

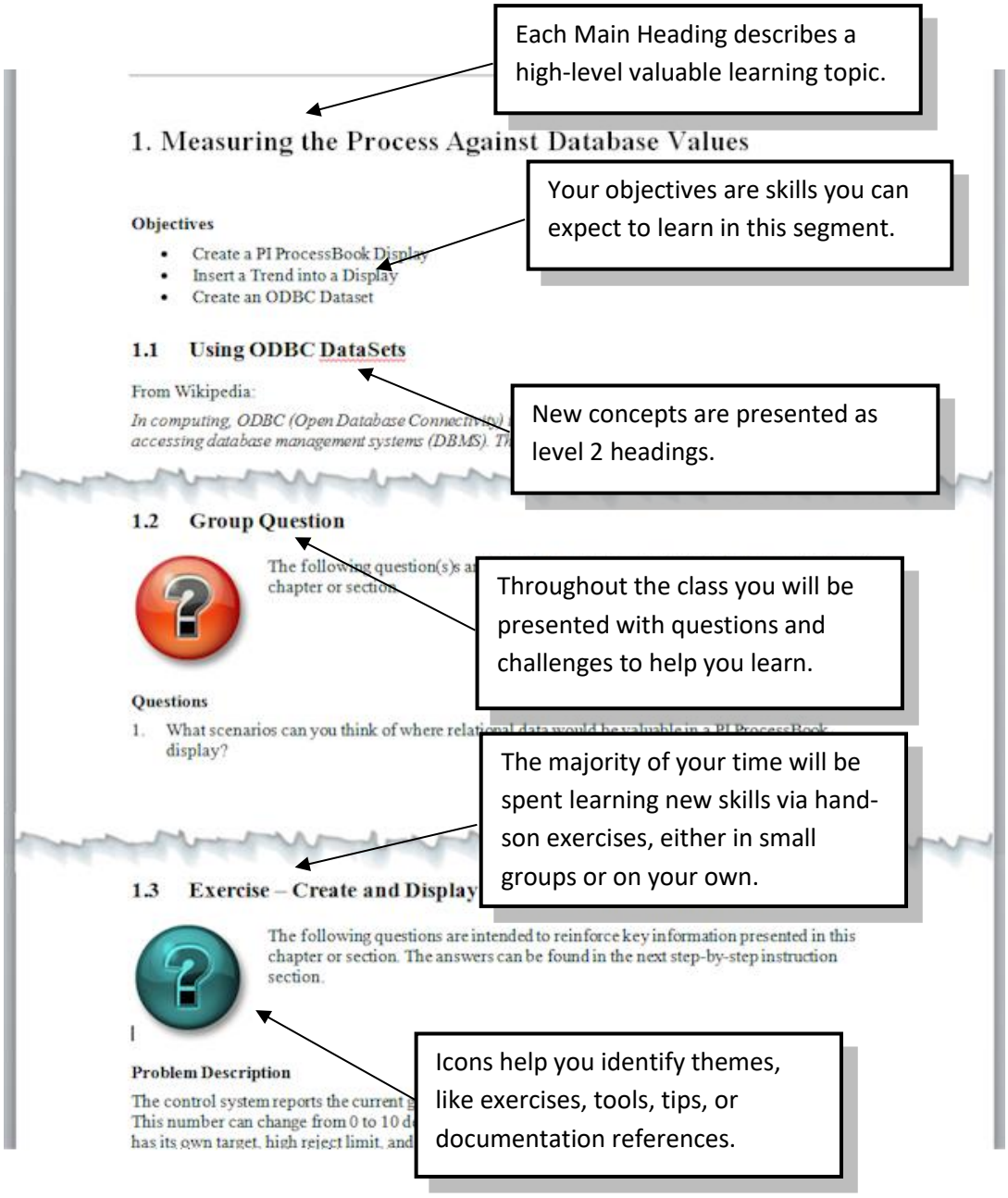
PI System Administration

Version 2024

© 2025 AVEVA Group Limited and its subsidiaries. All rights reserved.
AVEVA, the AVEVA logos and AVEVA product names are trademarks or registered trademarks of AVEVA Group Limited or its subsidiaries in the United Kingdom and other countries. Other brands and products names are the trademarks of their respective companies.

AVEVA Group Limited
High Cross, Madingley Road
Cambridge CB3 0HB, UK
Tel +44 (0)1223 556655
Fax +44 (0)1223 556666

How to Use this Workbook



Software Versions Used in this Document:

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

Software	Version
PI Data Archive	2024
PI AF Server	2024
PI OPC interface	2.7.1.41
PI Interface for RDBMS	3.24.6.36
PI Interface for UFL	3.6.7.144
PI API	2018 Patch 2
PI System Explorer	2024
PI Vision	2024

Table of Contents

1. **PI System Basics** 5
   - 1.1 What is a PI System? 5
   - 1.2 A typical PI System architecture 6
   - 1.3 Understanding PI Points 8
   - 1.4 Directed Activity – Search for PI Points using SMT 9
   - 1.5 Using the Tag Search 10
   - 1.6 Solo Exercise – Use the Tag Search 11
   - 1.7 Directed Activity – View PI Point data using PI Vision 12
   - 1.8 Writing Time in the PI System 14

