_XVZ0644	X Phase	2208	87.80
PT_XVZ0884	Z Phase	2208	103.06
PT_XVZ0911	X Phase	2208	122.20
<b>Total</b>		<b>44160</b>	<b>92.26</b>



To put the Months in chronological order, we will sort the Month Name column in the data set by the Month column where the months are numbered. Go to the **Data View** and click one of the fields to make the data show up:

Starter File - Part 1 Distribution Network - Power BI Desktop

File Home Help Table tools Column tools

Paste Get data Refresh Transform data Manage relationships New measure New column New table Quick measure View as Manage roles Publish

Id	Headquarters	TimeStamp	Month	Month Name	Week of the Year	Hour	Substation	Sing
815497	Heredia	7/2/2017 12:00:00 AM	7	July		27	0 San Pablo	Tr
815498	Heredia	7/2/2017 1:00:00 AM	7	July		27	1 San Pablo	Tr
815499	Heredia	7/2/2017 2:00:00 AM	7	July		27	2 San Pablo	Tr
815500	Heredia	7/2/2017 3:00:00 AM	7	July		27	3 San Pablo	Tr
815501	Heredia	7/2/2017 4:00:00 AM	7	July		27	4 San Pablo	Tr
815502	Heredia	7/2/2017 5:00:00 AM	7	July		27	5 San Pablo	Tr
815503	Heredia	7/2/2017 6:00:00 AM	7	July		27	6 San Pablo	Tr
815504	Heredia	7/2/2017 7:00:00 AM	7	July		27	7 San Pablo	Tr
815505	Heredia	7/2/2017 8:00:00 AM	7	July		27	8 San Pablo	Tr
815506	Heredia	7/2/2017 9:00:00 AM	7	July		27	9 San Pablo	Tr

Fields

Search

Transformer Load...

Circuit

Headquarters

Headquarters ...

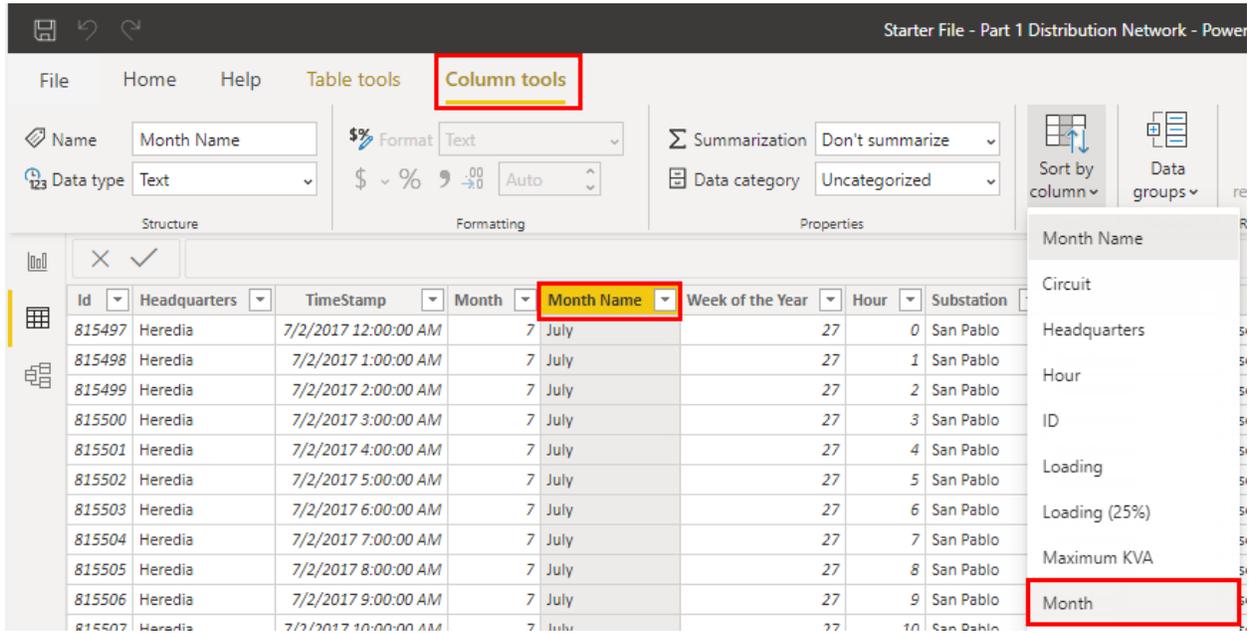
Headquarters

Substation

Single Transfor...

Initially this will be blank, click one of the fields and the data will show up

Select the **Month Name** column, open the **Column Tools** Ribbon, and **Sort by Column -> Month**:



The report should now look something like this:



Sort the table by Average of Loading:

Secondary Transformer	Phase	Service Hours	Average of Loading
PT_XYZ0063	X Phase	2208	78.23
PT_XYZ0065	Y Phase	2208	78.01
PT_XYZ0071	Y Phase	2208	95.68
PT_XYZ0096	X Phase	2208	116.99
PT_XYZ0109	X Phase	2208	130.50
PT_XYZ0126	X Phase	2208	82.92
PT_XYZ0195	Y Phase	2208	80.18
PT_XYZ0210	X Phase	2208	79.63
PT_XYZ0254	X Phase	2208	81.08
PT_XYZ0377	Z Phase	2208	117.11
PT_XYZ0410	Z Phase	2208	92.94
PT_XYZ0428	Z Phase	2208	81.24
PT_XYZ0566	Y Phase	2208	96.66
PT_XYZ0587	X Phase	2208	79.53
PT_XYZ0589	X Phase	2208	81.63
PT_XYZ0608	X Phase	2208	76.36

Click the bars on the Loading by Circuit chart and the Month slicer buttons and note how the service hours and transformers for that load range update on the table.

We will save formatting until the end in case we need to save time, but feel free to adjust the formatting and add a title.

### Linking to PI Vision

We have a PI Vision display for Transformers that we can link to from this report. We will utilize PI Vision URL Parameters to set the same Transformer in the PI Vision display that the user clicks on in the Power BI report. The URL parameters reference guide can be found in the [PI Live Library](#).

From within the client virtual machine, Navigate to:

<https://pisrv01.pischool.int/PIVision/#/Displays/2/TransformerTrends>

Take the above URL and append the following string to it in a text editor, then paste the URL into Chrome:

[?Asset=\\PISRV01\Distribution Network\Secondary Transformers\PT\\_XYZ0046](#)

Transformer PT\_XYZ0046 should be the selected Asset in the TransformerTrends display.

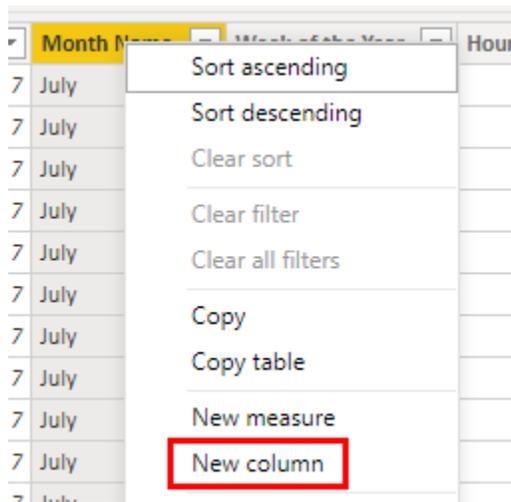
Note that the **?Asset** parameter denotes the path to the Asset in the PI AF hierarchy.

Once that is working, configure a Calculated Column to concatenate the URL with the Transformer asset path.

Go to the **Data** Tab:

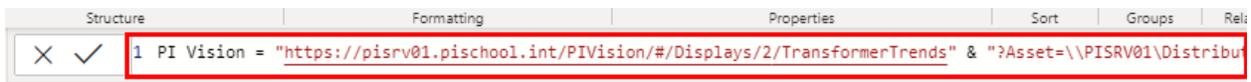
Id	Headquarters	TimeStamp	Month	Month Name	Week of the Year
815497	Heredia	7/2/2017 12:00:00 AM	7	July	27
815498	Heredia	7/2/2017 1:00:00 AM	7	July	27
815499	Heredia	7/2/2017 2:00:00 AM	7	July	27
815500	Heredia	7/2/2017 3:00:00 AM	7	July	27
815501	Heredia	7/2/2017 4:00:00 AM	7	July	27
815502	Heredia	7/2/2017 5:00:00 AM	7	July	27
815503	Heredia	7/2/2017 6:00:00 AM	7	July	27
815504	Heredia	7/2/2017 7:00:00 AM	7	July	27
815505	Heredia	7/2/2017 8:00:00 AM	7	July	27
815506	Heredia	7/2/2017 9:00:00 AM	7	July	27
815507	Heredia	7/2/2017 10:00:00 AM	7	July	27
815508	Heredia	7/2/2017 11:00:00 AM	7	July	27

Right click on the header of **ANY** column and select **New column**:



For the DAX formula, enter the following and **hit enter or click the checkmark**:

PI Vision = "https://pisrv01.pischool.int/PIVision/#/Displays/2/TransformerTrends" & "?Asset=\\PISRV01\\Distribution Network\\Secondary Transformers\\" & 'Transformer Loading'[Secondary Transformer]



Next scroll all the way to the right and find the **PI Vision column**, then select it.

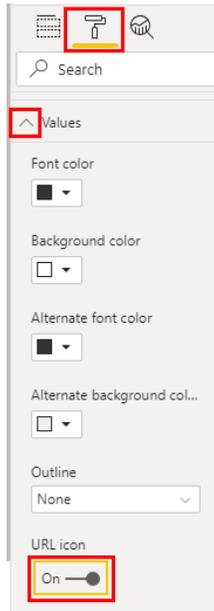
Go to the **Column Tools** ribbon, and change the **Data Category** to Web URL.

Age Average	Voltage Maximum	Voltage Mi
241.25	241.25	
12.149993896484	242.149993896484	242.149
11.199996948242	241.199996948242	241.199
241.875	241.875	
11.225006103516	241.225006103516	241.225
10.050003051758	240.050003051758	240.050
19.850006103516	239.850006103516	239.850
19.350006103516	239.350006103516	239.350
19.574996948242	239.574996948242	239.574
18.824996948242	238.824996948242	238.824
17.449996948242	237.449996948242	237.449
17.949996948242	237.949996948242	237.949
17.074996948242	237.074996948242	237.074

Now go back to the **Report Tab** and select the Table, then drag and drop the **PI Vision** field as one of the table values

The links are now displayed, and they work, but they are not pretty to look at. Luckily, Power BI has a feature that addresses this.

Go into the **Formatting Options**, scroll down to the Values section, and turn on the URL icon:



Now the links look much cleaner:

Secondary Transformer	Phase	Service Hours	Average of Loading	PI Vision
PT_XYZ0109	X Phase	2208	130.50	<a href="#">PI Vision</a>
PT_XYZ0911	X Phase	2208	122.20	<a href="#">PI Vision</a>
PT_XYZ0377	Z Phase	2208	117.11	<a href="#">PI Vision</a>
PT_XYZ0096	X Phase	2208	116.99	<a href="#">PI Vision</a>
PT_XYZ0884	Z Phase	2208	103.06	<a href="#">PI Vision</a>
PT_XYZ0566	Y Phase	2208	96.66	<a href="#">PI Vision</a>
PT_XYZ0071	Y Phase	2208	95.68	<a href="#">PI Vision</a>
PT_XYZ0410	Z Phase	2208	92.94	<a href="#">PI Vision</a>
PT_XYZ0644	X Phase	2208	87.80	<a href="#">PI Vision</a>
PT_XYZ1470	X Phase	2208	83.37	<a href="#">PI Vision</a>
PT_XYZ0126	X Phase	2208	82.92	<a href="#">PI Vision</a>
PT_XYZ0589	X Phase	2208	81.63	<a href="#">PI Vision</a>
PT_XYZ0428	Z Phase	2208	81.24	<a href="#">PI Vision</a>
PT_XYZ0254	X Phase	2208	81.08	<a href="#">PI Vision</a>
PT_XYZ0195	Y Phase	2208	80.18	<a href="#">PI Vision</a>
PT_XYZ0210	X Phase	2208	79.63	<a href="#">PI Vision</a>
PT_XYZ0587	X Phase	2208	79.53	<a href="#">PI Vision</a>
PT_XYZ0063	X Phase	2208	78.23	<a href="#">PI Vision</a>
PT_XYZ0065	Y Phase	2208	78.01	<a href="#">PI Vision</a>
PT_XYZ0608	X Phase	2208	76.36	<a href="#">PI Vision</a>
<b>Total</b>		<b>44160</b>	<b>92.26</b>	

Test the links to confirm that the PI Vision display is launched and the correct transformer is set.