2. **PI Interface Management** 20
   - 2.1 PI Connectors and PI Adapters 20
   - 2.2 Understanding important IT Concepts 21
   - 2.3 Define the role of a PI Interface 23
   - 2.5 Solo Exercise – Choose a PI Interface 24
   - 2.6 Common PI Interfaces 26
   - 2.7 Define the components of a PI Interface 26
   - 2.8 Define the PI Interface Configuration Utility 27
   - 2.9 PI Interface installation methodology 28
   - 2.10 Group Questions – PI Interface architecture 29
   - 2.11 Install and configure a PI Interface for OPC DA 31
   - 2.12 Digital tags 51
   - 2.13 Troubleshooting Exercise 53
   - 2.14 Configuring a reliable PI Interface 54
   - 2.15 Group Questions – Preventing data loss 54
   - 2.16 Defining the PI Buffer Subsystem 55
   - 2.17 Monitor the health of the PI Interface 66
   - 2.18 Troubleshooting Exercise 67

3. **Data Archive Management** 68
   - 3.1 Define the role of the Data Archive 68
   - 3.2 Describe the Data Archive subsystems 69
   - 3.3 Data Flow through the Data Archive 71
   - 3.4 Understanding Exception and Compression 80
   - 3.5 Data Archive Files 87
   - 3.6 Managing Archive Files 89
   - 3.7 Manage Tuning Parameters 96
   - 3.8 Connections to PI Data Archive 98
   - 3.9 Manage Data Archive Backups 99
   - 3.10 Troubleshooting Exercise 105

4. Asset Framework Management 106
4.1 Define the role of the Asset Framework 106
4.2 Define assets and attributes 110
4.3 PI System Explorer 112
4.4 Directed Activity – Organizing your PI Points into AF Assets 116
4.5 Solo Exercise – Building Assets from Templates using PI Builder 119
4.6 Directed Activity – Take advantage of your asset model in PI Vision 122
4.7 Components of PI Asset Framework 124
4.8 Data Flow when using the Asset Framework 125
4.9 AF architecture 127
4.10 Manage Asset Framework Backups 127
4.11 Asset Analytics Performance 131
4.12 Troubleshooting Exercise 135
5. PI Vision Management 136
5.1 Administration 136
5.2 Backups 137
6. PI System Security Management 138
6.1 Securing a PI System 138
6.2 Describe the ports used for PI System communication 140
6.3 Authentication vs. Authorization 144
6.4 Asset Framework Security 145
6.5 Data Archive Security 151
7. Monitoring a PI System 170
7.1 Monitoring Tools 170
7.2 Group Questions – What should I monitor? 171
7.3 Stale and Bad Points 173
8. Troubleshooting a PI System 174
8.1 Message Logs 174
8.2 Where to look for answers 177
9. Final Exercise 178
9.1 Final Exercise – Import performance data from servers 179
9.2 Final Exercise – Import data from text files 182
9.3 Final Exercise – Import data from SQL Server 185
10. Appendix 189
10.1 Final Exercise – Import performance data from servers 189
10.2 Final Exercise – Import data from text files 201
10.3 Final Exercise – Import data from SQL Server 215

1. PI System Basics

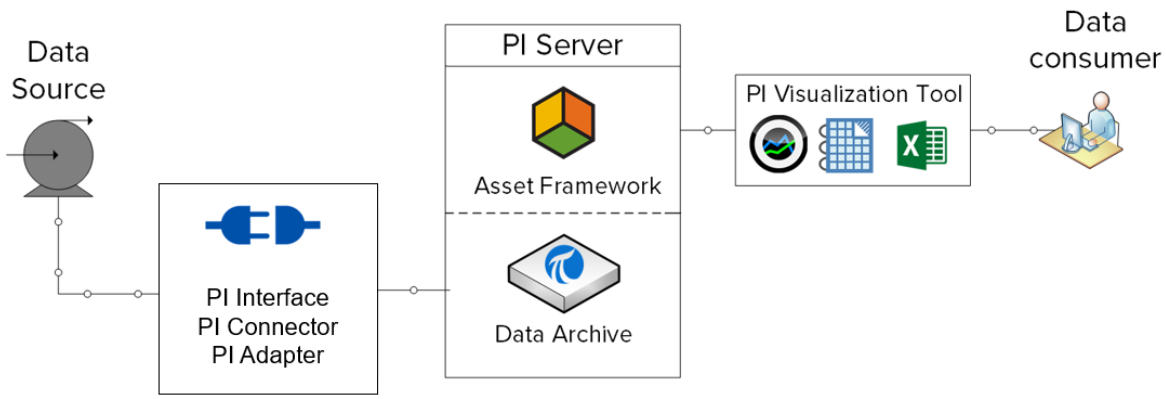
Objectives

- Describe the components of the PI System
- Describe a PI Point
- Find and view PI Point data using System Management Tools
- Find and view PI Point data using PI Vision
- Explain absolute and relative time in the PI System
- Translate and create PI Time expressions
- Explain how the Data Archive handles Times Zones and DST, and future data

1.1 What is a PI System?

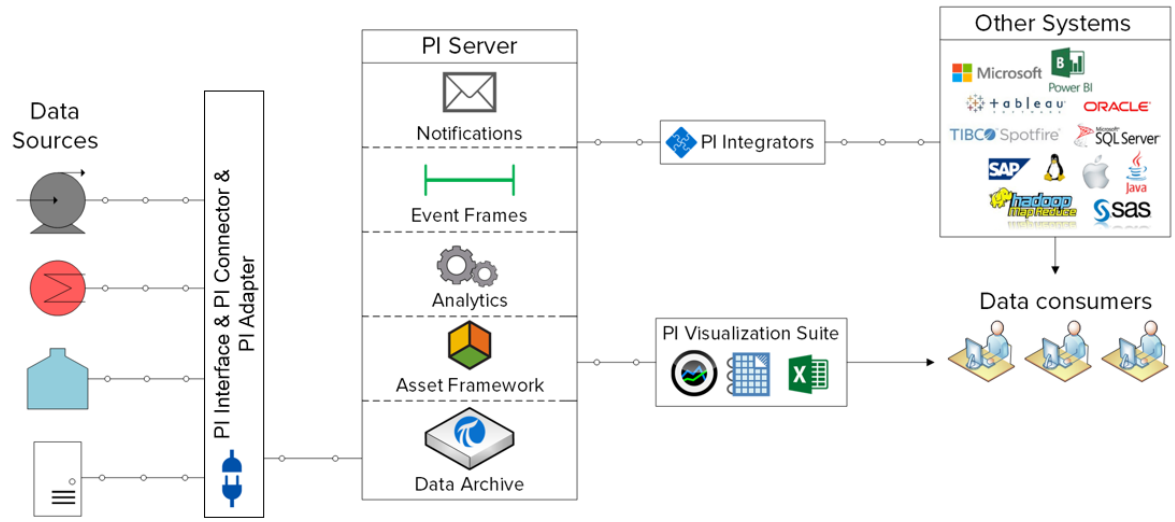
AVEVA is a software company, which means we don't sell any hardware. The only thing we sell is the PI System. It is important to understand that the PI System is just software, and it resides on servers and computers on a network.

The PI System is a software suite that collects, stores, and enhances data from your plant or process, and delivers it to users who need it. Simply put, the PI System is everything between the data source and the data consumer. The simplest possible PI System is made up of the following software components:



- **PI Interface, PI Connector or PI Adapter**: Collects data from a data source
- **PI Server**
  - **Data Archive**: Stores the data
  - **Asset Framework**: Organizes and enhances the data
- **PI Visualization Tool**: Displays the data to the consumer

A more complete PI System would look like this:

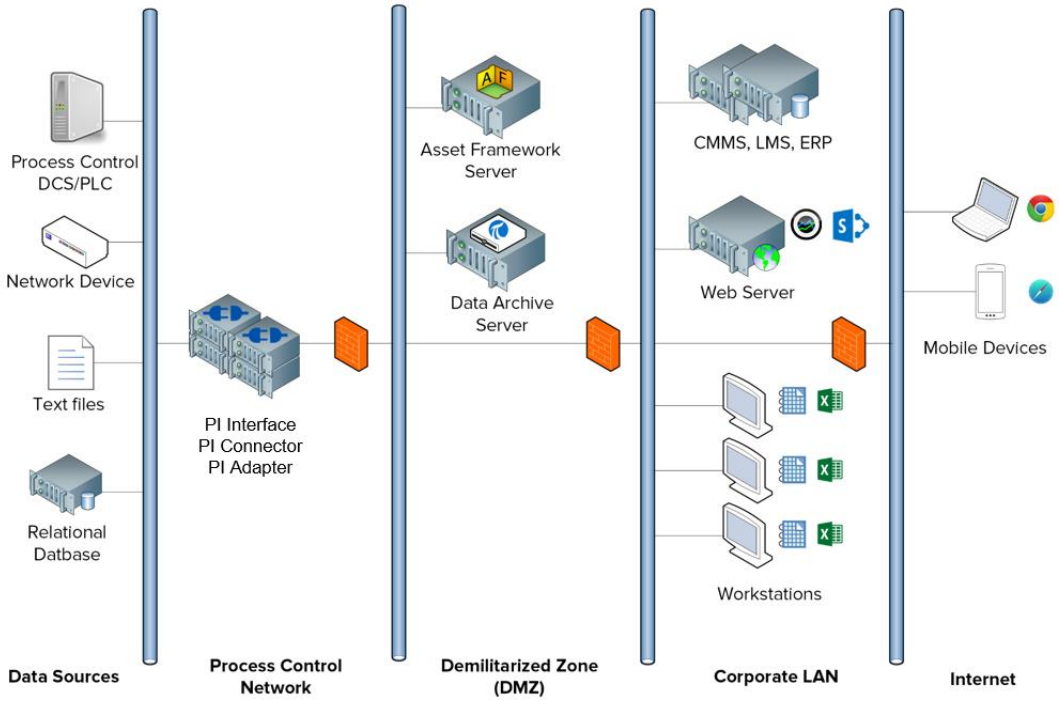


1.2 A typical PI System architecture

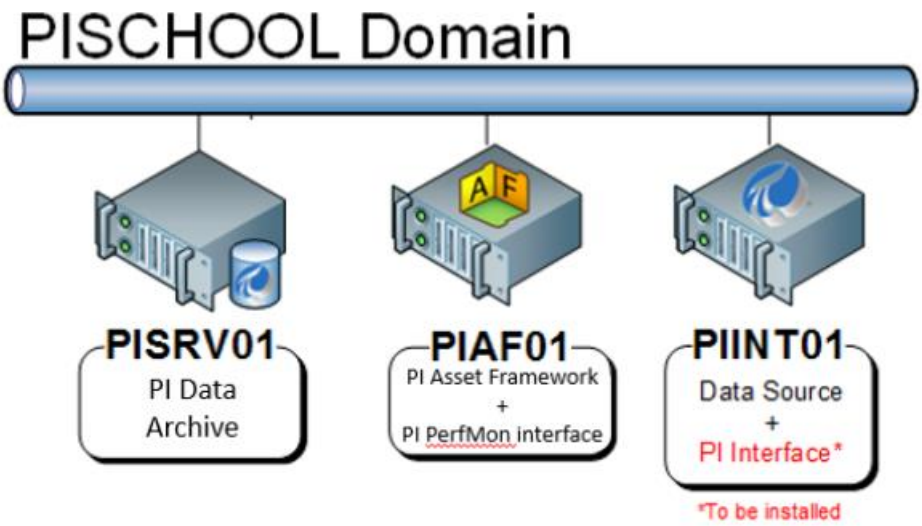
So far, we have discussed the PI System in terms of its software components. These components must be installed on computers and servers which reside on the same computer network as the data sources. The layout of the PI System components with respect to the computer network is called the “PI System architecture”.