## Lesson 5: Building the Fleet Generation Report

Now rather than a one-time export let's build a report that pulls current data upon refresh. To do so we'll configure a continuous Asset View that updates on a schedule. We'll be using a different AF Database this time: Online Fleet Generation.

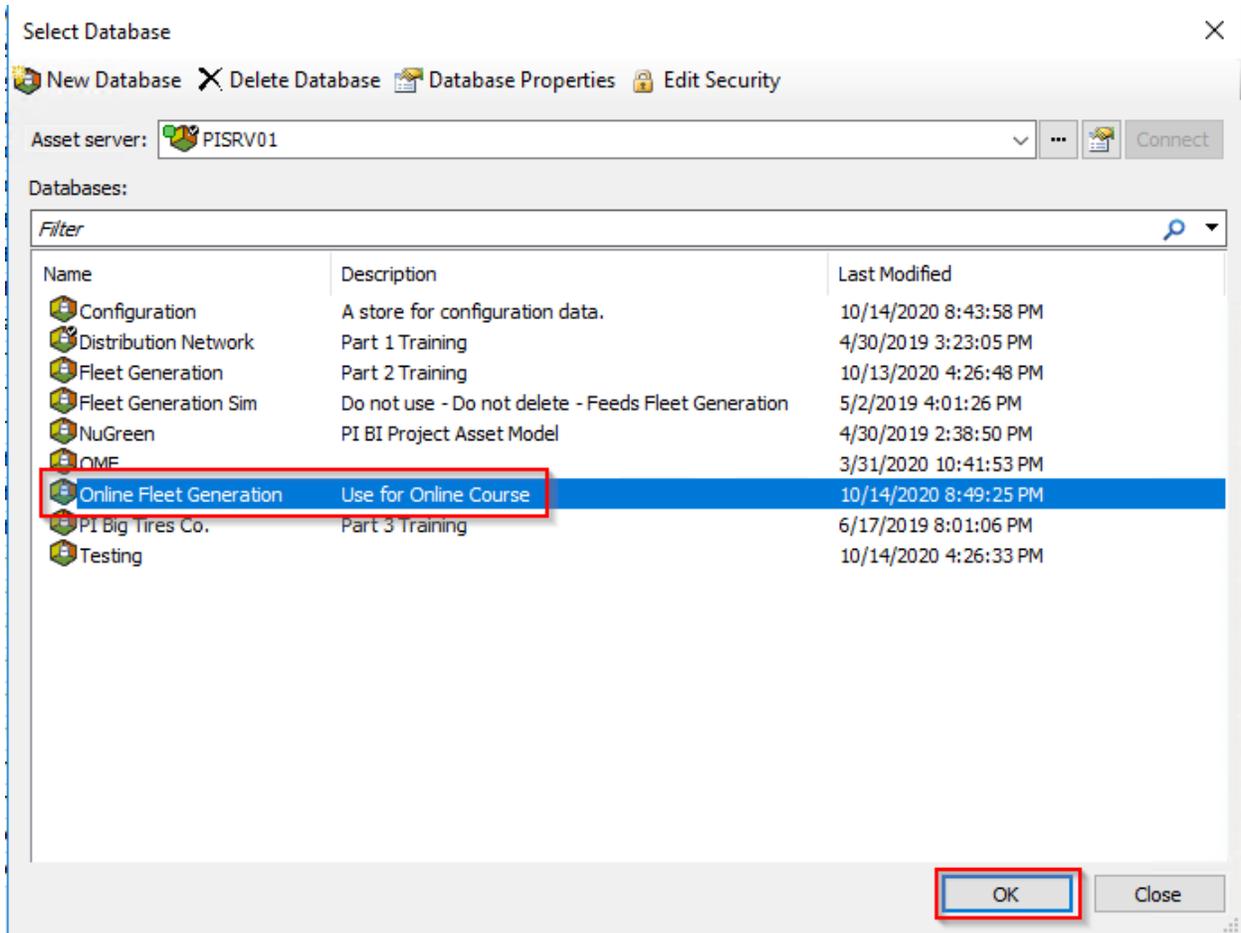
### The Online Fleet Generation Database

We wish to analyze a number of KPIs for several generating units in a fictitious power generation company. All data is simulated / random.

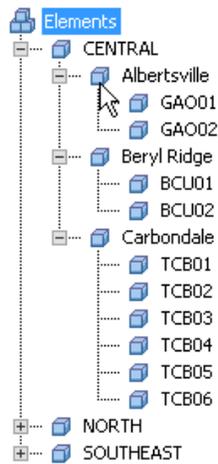
Open **PI System Explorer** and navigate to the **Online Fleet Generation database**.

*The regular Fleet Generation database is used for the classroom version of this course where PI SQL Client is part of the core material.*

*The Fleet Generation Sim database generates data for both the Fleet Generation and Online Fleet Generation databases.*



Browse the hierarchy, which is organized into Region, Station, and Unit.



Most of the child elements are based on the generic **Unit template**.

The screenshot displays the 'UNIT' template configuration in the software. The left pane shows a tree view with 'UNIT' highlighted. The right pane shows a table of attributes for the UNIT template.

Name	Description	Default Value
Category: <None>		
Carbon Emissions		0 g/kWh
Generating Efficiency		0 %
Generation Rate		0 \$/kWh
Total Hourly Gross Generation		0 MWhr
Utilization		0 %
Category: Demand		
Demand		0 MW
Category: Hourly Generation		
Gross Generation		0 MW
Net Generation		0 MW
Category: Identity		
Hourly Capacity		0
Operator		
Shift		0
Shift Hours	Number of Hours in the shift ...	0 h
Technology		0
Category: Status		
Unit Status		

Those in the CENTRAL region are based on the **Gas Turbine template**, which is derived from the UNIT template and has additional attributes.

The screenshot displays the 'Gas Turbine' template configuration in the software. The left pane shows a tree view with 'Gas Turbine' highlighted. The right pane shows a table of attributes for the Gas Turbine template.

Name	Description	Default Value
Category: <None>		
Exhaust Gas Temperature - #1 Probe	Exhaust Gas Temper...	0 °C
Exhaust Gas Temperature - #2 Probe	Exhaust Gas Temper...	0 °C
Gas Fuel Flow	Gas Fuel Flow	0 US gal/min
Gas Fuel Pressure	Gas Fuel Pressure	0 bar
Gas Turbine Speed	Gas Turbine Speed	0 rpm

Gas Turbines have all the attributes from the Gas Turbine template, but also inherit those from the UNIT Template:

The screenshot shows a software interface with a tree view on the left and a detailed attribute table for element GAO01 on the right. The tree view shows a hierarchy: CENTRAL > Albertsville > GAO01. The attribute table is titled 'GAO01' and has tabs for 'General', 'Child Elements', 'Attributes', 'Ports', 'Analyses', 'Notification Rules', and 'Version'. The table is filtered and shows the following data:

Name	Value
<b>Template: UNIT</b>	
Carbon Emissions	405 g/kWh
Demand	92.987 MW
Generating Efficiency	90.909 %
Generation Rate	0.078 \$/kWh
Gross Generation	324.25 MW
Hourly Capacity	550
Net Generation	294.25 MW
Operator	BSX
Shift	3
Shift Hours	8 h
Technology	Natural Gas
Total Hourly Gross Generation	335.75 MWhr
Unit Status	Active
Utilization	61.045 %
<b>Template: Gas Turbine</b>	
Exhaust Gas Temperature - ...	33.313 °C
Exhaust Gas Temperature - ...	32.956 °C
Gas Fuel Flow	66.304 US gal/min
Gas Fuel Pressure	41.766 bar
Gas Turbine Speed	56.658 rpm

## Preparing and Importing the Tables

For the report, we are going to separate the time-series data from the static data and configure table relationships to join the data sets together. Technically, we could design the Asset View such that the result set is a single table. However, in real life not all of the data is always in PI and several data sources must be joined together. A table with one row per unit will also be required for the Final Challenge in order to join Event Frames and geospatial information.

---

## Activity – Publish Unit Specifications (static data) Table

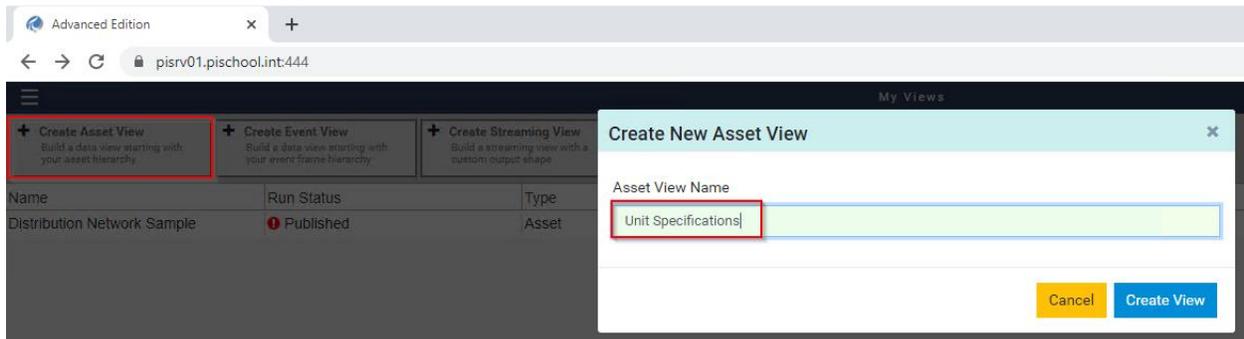
### Objective:

- Publish an Asset View containing static attribute data

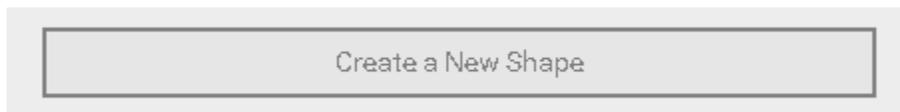
### Approach:

Open Google Chrome and Navigate to the PI Integrator for BA Web UI at <https://pisrv01.pischool.int:444>

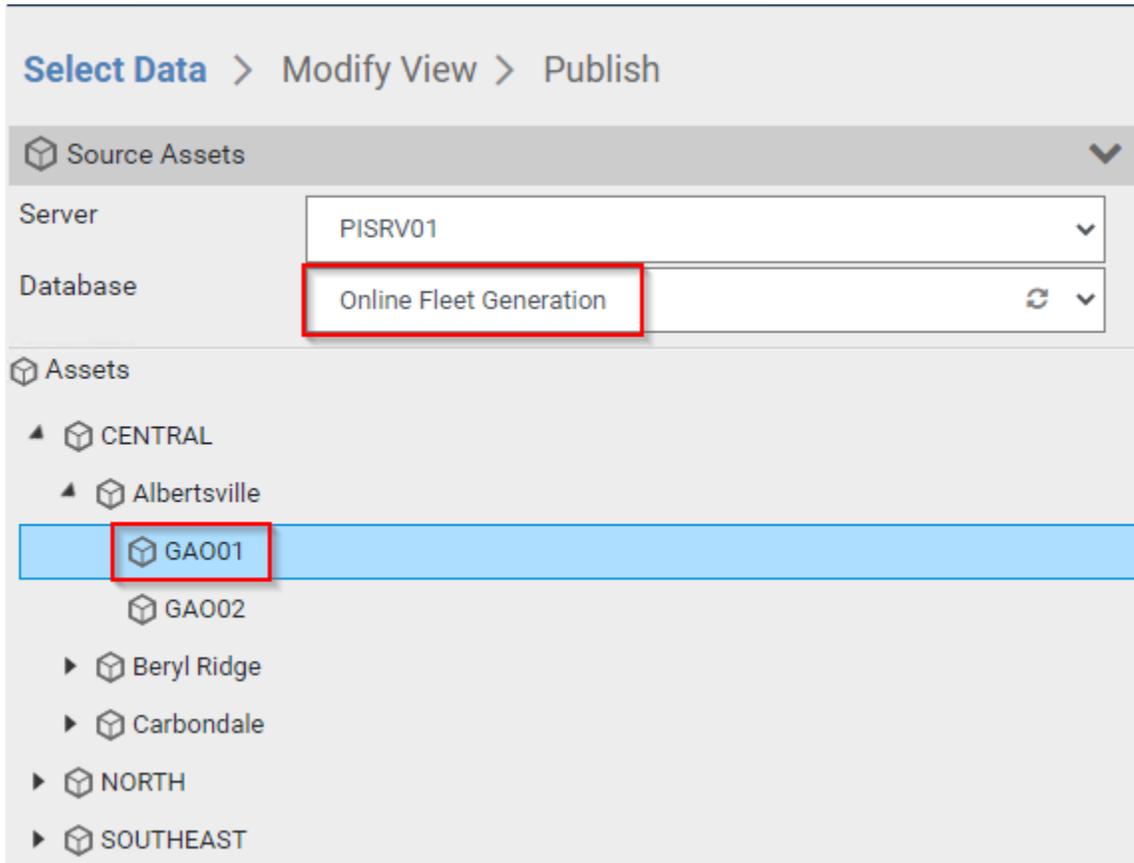
Click **Create Asset View** and name it **Unit Specifications**, click **Create View**:



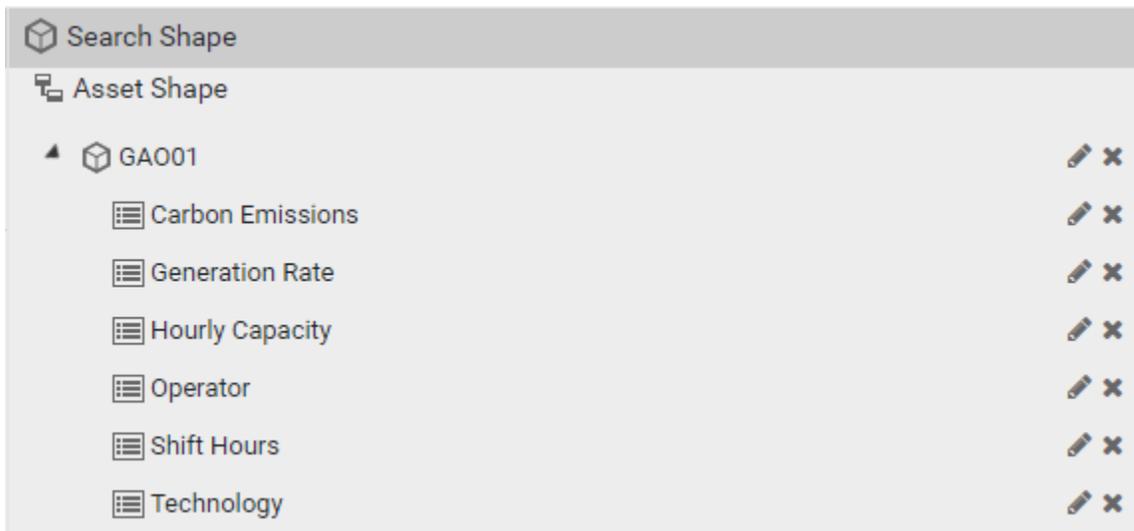
Click **Create a New Shape**



Select Online Fleet Generation as the AF Database, then drill down to GAO01.



The static attributes are: **Carbon Emissions**, **Generation Rate**, **Hourly Capacity**, **Operator**, **Shift Hours**, and **Technology**. Drag them to the shape configuration:



Edit the shape. Uncheck the box next to Asset Name and match on the UNIT template:

**Edit Filters** [X]

Asset Name  
GA001

Asset Template  Search Derived Templates  
UNIT [v]

Add a filter to also include Gas Turbines:

**Edit Filters** [X]

Asset Name  
GA001

Asset Template  Search Derived Templates  
UNIT [v]

Asset Category  
[v]

**+ Add Filter**

Cancel Save

Uncheck the box next to Asset Name and match on the Gas Turbine template then Save:

**Edit Filters** [X]

Filter 1 >

Filter 2 v

Asset Name

GA001

Asset Template  Search Derived Templates

Gas Turbine v

Asset Category

Remove Filter

Add Filter

Cancel Save

Click **Next**.

Change the Start Time and End Time to \* since we only want one row per unit and click Apply. **The preview may not be accurate (missing some units).**

Start Time

End Time

\* \*

Apply

Click **Next**.

Publish to the SQL Server Target (run once):

Select Data > Modify View > **Publish**

**Target Configuration**

SQL Server

**Run Mode**

Run Once

Run on a Schedule

**Summary**

Shape and Matches

- There are 30 Matching Instances
- There are no Time Series attributes selected

Timeframe and Interval

- Your Start Time is \*
- Your End Time is \*
- Your Time Interval gets an interpolated measurement **Every 1 minute**

Publish

When publishing is finished, check the statistics to confirm that 30 rows were published:

Overview Log Security View Configuration **Statistics**

**Run History**

Run Instances	<input checked="" type="radio"/> Duration seconds	<input type="radio"/> Rows Written	<input type="radio"/> Rows Filtered	<input type="radio"/> Error Count
Oct 14, 2020 2:18:10 PM	0.669	30	0	0

## Activity – Publish Unit Performance (dynamic data) Table

### Objective:

- Publish an Asset View containing dynamic attribute data

### Approach:

Create a new **Asset View** name **Unit Performance**.

Create a new shape. Navigate to GAO01 in the Online Fleet Generation database.

Drag **Demand**, **Generating Efficiency**, **Gross Generation**, **Net Generation**, **Shift**, **Total Hourly Gross Generation**, **Unit Status**, and **Utilization** to the shape configuration (exclude those attributes specific to Gas Turbines):



Edit the shape to match UNITS and Gas Turbines (add a second filter) as per the previous exercise:

Filter 1
▼

Asset Name  

GA001

 Asset Template    Search Derived Templates  

UNIT ▼

Filter 1
>

Filter 2
▼

Asset Name  

GA001

 Asset Template    Search Derived Templates  

Gas Turbine ▼

There should be 30 matches. Click **Next**.

Change the Value Mode to sample every 1 hour and Save Changes:

☰ Edit Value Mode  
Interpolated Values  
Every 1 minute

Demand	Generatin
67.468	75.948
67.094	75.903
66.720	75.852
66.346	75.801

Edit Value Mode
✕

Sampled Values  

Sample values every 1 hours ▼

 Interpolate ⓘ  
 Exact ⓘ  
 Use Key Column Demand ▼

Cancel
Save Changes

Change the **Start Time** to **\*-7d** and click **Apply**.

Start Time: \*-7d  
End Time: \*  
Apply

Click **Next**.

Select **SQL Server** as the Target. Run on an **hourly schedule** to keep the data current. Click **Publish**:

Select Data > Modify View > Publish

**Target Configuration**  
SQL Server

**Run Mode**  
 Run Once  
 Run on a Schedule

First Run  
\*

Recur every 1 hours

**Summary**  
Shape and Matches  
• There are 30 Matching Instances  
Timeframe and Interval  
• Your Start Time is \*-7d  
• Your End Time is \*  
• Your Time Interval gets an interpolated measurement Every 1 hour

Publish

When publishing is finished, check the Statistics to confirm that 5070 rows were published:

Overview Log Security View Configuration **Statistics**

**Run History**

Run Instances	Duration seconds	Rows Written	Rows Filtered	Error Count	
Oct 14, 2020 3:05:40 PM	2.469	5070	0	0	
<b>Total</b>	1	2.469	5,070	0	0

## Activity – Import the Unit Specifications and Unit Performance Tables

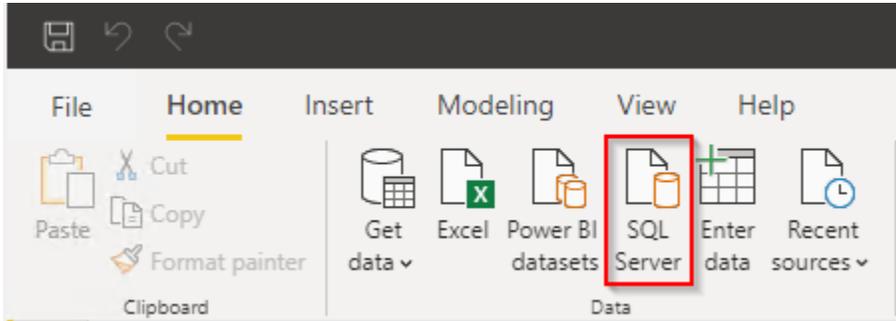
### Objective:

- Import the SQL Server tables created in the previous exercises
- Rename the 'UNITGas Turbine' columns

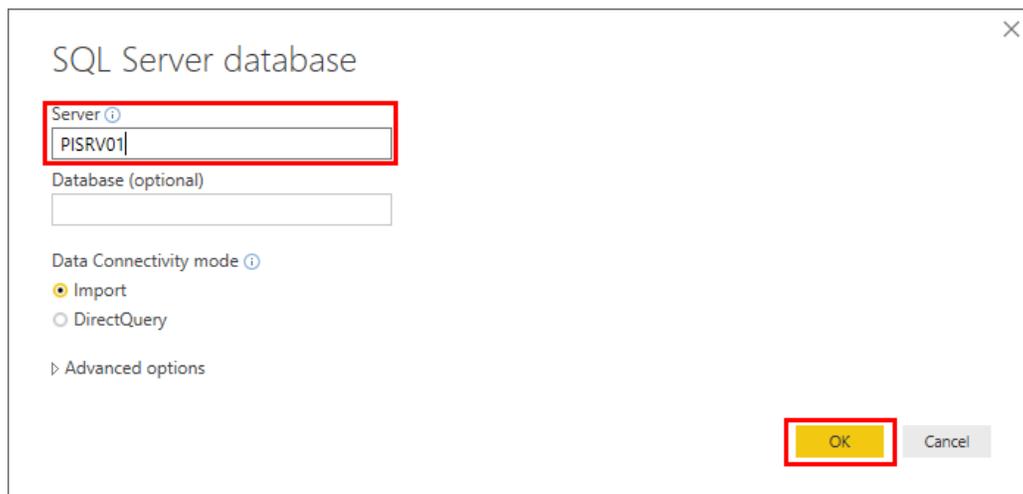
### Approach:

Open a new Power BI report.

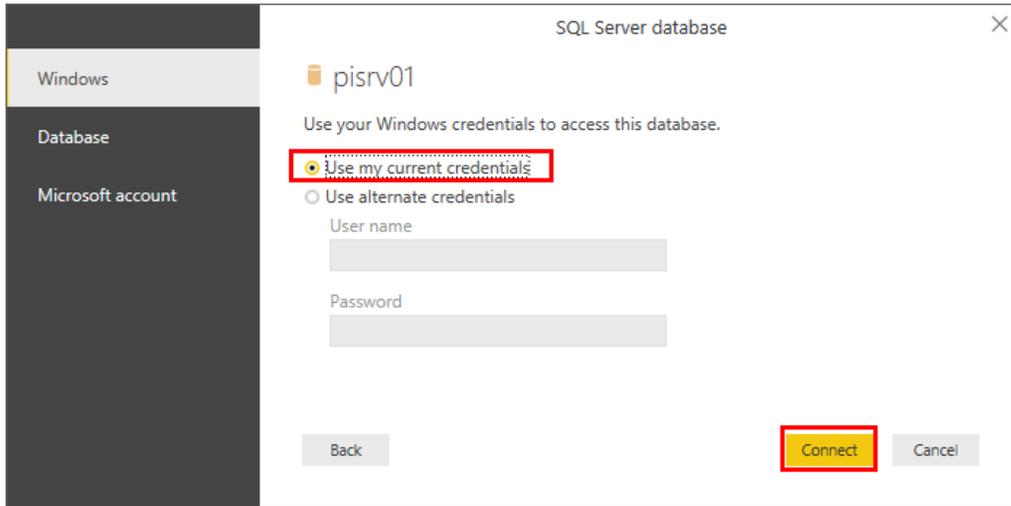
Import data from SQL Server:



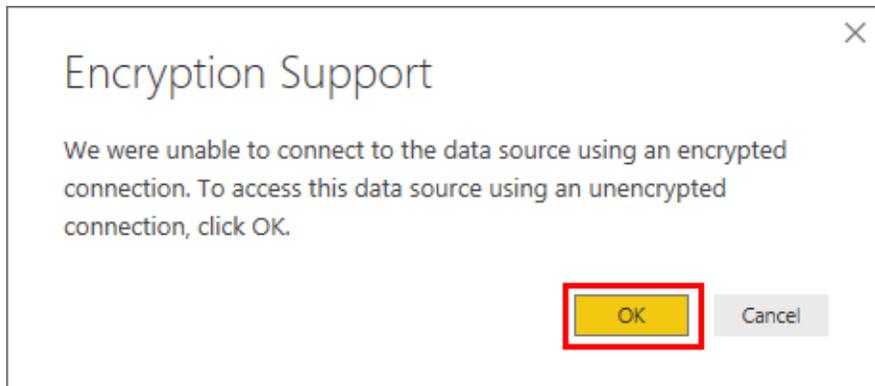
Enter **PISRV01** as the server name and click **OK**.



If Prompted, Leave “use my current credentials” selected and click **Connect**:



There may be a warning that the connection is not encrypted, this can be safely ignored, **click OK if prompted:**



Expand the PIInt database and Select the **Unit Performance** and **Unit Specifications** tables, click **Load**:

The Navigator window shows a tree view of databases. Under 'PIInt [4]', 'Unit Performance' and 'Unit Specifications' are selected and highlighted with a red box. Below the tree, a 'Load' button is highlighted with a red box.

The 'Unit Specifications' table is displayed with the following data:

Id	UNITGas Turbine	TimeStamp	Carbon Emissions	Generat
1	ALX01	10/14/2020 2:18:09 PM	17	
2	BAJ02	10/14/2020 2:18:09 PM	17	
3	CEC01	10/14/2020 2:18:09 PM	17	
4	MAM01	10/14/2020 2:18:09 PM	970	
5	MAM02	10/14/2020 2:18:09 PM	970	
6	MAM03	10/14/2020 2:18:09 PM	970	
7	MAM04	10/14/2020 2:18:09 PM	970	
8	MND01	10/14/2020 2:18:09 PM	405	
9	MND02	10/14/2020 2:18:09 PM	405	
10	PLT01	10/14/2020 2:18:09 PM	17	
11	PLT02	10/14/2020 2:18:09 PM	17	
12	POE01	10/14/2020 2:18:09 PM	17	
13	PQE02	10/14/2020 2:18:09 PM	405	
14	PQE03	10/14/2020 2:18:09 PM	405	
15	PQE04	10/14/2020 2:18:09 PM	405	
16	PTC01	10/14/2020 2:18:09 PM	970	
17	PTC02	10/14/2020 2:18:09 PM	970	
18	PTC03	10/14/2020 2:18:09 PM	970	
19	ZMN01	10/14/2020 2:18:09 PM	970	
20	ZMN02	10/14/2020 2:18:09 PM	970	
21	BCU01	10/14/2020 2:18:09 PM	405	
22	BCU02	10/14/2020 2:18:09 PM	405	
23	GAO01	10/14/2020 2:18:09 PM	405	

Once the tables have loaded, right-click -> **Rename the "UNITGas Turbine" column to UNIT for both tables.**

The screenshot shows the data table with a context menu open over the 'UNITGas Turbine' column header. The 'Rename' option is highlighted with a red box. A red arrow points from the 'Rename' option to the 'Unit Specifications' table in the Fields pane on the right, with the text 'Rename in both tables!' next to it.

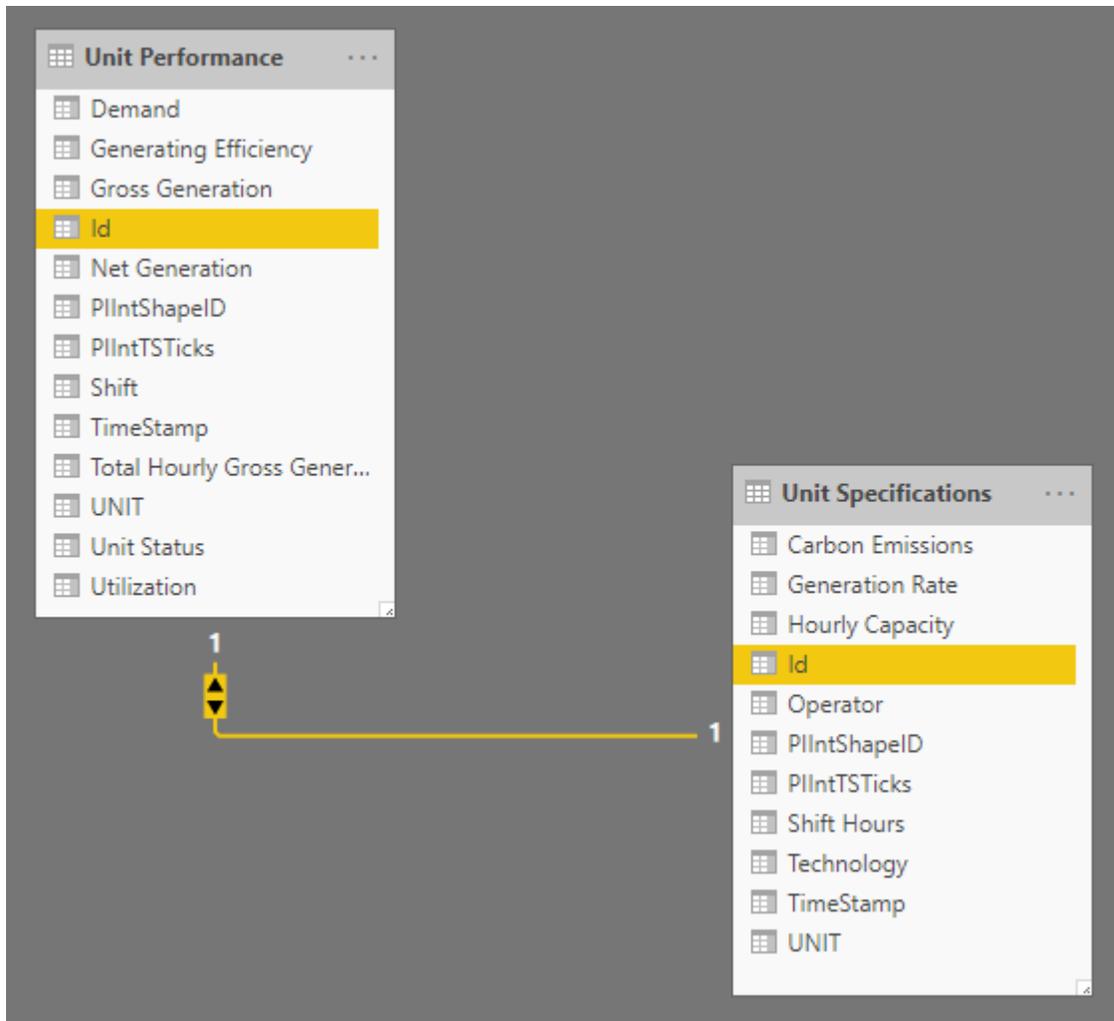
## Activity – Create Table Relationships

### Objective:

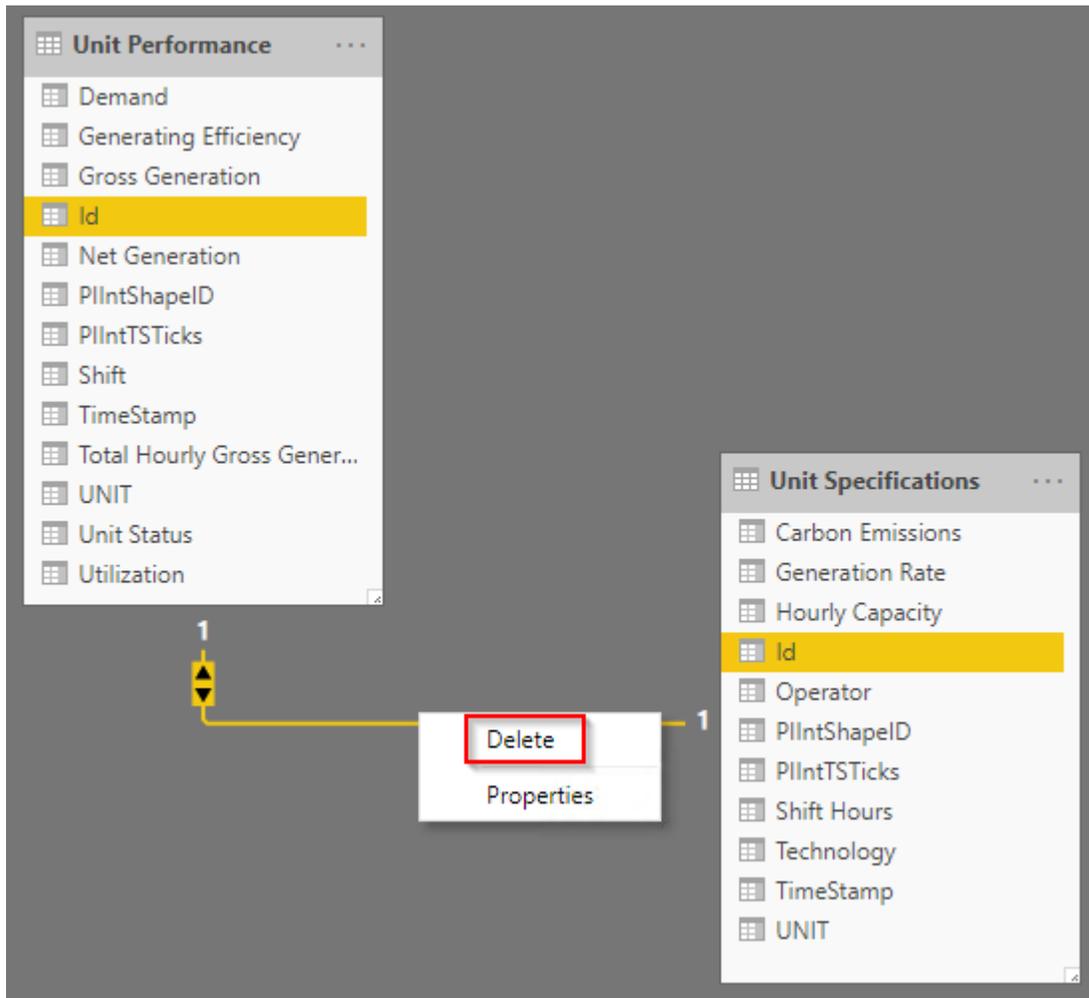
- Delete the automatically created table relationship between Id columns.
- Create a relationship between UNIT columns.

### Approach:

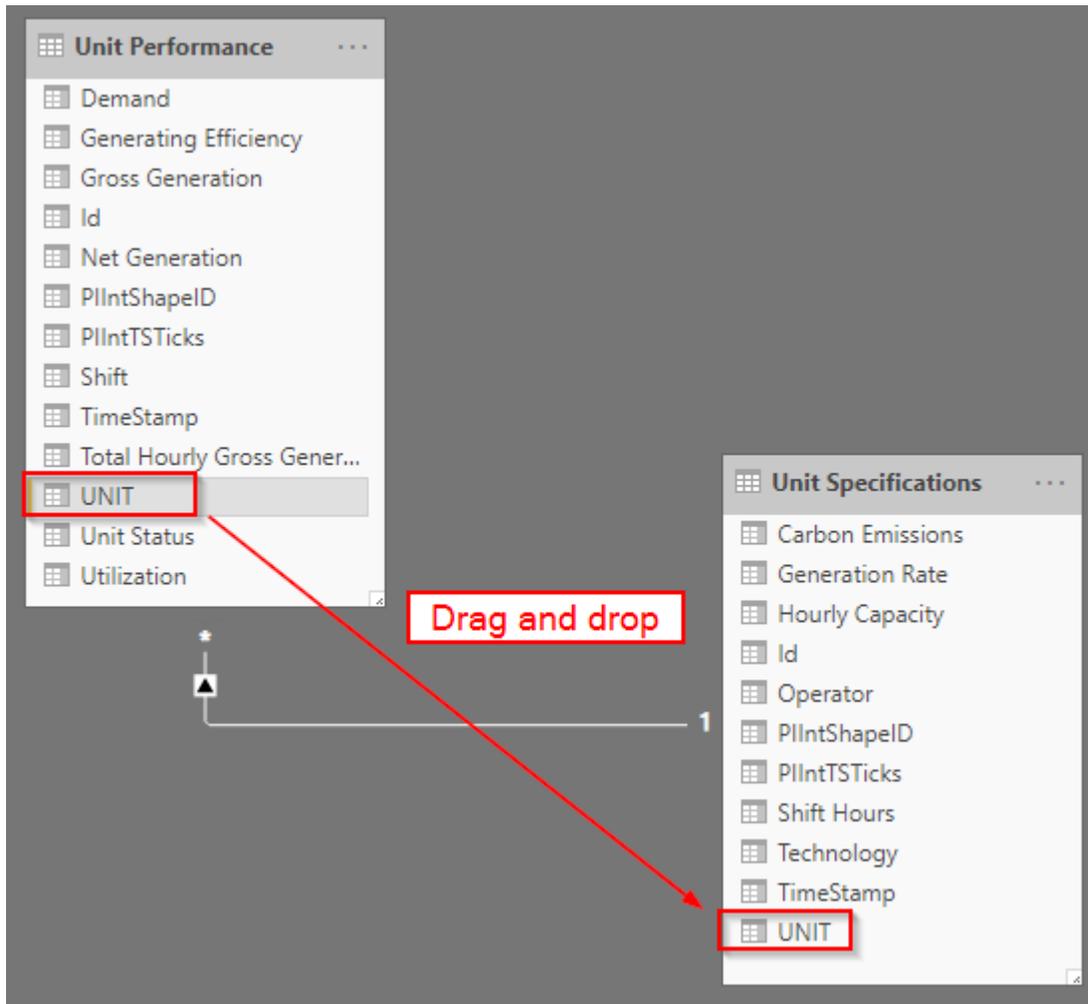
In Power BI, Go to the **Relationships** tab, and click on the line:



This relationship was automatically detected by Power BI and is purely coincidental. The Id is simply the row number. **Right-click -> Delete** the relationship.