PI System architectures can range from very simple to very complex. Theoretically, all the PI System components could be installed on the same computer. In practice, this is rarely the case. Multiple factors will come into play when choosing a PI System architecture, including security, performance, and scalability.

Below is a typical PI System architecture:



In this course, we will be working in a Virtual Learning Environment. Below is a diagram of our PI System architecture:



1.3 Understanding PI Points

Any data that has a changing value over time can be collected and stored in the Data Archive.

In a process, this could be:

- The temperature in a tank
- The volumetric flow through a pump
- The speed of a propeller

These changing values all represent *streams* of data.

PI Points, also referred to as PI Tags, are what define the data streams stored in the Data Archive. Whenever a PI System administrator wants to collect a new stream of data, he or she must create a PI Point.

1.3.1 Defining key PI Point Attributes

PI Point *attributes* are what define the PI Point. They have multiple different functions, including:

- Specifying how to collect data at the data source
- Defining which PI Interface is responsible for collecting the data
- Describing the data stream so users can search for it

There can be more than 50 different attributes defining a PI Point. Here are a few key attributes:

- **Name:** The name of the PI Point, which must be unique within the Data Archive.
- **Description**: A free text field attached to a PI Point, often used to enter a human friendly description of the PI Point. For example, a temperature point might be TC365674A.pv and the descriptor could be 'Reactor 65 Operating Temp'. Note that PI Points are not required to have a Description.
- **Point Type:** This attribute defines the type of data that is stored in the Data Archive.
- **Point Source**: This attribute commonly specifies which PI Interface is collecting the data for the PI Point.

Note: We will continue our discussion of PI Point Attributes in the section "Define the relationship between PI Point Attributes and PI Interface configuration"

1.4 Directed Activity – Search for PI Points using SMT



In this part of the class, you will perform a learning activity to explore the different concepts presented in this chapter or section. You may be invited to watch what the instructor is doing or perform the same steps at the same time. You may play a game or hold a quiz. Your instructor will have directions.

Activity Objectives

PI System users can use a variety of different programs to interact with the PI System.

As a PI System administrator, one of the applications you will use most is "System Management Tools". It is used by PI System administrators for a variety of management tasks. You will learn how to use System Management Tools (SMT) throughout the duration of the training.

In this directed activity, we will use SMT to search for PI Points and display current data for those PI Points.

Approach

Step 1 : From PISRV01, run the program "System Management Tools" 

Step 2 : Navigate to the tool Data > Current Values

Step 3 : Select the "Tag Search" icon 

Step 4 : Change the Point Source field to "R" and hit "Search"

Step 5 : Click "Select All" and then "OK"

Step 6 : To remove all the PI Points from the list, use the "Remove All" button 

What uses of the "Current Values" tool of SMT can you come up with, from a PI System administrator's perspective?

_____

_____

_____

_____

1.5 Using the Tag Search

The Tag Search functionality is similar in all PI System programs, such as SMT and other visualization tools. In SMT, the tag search function is used in the “Data”, “Points” and “IT Points” tabs. Users are able to search for PI Points by specifying values for various PI Point attributes. Here are a few tips and tricks related to using the Tag Search.

Using the PI Point Name (Tag Mask)

If your organization has a standard, convenient naming convention, or if you are very familiar with the PI Points in your plant, then it will be very easy to search for PI Points by their name. However, some people do not have that luxury.

Using the Descriptor

If the Descriptor attribute is always used when creating a PI Point, then the Descriptor is a good attribute to use when searching for PI Points. The downside to searching by Descriptor is that it can be resource intensive.

Using the Point Source

For a PI System administrator who is familiar with the PI System, searching by Point Source can be extremely helpful, as it allows you to bring up a list of all of the points that are associated with a specific PI Interface, and therefore a specific data source.

Wildcards

Remember that wildcards can be used in any of the above searches.

Use * to replace any number of characters like in this example:

flow* = flow_meter1, flow_meter2, flow_meter3, flowrate_pump1, flowrate_pump2

Use ? to replace one character like in this example:

flow_meter? = flow_meter1, flow_meter2, flow_meter3

Note: The Tag Search is not case sensitive

1.6 Solo Exercise – Use the Tag Search



This is a solo exercise that is designed to maximize learning in a specific topic area. Your instructor will have instructions and will coach you if you need assistance during the exercise.

Exercise Objectives

- Familiarize yourself with the Tag Search

Problem Description

As a PI System administrator, you will be using the Tag Search to answer questions about the current state of your PI System using System Management Tools.