Form the correct relationship by **dragging UNIT from one table to UNIT on the other table**. Recall UNITGas Turbine was renamed to UNIT for both tables in a previous step.



Now the Unit Performance table can be filtered based on selections from the Unit Specifications table, for example using a slicer to filter based on Operator.

## Augmenting the Data using DAX

Next, we will add a few calculations to the Unit Performance table that will help assess the total Emissions produced and the total cost of generation. We will also add columns for the day of the week and sort the Weekday in Sunday -> Saturday order.

DAX is a collection of functions, operators, and constants that can be used in a formula, or expression, to calculate and return one or more values. Stated more simply, DAX helps you create new information from data already in your model.

For more information, consult the [Microsoft Documentation](#).

---

**Activity – Calculate the amount of CO2 produced every hour****Objective:**

- Add a DAX formula Calculate the amount of CO2 produced every hour

**Approach:**

In Power BI, navigate to the **Data Tab** and select the **Unit Performance** table.

Right-click any column and add a **new column**. Enter the following formula:

`CO2 = 'Unit Performance'[Total Hourly Gross Generation]*RELATED('Unit Specifications'[Carbon Emissions])`



The screenshot shows the DAX formula bar in Power BI. It contains the formula: `CO2 = 'Unit Performance'[Total Hourly Gross Generation]*RELATED('Unit Specifications'[Carbon Emissions])`. The formula is displayed in a light blue font on a white background. There are 'X' and 'checkmark' icons on the left side of the bar.

Note that Total Hourly Gross Generation has units of MWh, and Carbon Emissions has units of g/kWh. Grams/kWh is the same as Kilograms/MWh, and therefore the result will be in KG.

### Activity – Calculate the Generation Cost

#### Objective:

- Add the cost calculation column to your **Unit Performance** table

#### Approach:

- Add and additional column named **Cost** to the Unit Performance table with the dollar cost per hour.
- Take note of the input units of measure. Cost should be in dollars. Therefore  $\text{Cost} = \text{Total Hourly Gross Generation (from the Unit Performance table)} * \text{Generation Rate (from the Unit Specifications table)} * 1000$

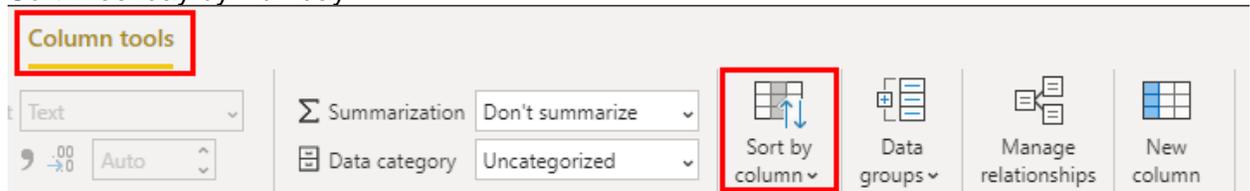
### Activity – Add Column for Day of the Week and sort

#### Objective:

- Add the day of the week to your **Unit Performance** table, also add a column with the numerical day of the week and sort by this value

#### Approach:

- Add and additional column named **Weekday** which shows the day of the week as a string using the `FORMAT()` function
- Add another column named **Numday** which gives the numerical day of the week using the `WEEKDAY()` function
- Sort Weekday by Numday

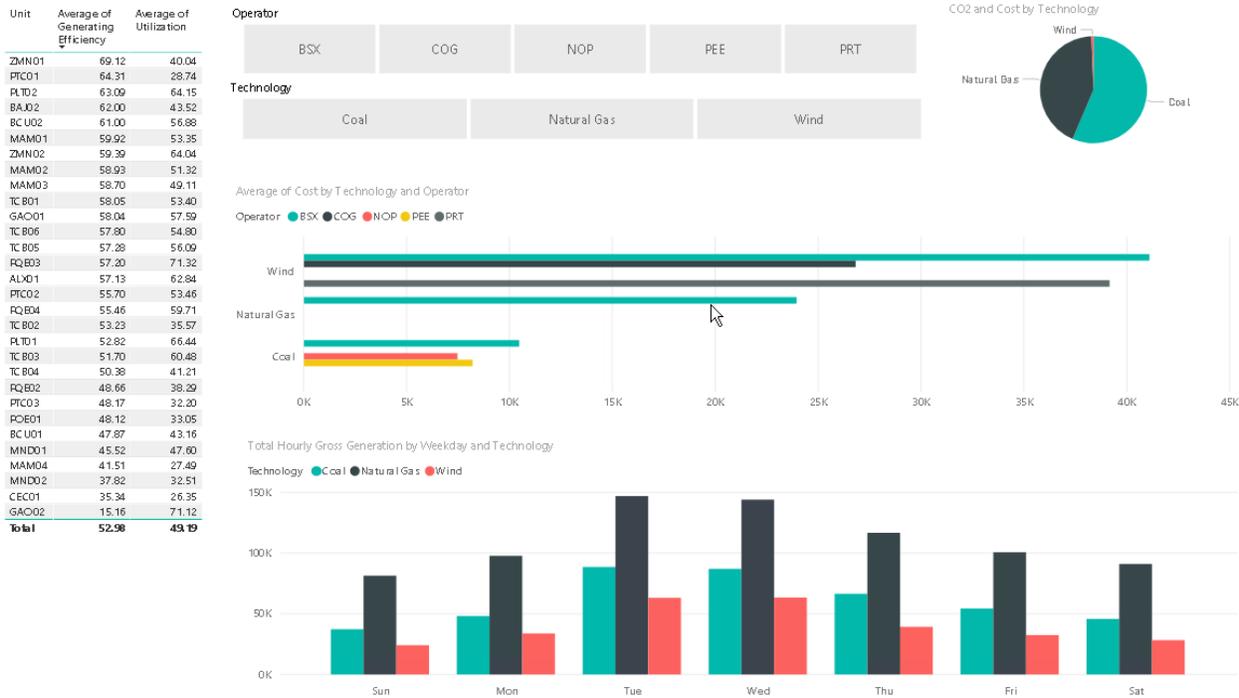


## Configuring the Visualizations

Now we will add visuals to the report to convey useful information about the generating units.

### Activity – Build the Report

**Objective:** Build an interactive Report comparing KPIs for different generation technologies and operators.

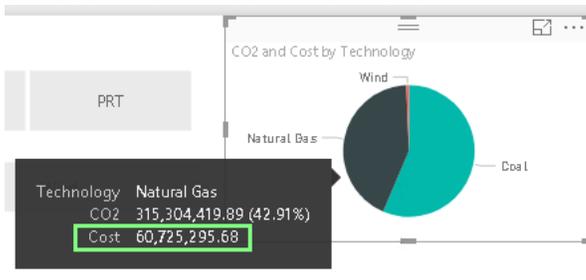


Approach:

- Add a **Table** showing Average Generating Efficiency and Average Utilization by Unit

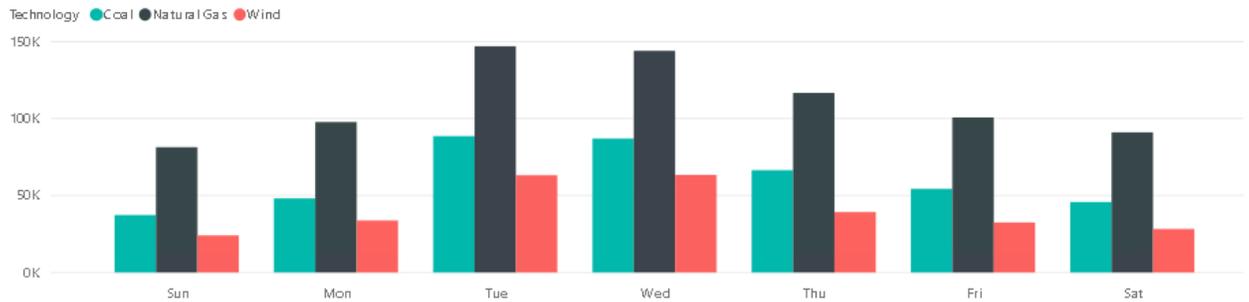
Unit	Average of Generating Efficiency	Average of Utilization
ZMN01	69.12	40.04
PTC01	64.31	28.74
PLTD2	63.09	64.15
BAJ02	62.00	43.52
BC U02	61.00	56.88
MAM01	59.92	53.35
ZMN02	59.39	64.04
MAM02	58.93	51.32
MAM03	58.70	49.11
TC B01	58.05	53.40
GAO01	58.04	57.59
TC B06	57.80	54.80
TC B05	57.28	56.09
RQB03	57.20	71.32
ALXD1	57.13	62.84
PTC02	55.70	53.46
RQB04	55.46	59.71
TC B02	53.23	35.57
PLTD1	52.82	66.44
TC B03	51.70	60.48
TC B04	50.38	41.21
RQB02	48.66	38.29
PTC03	48.17	32.20
POE01	48.12	33.05
BC U01	47.87	43.16
MND01	45.52	47.60
MAM04	41.51	27.49
MND02	37.82	32.51
CEC01	35.34	26.35
GAO02	15.16	71.12
<b>Total</b>	<b>52.98</b>	<b>49.19</b>

- Add a **Pie Chart** showing how the **CO2 emissions** from each generation technology contribute to the whole. Add a **Tooltip** that shows the **Cost** when the user hovers over the Pie Chart



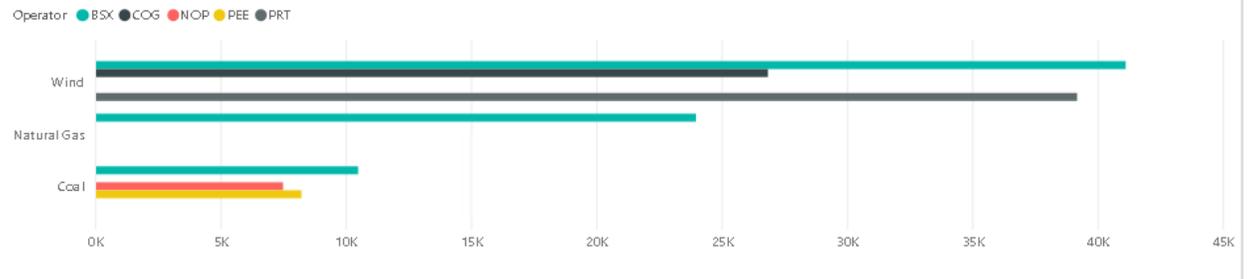
- Add a **Clustered Column Chart** showing the Sum of Total Hourly Gross Generation with Technology as the Legend and Weekday as the Axis

Total Hourly Gross Generation by Weekday and Technology

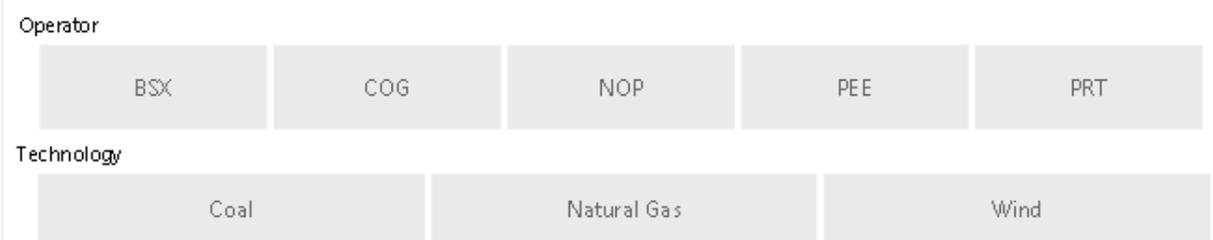


- Add a **Clustered Bar Chart** showing the Average Hourly Cost with Operator as the Legend and Technology as the Axis.

Average of Cost by Technology and Operator



- Add **Slicers** for the Operator and Technology

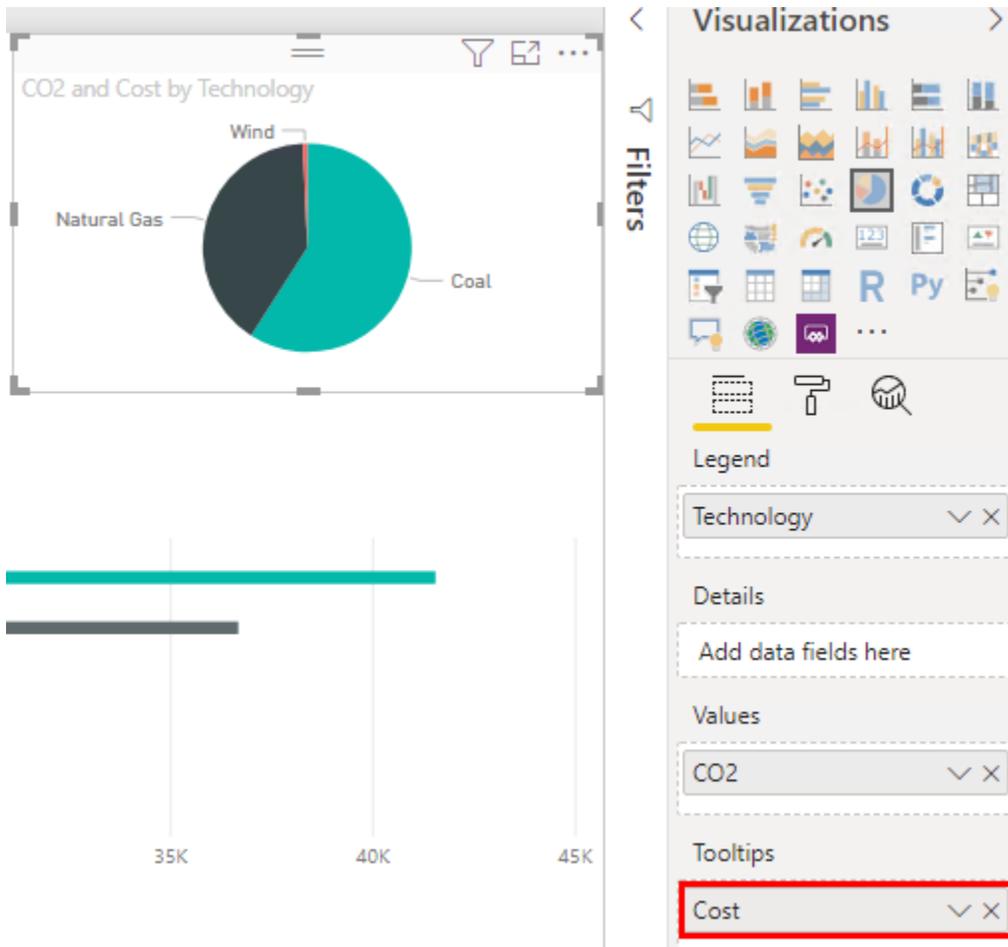


- Optionally improve the look and feel of the report through the use of formatting. Bump up the font sizes, adjust column names and titles, etc.

### Solution – Build the Report

Solution file is in **C:\Class\Part 2 - PI OLEDB Enterprise\Solutions**. There should be enough screenshots in the exercise itself.

The only part that hasn't been explicitly covered is adding the tooltip, which can be accomplished by adding the Cost (summarize as Sum) from the Unit Performance table to the Tooltips field:



## Final Challenge: Fleet Generation Report with Event Frames

The intent of the final challenge is to test your understanding of the course material by going through the entire report development process on your own but first let's get some practice configuring Event Frame views.

Event frames for the last week may need to be backfilled depending on when the virtual machines were deployed and how long they've been running for.

### Activity – Backfill Event Frames in the Online Fleet Generation database

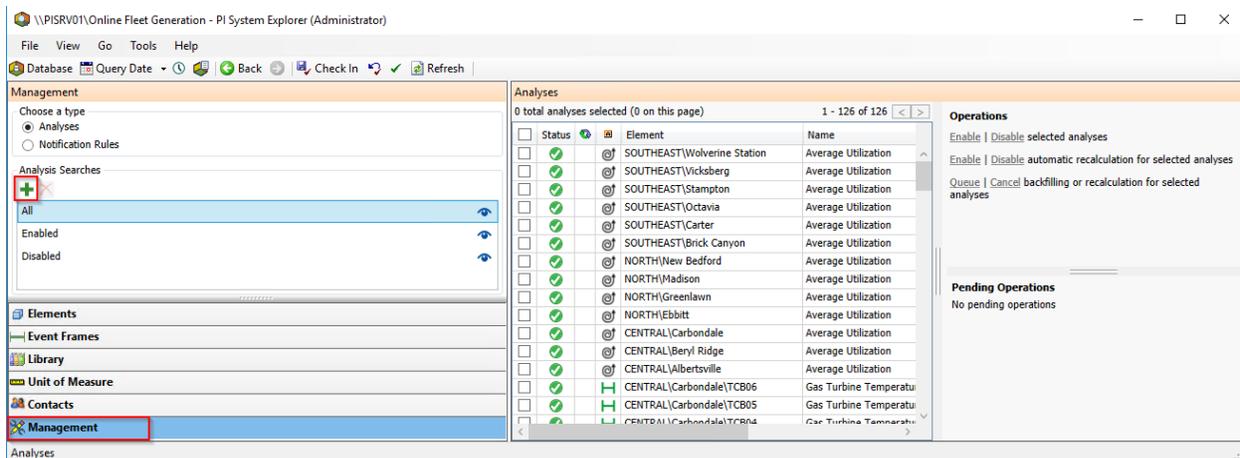
#### Objective:

- Generate the Gas Turbine Temperature Anomaly and Inactivity Event Frames for use in subsequent exercises.

#### Approach:

Open PI System Explorer and go to the **Online Fleet Generation** database.

In the **Management** area, add a new **Search**:



The screenshot shows the PI System Explorer interface for the 'Online Fleet Generation' database. The 'Management' area is active, and a new search has been added to the 'Analysis Searches' list. The search is named 'All' and is currently disabled. The 'Management' tab is highlighted in the left-hand navigation pane.

Status	Element	Name
<input type="checkbox"/>	SOUTHEAST\Wolverine Station	Average Utilization
<input checked="" type="checkbox"/>	SOUTHEAST\Wicksberg	Average Utilization
<input checked="" type="checkbox"/>	SOUTHEAST\Stampton	Average Utilization
<input checked="" type="checkbox"/>	SOUTHEAST\Octavia	Average Utilization
<input checked="" type="checkbox"/>	SOUTHEAST\Carter	Average Utilization
<input checked="" type="checkbox"/>	SOUTHEAST\Brick Canyon	Average Utilization
<input checked="" type="checkbox"/>	NORTH\New Bedford	Average Utilization
<input checked="" type="checkbox"/>	NORTH\Madison	Average Utilization
<input checked="" type="checkbox"/>	NORTH\Greenlawn	Average Utilization
<input checked="" type="checkbox"/>	NORTH\Ebbitt	Average Utilization
<input checked="" type="checkbox"/>	CENTRAL\Carbondale	Average Utilization
<input checked="" type="checkbox"/>	CENTRAL\Beryl Ridge	Average Utilization
<input checked="" type="checkbox"/>	CENTRAL\Albertsville	Average Utilization
<input checked="" type="checkbox"/>	CENTRAL\Carbondale\TCB06	Gas Turbine Temperatu
<input checked="" type="checkbox"/>	CENTRAL\Carbondale\TCB05	Gas Turbine Temperatu
<input checked="" type="checkbox"/>	CENTRAL\Carbondale\TCB04	Gas Turbine Temperatu

Name it **Gas Turbine Temperature Anomaly**, then **Add Criteria**:

Search on **Name: Gas Turbine Temperature Anomaly** and click **OK**:

Select all analyses, click **Queue**, enter **\*-7d** as the **Start** time, check the acknowledgment box, and start the backfill:

✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Status	Element	Name	Template	Backfilling							
✓	CENTRAL\Carbondale\TCB06	Gas Turbine Temperature Anomaly	Gas Turbine Temperature Anomaly	✓							
✓	CENTRAL\Carbondale\TCB05	Gas Turbine Temperature Anomaly	Gas Turbine Temperature Anomaly	✓							
✓	CENTRAL\Carbondale\TCB04	Gas Turbine Temperature Anomaly	Gas Turbine Temperature Anomaly	✓							
✓	CENTRAL\Carbondale\TCB03	Gas Turbine Temperature Anomaly	Gas Turbine Temperature Anomaly	✓							
✓	CENTRAL\Carbondale\TCB02	Gas Turbine Temperature Anomaly	Gas Turbine Temperature Anomaly	✓							
✓	CENTRAL\Carbondale\TCB01	Gas Turbine Temperature Anomaly	Gas Turbine Temperature Anomaly	✓							
✓	CENTRAL\Beryl Ridge\BCU02	Gas Turbine Temperature Anomaly	Gas Turbine Temperature Anomaly	✓							
✓	CENTRAL\Beryl Ridge\BCU01	Gas Turbine Temperature Anomaly	Gas Turbine Temperature Anomaly	✓							
✓	CENTRAL\Albertsville\GAO01	Gas Turbine Temperature Anomaly	Gas Turbine Temperature Anomaly	✓							
✓	CENTRAL\Albertsville\GAO02	Gas Turbine Temperature Anomaly	Gas Turbine Temperature Anomaly	✓							

Repeat the above process for the Inactive Units Event Frame Analyses:

The screenshot shows the PI System Explorer interface with the following components:

- Management Panel:** Search Name: Inactive Units. Name: Inactive Units. Add Criteria button. Note: \* Analyses that match all of these criteria will be displayed.
- Analyses Table:** 30 total analyses selected (30 on this page). Columns: Status, Element, Name, Template, Backfilling. All elements are 'Inactive Units'.
- Operations Panel:** Enable | Disable selected analyses. Enable | Disable automatic recalculation for selected analyses. Queue | Cancel backfilling or recalculation for selected analyses. Start: -7d. End: [blank]. What should we do with existing data?  Leave existing data and fill in gaps.  Permanently delete existing data and recalculate.  I acknowledge that my selection contains event frame analyses. Event frames in the time range will be permanently deleted. This will result in loss of annotations and acknowledgements associated with the event frames. Queue button.

Status	Element	Name	Template	Backfilling
✓	SOUTHEAST\Vicksberg\MAM02	Inactive Units	Inactive Units	✓
✓	SOUTHEAST\Vicksberg\MAM03	Inactive Units	Inactive Units	✓
✓	SOUTHEAST\Vicksberg\MAM04	Inactive Units	Inactive Units	✓
✓	SOUTHEAST\Wolverine Station\ALX01	Inactive Units	Inactive Units	✓
✓	SOUTHEAST\Vicksberg\MAM01	Inactive Units	Inactive Units	✓
✓	SOUTHEAST\Stamton\MND02	Inactive Units	Inactive Units	✓
✓	SOUTHEAST\Stamton\MND01	Inactive Units	Inactive Units	✓
✓	SOUTHEAST\Octavia\ZMN02	Inactive Units	Inactive Units	✓
✓	SOUTHEAST\Octavia\ZMN01	Inactive Units	Inactive Units	✓
✓	SOUTHEAST\Carter\BAJ02	Inactive Units	Inactive Units	✓
✓	SOUTHEAST\Brick Canyon\PLT02	Inactive Units	Inactive Units	✓
✓	SOUTHEAST\Brick Canyon\PLT01	Inactive Units	Inactive Units	✓
✓	NORTH\New Bedford\POE01	Inactive Units	Inactive Units	✓
✓	NORTH\Madison\CEC01	Inactive Units	Inactive Units	✓
✓	NORTH\Greenlawn\PTC03	Inactive Units	Inactive Units	✓
✓	NORTH\Greenlawn\PTC02	Inactive Units	Inactive Units	✓
✓	NORTH\Greenlawn\PTC01	Inactive Units	Inactive Units	✓
✓	NORTH\Ebbitt\POE04	Inactive Units	Inactive Units	✓
✓	NORTH\Ebbitt\POE03	Inactive Units	Inactive Units	✓
✓	NORTH\Ebbitt\POE02	Inactive Units	Inactive Units	✓
✓	CENTRAL\Carbondale\TCB06	Inactive Units	Inactive Units	✓

## Activity – Create an Event View with PI Integrator for BA

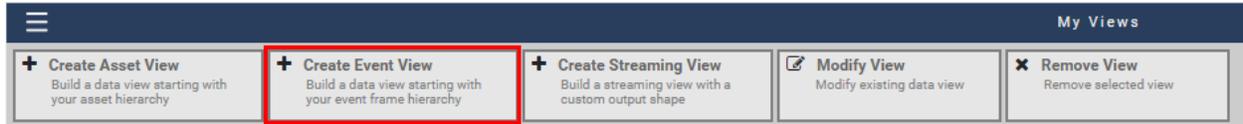
### Objective:

- Create an Event View with PI Integrator for BA

### Approach:

We'll create an event view for Gas Turbine Temperature Anomaly events. Open Chrome and navigate to <https://pisrv01.pischool.int:444/>.