Approach

1. How many PI Points have been created for “Reactor 1” so far?

2. Do the PI Points with Point Source “L” have recent values?

3. Search for all PI Points. Do we have a single naming convention for our PI Points?

4. Which PI Point has a PointID of 39?

1.7 Directed Activity – View PI Point data using PI Vision



In this part of the class, you will perform a learning activity to explore the different concepts presented in this chapter or section. You may be invited to watch what the instructor is doing or perform the same steps at the same time. You may play a game or hold a quiz. Your instructor will have directions.

Activity Objectives

To learn how to administer a PI System, it is important to understand the point of view of a PI System user. A popular PI System visualization tool is “PI Vision”. PI Vision allows users to access PI System data through a web browser and rapidly create ad-hoc displays to view their data.

In this directed activity, we will be simulating the experience of a PI System user searching for the last 12 hours of Reactor 1 temperature data using PI Vision.

Approach

As a PI System user, you want to see a trend of the temperature in Reactor 1 for the past five hours.

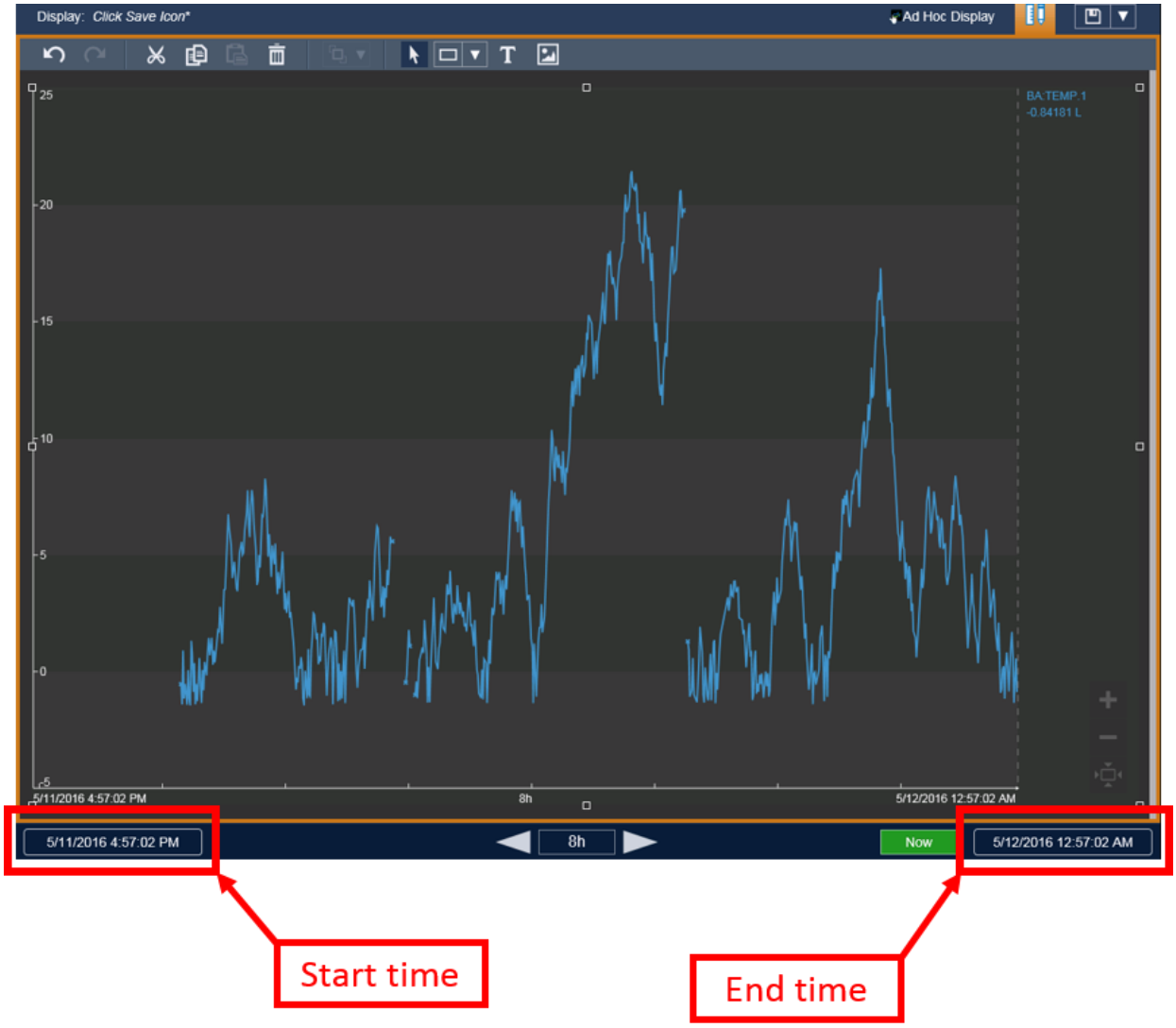
Step 1 : From PISRV01, click on the “PI Vision” bookmark that you can find on the desktop.

Step 2 : On the PI Vision homepage, select the “New Display” button in the upper-right hand corner: 

Step 3 : When using PI Vision to search for PI Points, we can either use the PI Point name or description. In the upper left-hand corner, search “reactor 1 temperature”.

Step 4 : Drag and drop the PI Point onto the display. This creates a trend symbol. Resize the trend as needed.

Step 5 : On the lower left-hand and right-hand corners of the display, we see the start and end times of the trend:



Click on the start time and change the text from "*-8h" to "*-12h".

Step 6 : Click on the save icon  at the top right of the display, and name your display "Reactor 1 Temperature"

Now imagine that the Descriptor had not been filled out for this PI Point. How could the user find the correct data? In chapter 4, we will see how a PI System administrator can build a user-friendly PI System using the Asset Framework.