Create an Event View:



Name the event view **Gas Turbine Temperature Anomalies** and click Create View:

The screenshot shows a dialog box titled "Create New Event View". It has a close button (X) in the top right corner. Below the title bar, there is a label "Event View Name" followed by a text input field containing the text "Gas Turbine Temperature Anomalies". At the bottom right of the dialog, there are two buttons: a yellow "Cancel" button and a blue "Create View" button. The "Create View" button is highlighted with a red border.

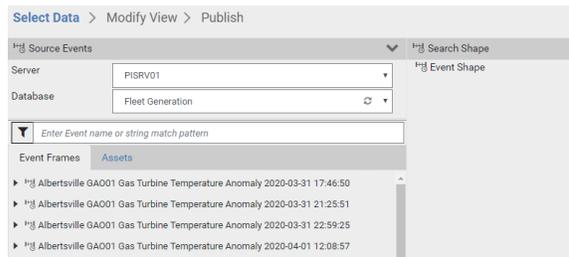
Create a new Shape:

The screenshot shows the "Select Data > Modify View > Publish" interface. It features three main columns: "Source Events", "Search Shape", and "Matches". Below the columns, there is a message: "Import a shape from an existing view or create a new one". Underneath this message, there are two buttons: "Create a New Shape" (highlighted with a red border) and "Import a Shape from Another View".

Point at the Online Fleet Generation database, since that's where the Event Frames are:

The screenshot shows the "Select Data > Modify View > Publish" interface. The "Source Events" section is expanded, showing a "Server" dropdown set to "PISRV01" and a "Database" dropdown set to "Online Fleet Generation". A search bar labeled "Enter Event name" is visible. Below the search bar, there is a list of "Event Frames" with a search icon and the text "As". The list includes several entries, with "Online Fleet Generation" highlighted in blue and a red border. Other entries include "Distribution Network", "Fleet Generation", "Fleet Generation Sim", "NuGreen", "OMF", "PI Big Tires Co.", and "Testing".

Recent Event Frames should show up in the preview. **Everyone's data is random, so everyone's preview will look different:**



You just need to find an event frame of the proper type in order to start building the shape, but let's look at the filtering options which will allow you to narrow down the search.

Click the filter icon:

These settings allow filtering the preview and will help you find the event you're looking for. On a production system there could be over a million Event Frames spanning many different types.

**Filter Events by Time** is pretty straightforward.

**Filter Events by Asset** allows you to filter by primary referenced Element (the Element in AF whose Analysis generated the event). **No need to set anything here, this is just an example.**

Filter Events by Events allows you to filter by Event Frame name or Event Frame Template. **Select Gas Turbine Temperature Anomaly** as the Event Template and click **Apply Filters** to filter out the Inactivity Events:

Event Name  
Enter Event name or string match pattern

Event Template  
Gas Turbine Temperature Anomaly

Clear All Filters Apply Filters

And while we won't set anything, let's take a look at **More Options**. Click the question mark to see explanations for the **Search Mode**. Minimum Duration and Maximum Duration are self-explanatory. **All descendants** applies to hierarchical event frames, which we don't have.

Database

Event Frame Search Options

Includes all objects whose start time is within the specified range

**End Inclusive**  
Includes all objects whose end time is within the specified range

**Inclusive**  
Includes all objects whose start and end time are within the specified range

**Overlapped**  
Includes all objects whose start and end time overlap with the specified range

Search Mode: Overlapped

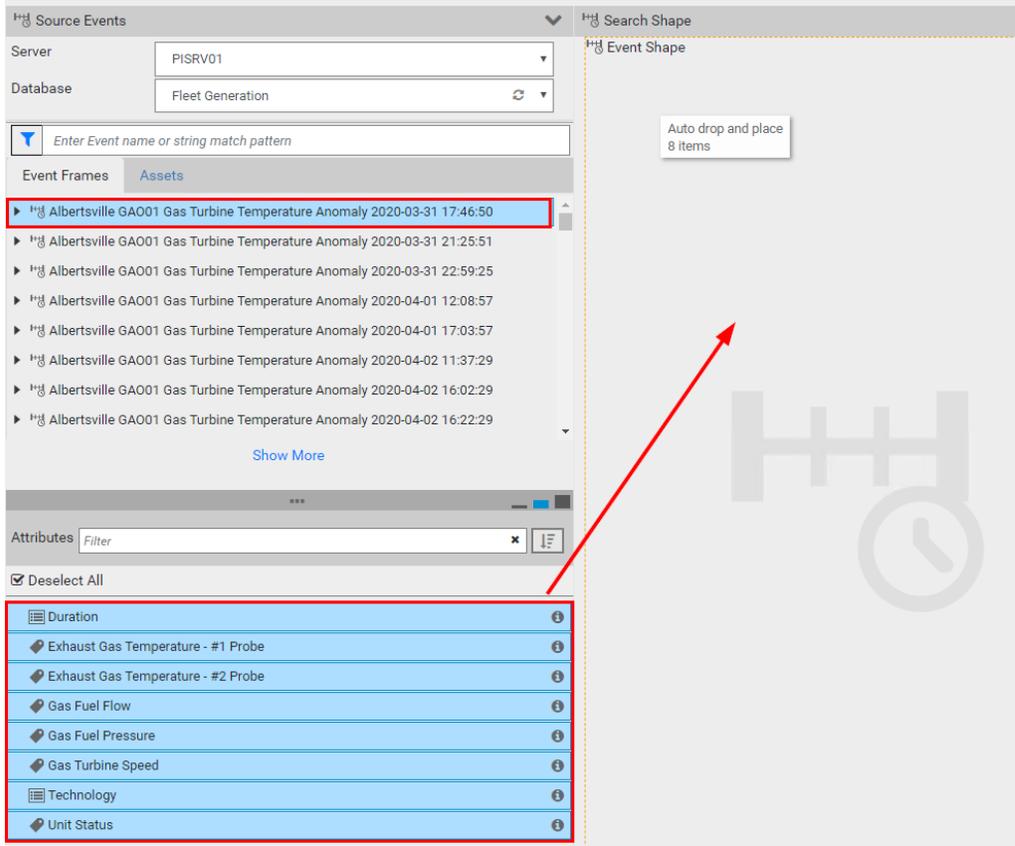
Minimum Duration: none

Maximum Duration: none

All descendants

If you can't find Gas Turbine Temperature Anomaly events, it's possible that they weren't backfilled. Check to see if they exist in the Online Fleet Generation database through PI System Explorer and troubleshoot as necessary.

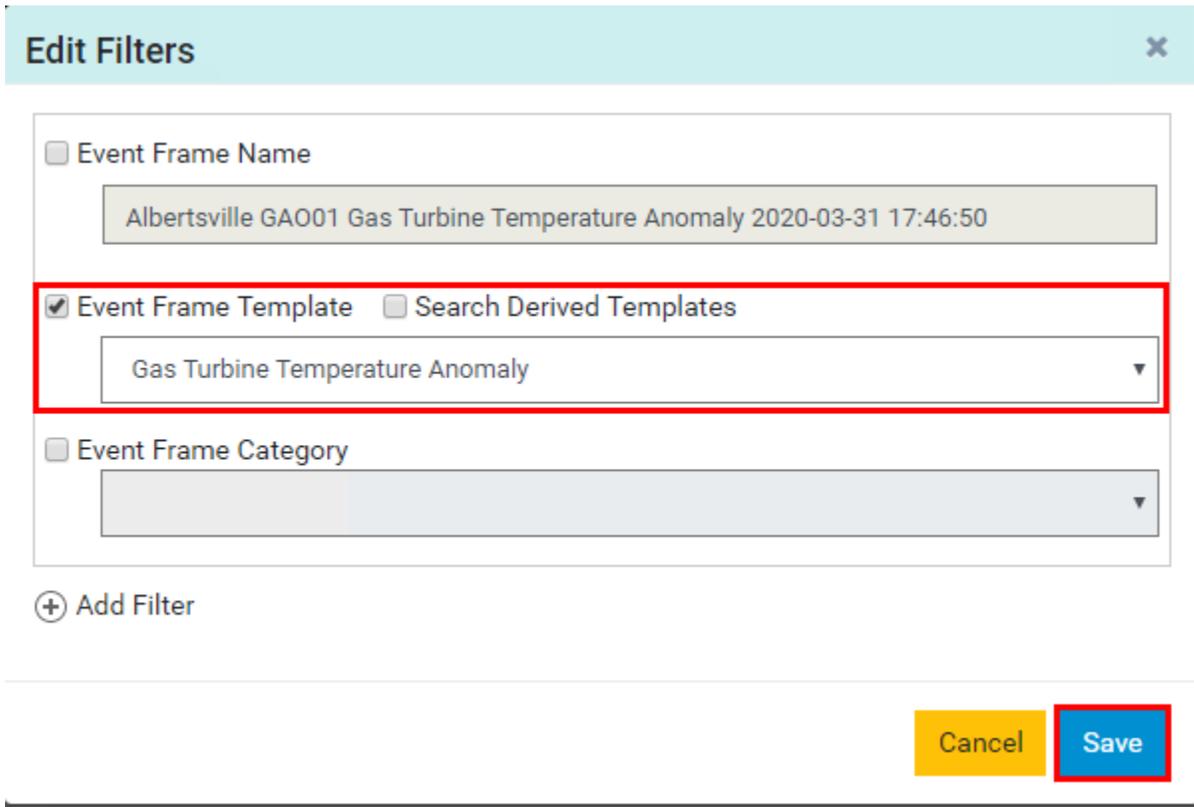
Once you see some events, you can start to configure the Shape. Click one of the Events, then drag and drop all Attributes:



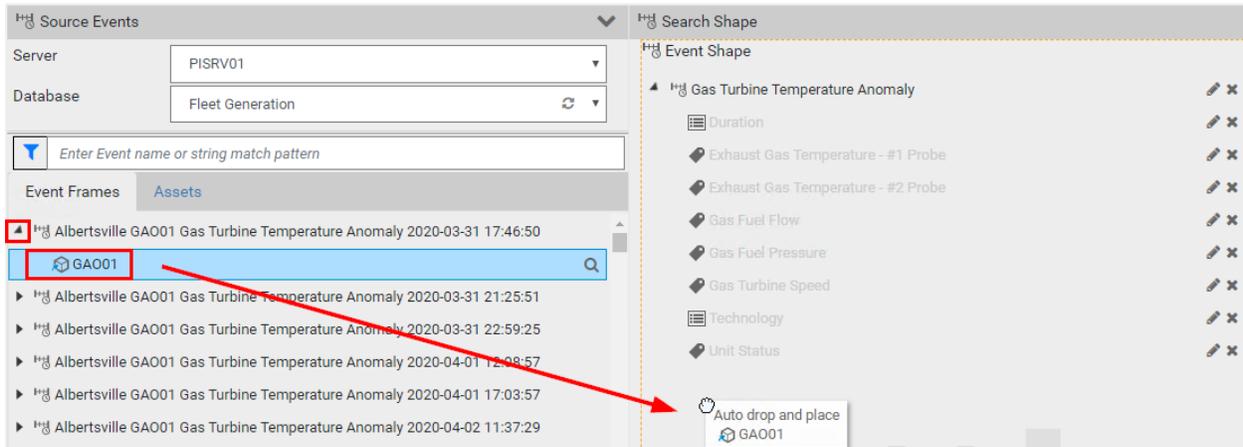
Edit the Shape:



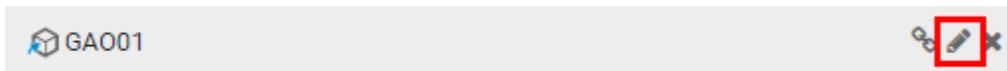
Uncheck the box next to Event Frame Name and match Event Frames by Template then Save.



Expand the Event Frame and drag and drop the Gas Turbine to the shape configuration:



Edit the Gas Turbine object:



Filter by Asset Template, click Save:

**Edit Filters** ✕

Asset Name

Asset Template  
  Search Derived Templates  
  Primary Reference Asset

Asset Category

⊕ Add Filter

Cancel
Save

You should see a bunch of matches. Go to the Next screen:

The screenshot shows the EventViewDesigner interface with the following details:

- Browser:** Advanced Edition, URL: pisrv01.pischool.lint:444/EventViewDesigner
- Page Title:** Gas Turbine Temperature Anomalies
- Navigation:** Select Data > Modify View > Publish
- Source Events:** Server: PISRV01, Database: Fleet Generation
- Search Shape:** Event Shape: Gas Turbine Temperature Anomaly
  - Duration
  - Exhaust Gas Temperature - #1 Probe
  - Exhaust Gas Temperature - #2 Probe
- Matches:** Found 100+ Matches
  - Carbondale TC803 Gas Turbine Temperature Anomaly 2020-03-31 17:46:50
  - Carbondale TC806 Gas Turbine Temperature Anomaly 2020-03-31 17:46:50
  - Carbondale TC804 Gas Turbine Temperature Anomaly 2020-03-31 17:46:50
  - Carbondale TC802 Gas Turbine Temperature Anomaly 2020-03-31 17:46:50
- Buttons:** A red 'Next' button is visible in the top right corner of the interface.

Change the Event Frame Duration Data Content to Second otherwise it will be displayed as a round number of hours.

Be sure to click Apply Changes.

The screenshot shows a data table with columns: Start Time, End Time, Event Frame Duration, Time Stamp, Duration, 1st Gas Temperature, 2nd Gas Temperature, Fuel Flow, Gas Fuel Pressure, Gas Turbine Speed, and Technology. The 'Event Frame Duration' column is highlighted in blue. The 'Column Details' panel on the right shows the following settings:

- Name: Event Frame Duration
- Data Content: **Second** (highlighted in red)
- Time Context: Event Frame Duration
- Data Type: Integer
- Buttons: Remove Column, **Apply Changes** (highlighted in red)

Change the Start Time to \*-7d and end time to \*, click Apply, then move to the Next Screen:

The screenshot shows the configuration interface for the Event Frame Duration column. The 'Start Time' field is set to '\*-7d' and the 'End Time' field is set to '\*'. The 'Apply' button is highlighted in red. There are also 'Back' and 'Next' buttons visible at the top right.

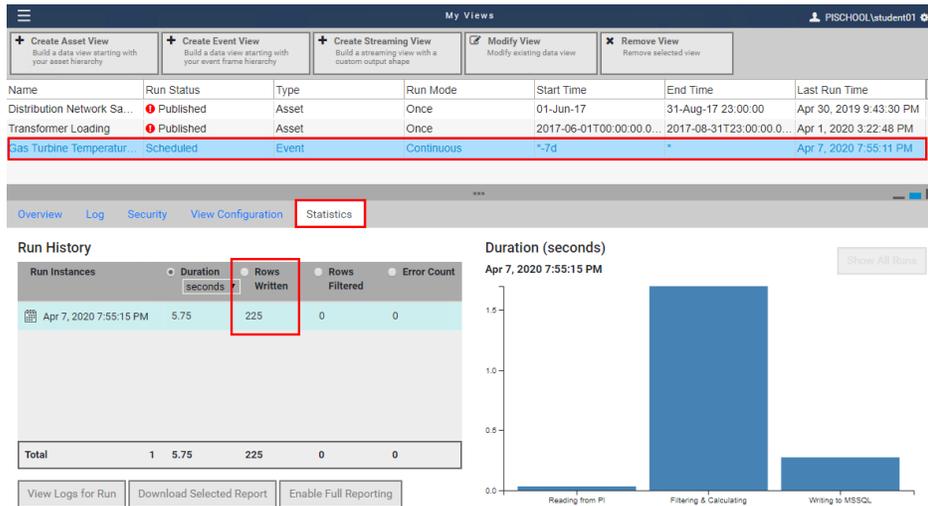
Select SQL Server as the target and have it run on an hourly schedule to keep the Event Frames current, then click Publish.

The screenshot shows the 'Publish' configuration screen. The 'Target Configuration' is set to 'SQL Server'. The 'Run Mode' is set to 'Run on a Schedule'. The 'First Run' is set to '\*'. The 'Recur every' is set to '1' hours. The 'Publish' button is highlighted in red.

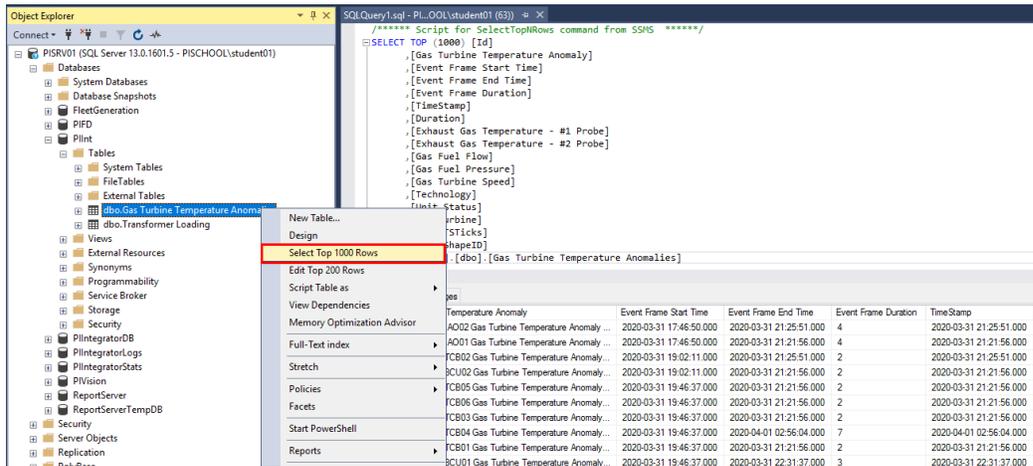
**Summary**

- Shape and Matches
  - There are 100+ Matching Instances
- Timeframe and Interval
  - Your Start Time is \*-7d
  - Your End Time is \*
  - Your Time Interval gets an interpolated measurement Every 1 minute

Check the statistics to confirm that Rows were written:



Optionally use **SQL Server Management Studio** to confirm that Event Frames were written to the SQL Server table:



## Activity – Build the Final Report

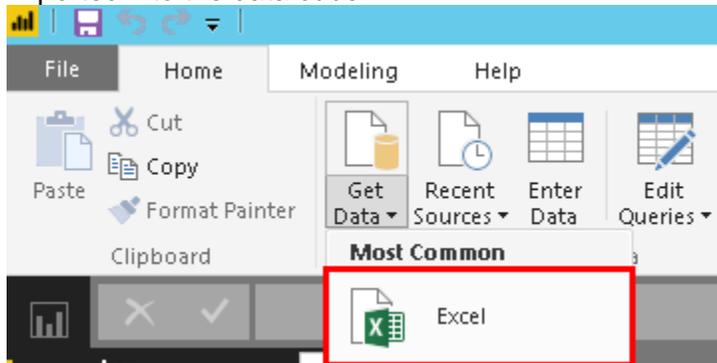
This is your chance to build a report from scratch using the concepts learned earlier in the course. Let's see if you can remember how to do everything!

### Objectives:

- Incorporate downtime (Inactivity) events
- Determine the carbon footprint of each unit and display on a US map.

### Approach:

- Create a new Sheet in the Fleet Generation Report (the imported tables will be re-used)
- Geospatial information for all units in Fleet Generation is located in **C:\Class\Final Exercise\Unit Coordinates.xlsx**. This data will need to be imported into the data cube.



- Use PI Integrator for BA to publish an Event View for Inactivity events
  - You need a column to form the relationship between the Unit Specifications table and the Inactivity Event Frames, it's probably easiest to join on Unit Name (GAO01, etc).
  - Configure a continuous Event View that updates hourly. The initial publication should include the last 7 days
- Import the Inactivity events for the last 7 days using the above Event View
- Create the table relationships (should happen automatically if all columns are named Units).
  - Between the Unit Specifications table and the longitude/latitude table
  - Between the Unit Specifications table and the Inactivity query results
- Insert a table showing the number of downtime events (Inactivity Event Frames) and average duration of event frames for each unit. Add the Average Utilization to the same table.
- Insert a map within the client to display the location of each of the units and the associated total hourly carbon emissions.
- Configure the report in such a way that the Table relationships are tested. Use data from multiple tables in the same Visual.

- Customize the display to make it more user friendly by improving the formatting and adding slicers.

**Hints:**

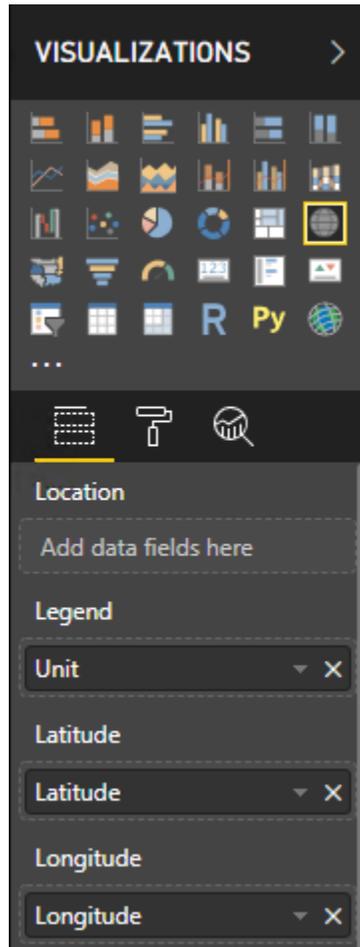
- When configuring the Event View, the Data Context must be set to Second or else it will round to the nearest whole hour (which will always be zero).

The screenshot shows the 'InactivityTest' interface with a data table and a 'Column Details' configuration panel. The table has two columns: 'Event Frame Duration' and 'Demand'. The 'Event Frame Duration' column is highlighted with a red border. The 'Column Details' panel is open, showing the following settings:

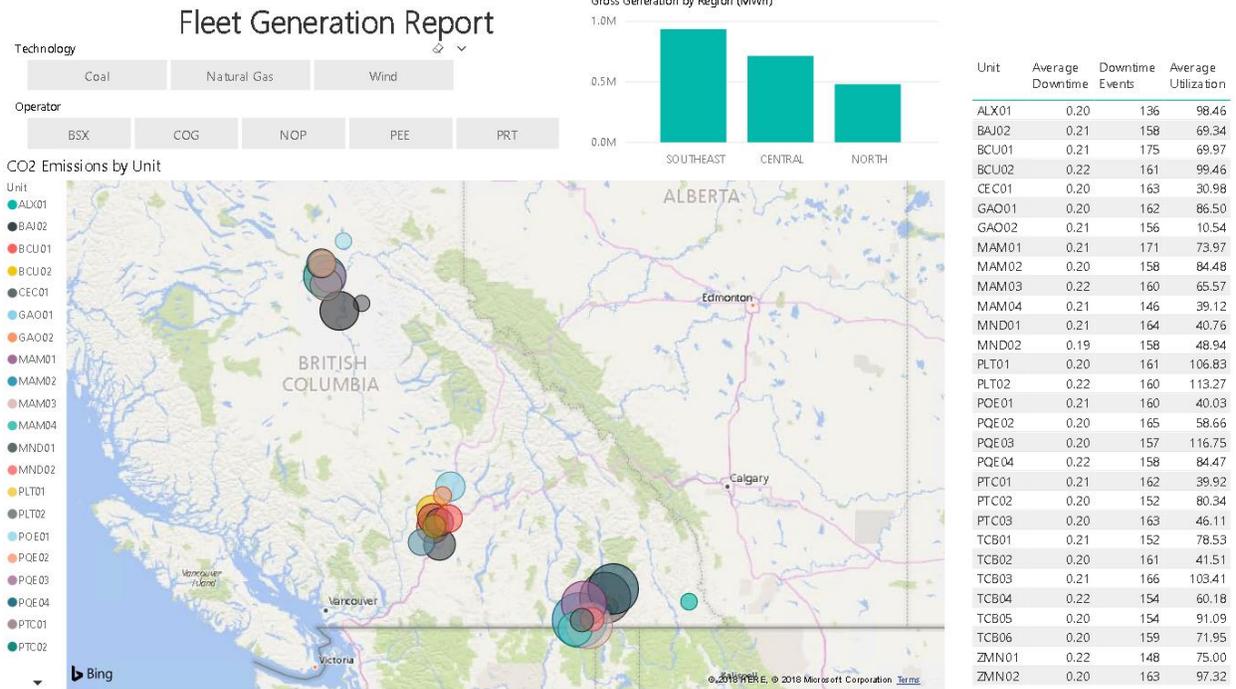
- Name:** Event Frame Duration
- Data Content:** Second
- Time Context:** Event Frame Duration
- Data Type:** Integer
- Buttons:** Remove Column, Apply Changes

Event Frame Duration	Demand
800	142.12
800	222.234
800	175.188
800	226.886
800	231.862
3000	265.04
300	183.634
300	193.162
300	182.241
300	254.449
800	208.43

- Use the ordinary map, not the ESRI one. Drag and drop latitude and longitude from the table that was imported from the Unit Coordinates.xlsx spreadsheet.



A sample of what the report could look like:



The above report can be found in **C:\Class\Final Exercise\Solution**