1.8 Writing Time in the PI System

The Data Archive stores data that changes with time, known as time-series data.

As we saw in the previous directed activity, when users request data from the PI System, they need to know how to request the specific time, or time range, for the data they want to see.

1.8.1 Fixed time vs. Relative time

There are two options for specifying time in the PI System:

- **Fixed Time**: An expression that signifies a specific date and time, which will never change.

  When to use it: When you want to save a view of your PI System data for a specific time in history.

  Example: A user is creating a report that analyzes an equipment failure event which occurred on the 5th of January.

- **Relative Time**: An expression that signifies a date and time relative to the current date and time.

  When to use it: When you want to create a *dynamic* view of your data, which can be used to view data in real-time, or re-used on a periodic basis to create periodic reports.

  Example: A user is creating a report that summarizes the weekly production totals. By using relative time expressions, the user will be able to re-use this report every week.

1.8.2 Fixed time expression syntax

A fixed time expression is an expression which includes a date, and optionally a time. If the time is omitted, midnight is assumed.

Expression	Meaning
23-aug-12 15:00:00	3:00 p.m. on August 23, 2012
25-sep-12	00:00:00 (midnight) on September 25, 2012

The PI System can interpret many different formats for fixed time. In the event of an ambiguous input, the Windows Region and Language settings of the computer where the PI Visualization Tool is installed take precedence. For example:

Expression	Region and Language Format	Meaning
1/5/2015	English (United States)	00:00:00 (midnight) on January 5th 2015
1/5/2015	English (Canada)	00:00:00 (midnight) on May 1st 2015

1.8.3 Relative time expression syntax

These expressions are used to signify a date and time *relative* to the current time. PI System time expressions might include:

- Only a reference time, such as "y"
- Only a time offset, such as "+3h"
- A reference-time with a time offset, such as "y+3h"

Reference-time abbreviations

A reference-time abbreviation represents a specific time relative to the current time

Abbreviation	Full	Reference-time
*		Current time.
t	today	00:00:00 (midnight) of the current day
y	yesterday	00:00:00 (midnight) of the previous day
sun	sunday	00:00:00 (midnight) on the most recent Sunday
mon	monday	00:00:00 (midnight) on the most recent Monday
tue	tuesday	00:00:00 (midnight) on the most recent Tuesday
wed	wednesday	00:00:00 (midnight) on the most recent Wednesday
thu	thursday	00:00:00 (midnight) on the most recent Thursday
fri	friday	00:00:00 (midnight) on the most recent Friday
sat	saturday	00:00:00 (midnight) on the most recent Saturday
YYYY		00:00:00 (midnight) on the current day and month in year *YYYY*
M-D or *M/D*		00:00:00 (midnight) on the *D*th day of month *M* in the current year
DD		00:00:00 (midnight) on the *DD*th day of the current month

Time-unit abbreviations

A time-unit abbreviation represents a specific unit of time, which can be used to define an offset.

Abbreviation	Time-unit
s	second
m	minute
h	hour
d	day
w	week
mo	month
y	year

Reference-time and offset expression

When included with a reference-time abbreviation, a time offset adds or subtracts from the specified time (indicated by either + or -) and a time unit with a value

Expression	Meaning
*-1h	One hour ago
t+8h	08:00:00 (8:00 a.m.) today
y-8h	16:00:00 (4:00 p.m.) the day before yesterday
mon+14.5h	14:30:00 (2:30 p.m.) last Monday
sat-1m	23:59:00 (11:59 p.m.) last Friday

Time Offsets

Entered alone in a time field, time offsets specify a time relative to an implied reference-time. The implied reference-time depends on the field where you enter the expression:

- For a start time, the reference-time is the current clock time.
- For an end time, the reference-time is the start time.
- For a single time stamp, the reference-time is the current clock time.

Time field	Expression	Meaning
Start time	-1d	One day before the current clock time (24 hours before the current clock time)
End time	+6h	Six hours after the start time
End time	-30m	30 minutes before the start time
Time stamp	-15s	15 seconds before the current clock time

1.8.4 Rules for creating time expressions

Rule 1. You should only include a single time offset in an expression. Including multiple offsets can lead to unpredictable results. For example, the following time expressions should be avoided:

*+1d+4h
t-1d+12h

Rule 2. To define a time offset you must include a valid value with any time unit. Only for *seconds*, *minutes* or *hours*, you can specify a fractional value. You cannot specify fractional values for other time units.

Rule 3. A fixed timestamp consists of the fields of Year, Month, Day and Time (hours, minutes and seconds). If any of these fields are not specified in the PI time expression, the following values will be assumed by default:

- If Time is not specified, then the default value would be Midnight.
- If Day is not specified, then the default value would be Current Day.
- If Month is not specified, then the default value would be Current Month.
- If Year is not specified, then the default value would be Current Year.

1.8.5 Group Exercise – Translate relative time expressions



This is a group exercise that is designed to maximize learning in a specific topic area. Your instructor will have instructions and will coach you if you need assistance during the exercise.

Exercise Objectives

- Translate relative time expressions into their meaning
- Create relative time expressions
- Use relative time expressions in PI Vision

Problem Description

Translate the relative time expressions below into their meanings:

Expression	Meaning
* - 30m	
y + 8h	
T	
Y	
Thu	
Tuesday – 2d	
18	
y-2y	

Express the following times in valid PI System time abbreviations:

Expression	Meaning
	Today at 6:00 AM
	Monday at 6:30 am
	12 hours ago
	The first day this month
	The beginning of the year
	7:00 am yesterday
	15 minutes ago

Use your knowledge of PI System time abbreviations to trend data on your “Reactor 1 temperature” display:

1. View the data from yesterday at midnight to today at midnight
2. View the data for the first operator shift yesterday, from 8:30 AM to 4:30 PM
3. View the data from Sunday at midnight last week to Sunday at midnight this week.

1.8.6 How Does the PI System Adjust for Time Zones and DST?

The short answer is it does not!

When data is collected by the PI System, it is collected in UTC (Universal Coordinated Time), or what used to be called Greenwich Mean Time (GMT). This means that each day has exactly 24 hours. The PI System user's computer makes any adjustments for time based on the local date and time settings, such as time zone or DST.

If your region observes DST, once a year the day will look like it has 23 hours and another 25, but the Data Archive never knows anything other than 24-hour days.

In addition, because the clients and the Data Archive know what time zone they are in, the data can be viewed with respect to the s*erver time* or the c*lient time*. This is determined by a setting in the PI Visualization tool.

1.8.7 Future Data

Data Archive version 2015 introduced the capability of storing "future data" in the Data Archive. Future data is data with a timestamp in the future. The Data Archive can now store data within a time range of January 1970 through January, 2038.

How might future data be useful? For example, if you have forecasting software which predicts the production of your plant, you can save this data in a "future" PI Point on the Data Archive. As you collect the actual production data in another PI Point, you can compare the two in real-time.

When creating a PI Point, a "Future" attribute determines whether the PI Point is created as a "historic" or "future" PI Point. After creation, a PI Point can never be switched from one to the other. Therefore, you never over-write your future data with historic data, the two datasets are always kept separate.

To request data for a future timestamp in a tool like PI Vision, you can enter the same type of expressions as discussed in the previous sections (using fixed time, or relative time). Some example expressions are:

Expression	Meaning
*+1h	An hour from now
t+3d	Three days from today at midnight
Y+1y	A year from yesterday

2. PI Interface Management

Objectives

- Define the role of a PI Interface
- Select the correct PI Interface for a given data source
- Discuss the variety of architecture possibilities
- Describe the PI Interface Configuration Utility
- Create a PI Point for an existing PI Interface
- Describe PI Interface installation and configuration methodology
- Install and configure a new PI Interface for OPC DA instance
- Create a PI Point using SMT
- Create a PI Point using PI Builder
- Describe PI Buffering
- Explain data flow through the PI Interface node
- Configure and validate PI Buffering

2.1 PI Connectors and PI Adapters

Throughout the first chapter, we referenced PI Interfaces, PI Connectors and PI Adapters as components that collects data from a data source. Here are the main differences between these. For more details, refer to this article.

Feature	PI Adapters	PI Connectors	UniInt Interfaces
OS Requirements	Windows and Linux-based	Windows	Interface-specific (mostly Windows)
Configuration tool	Any application capable of making REST calls (remotely & locally)	Web browser (remotely & locally)	PI Interface Configuration Utility (locally only)
Automatically Creates AF Assets	Yes, only for health and diagnostics	Yes	No
Automatically Creates PI Points	Yes	Yes	In general no

This chapter will focus solely on PI Interfaces as they are still most common application for data collection.

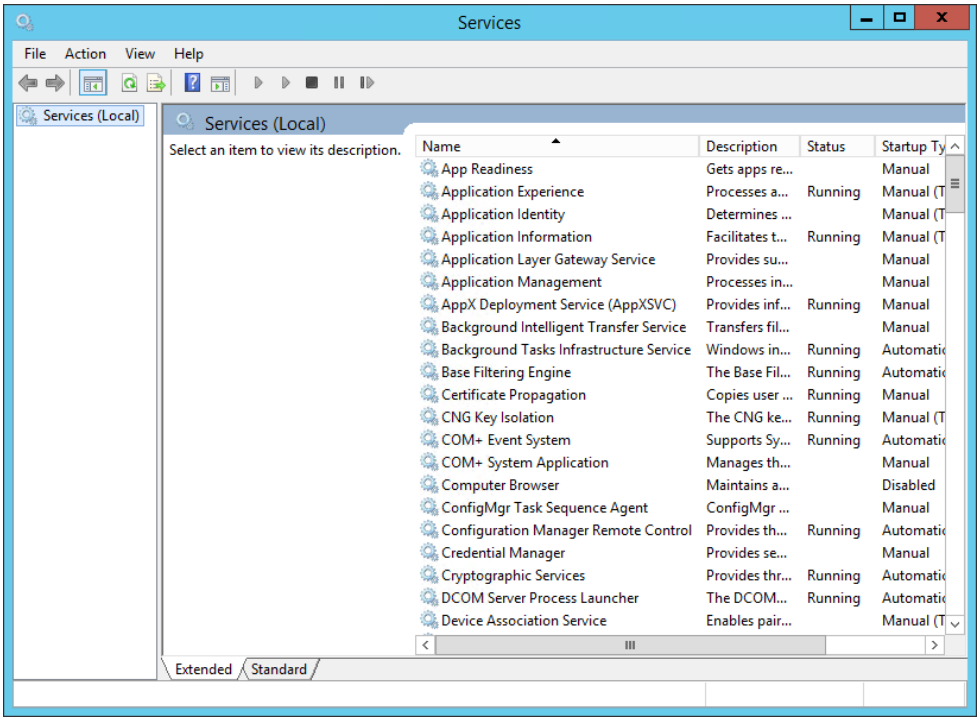
You can find examples of PI Connector installation and PI Adapter installation on our YouTube channel.

2.2 Understanding important IT Concepts

This class was designed for individuals with a basic understanding of IT fundamentals. If you are not an IT professional, there are a few basic concepts to grasp before moving forward.

2.2.1 Windows Service

A Windows Service is a computer program, or application, that runs in the background on a Windows Operating System. These computer programs do not require user interaction to run. Windows Services are mostly managed through the “Services” snap-in (services.msc).



2.2.2 Windows Domain

A Windows Domain is a Windows computer network where all the users and computers are registered in a central database, called the Active Directory. The computer on which the Active Directory runs is called the Domain Controller.

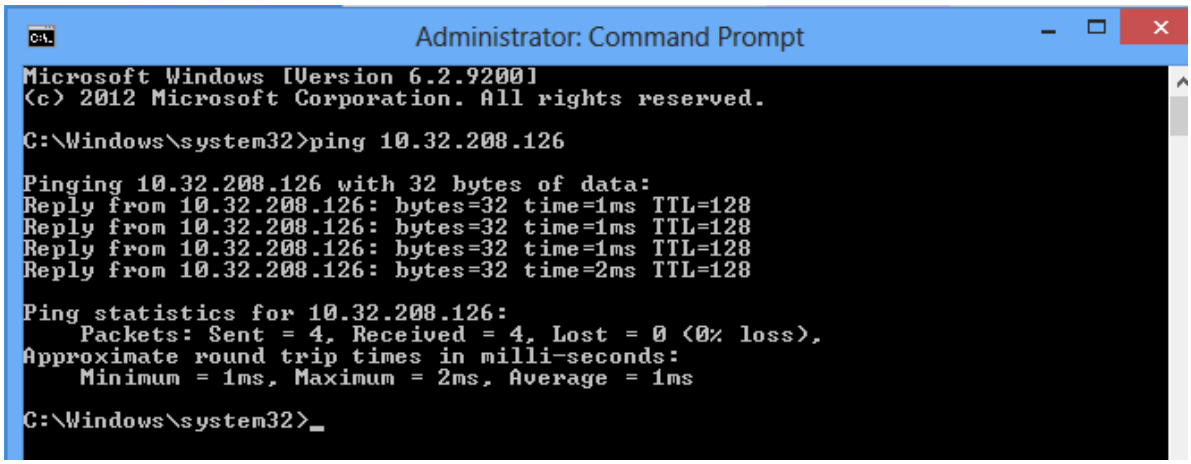
Resources on the domain (e.g., databases) can take advantage of the Active Directory to manage user security.

2.2.3 Port

On a computer network, a port is an endpoint of communication. It is used by the Operating System to direct incoming data (or, more specifically, packets of data) to the correct computer program or service. When communicating over a network, it is important to know the port number used by the application or service we are trying to reach.

2.2.4 Windows Command Prompt

The Windows Command Prompt is the command-line interface of a Windows Operating System. It can be used to issue commands to the Operating System in the form of successive lines of text. The Windows Command Prompt application name is cmd.exe.



2.2.5 Windows PowerShell

Like the Windows Command Prompt, Windows PowerShell is a command-line interface, but it is also an object oriented scripting language. It was designed by Microsoft to address the limitations of the Windows Command Prompt, making it easier to create scripts that automate management tasks in Windows.

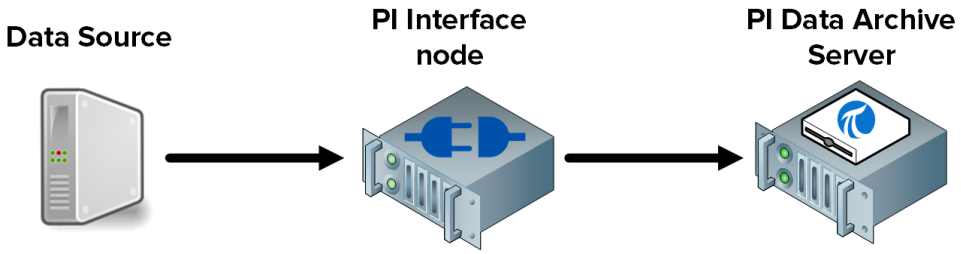
2.2.6 Client and application

A server or an application is the program that performs certain actions. A client is a utility to interact with this program. For example, Google servers search the worldwide web for certain words. We use our web browsers to request and see the results. An example from PI System is PI Data Archive and PI System Management Tools (PI SMT). PI Data Archive is an application that consists of several components that receives, stores and delivers data. We use PI System Management Tools to configure PI Data Archive. We can configure PI Data Archive from a remote machine using PI SMT.

Application or server	Client
Performs the tasks (can work without a client)	**Used to interact with server**
Google Search Server	Web browser
Email Server	Outlook
PI Data Archive	PI System Management Tools (PI SMT)
PI Asset Framework	PI System Explorer
PI Interface	PI Interface Configuration Utility (PI ICU)

2.3 Define the role of a PI Interface

In the section “What is a PI System?” we learned that a PI Interface is one of the crucial software components of a basic PI System. It is responsible for collecting data from a data source and sending it to the Data Archive. Each PI Interface is responsible for collecting data for specific PI Points on the Data Archive.



AVEVA has released over 450 different PI Interfaces that collect data from a wide variety of data sources. Almost anything that generates time-series data can be a data source, including web pages, relational databases, and other PI Systems. However, typically, data from a plant’s process is collected from *DCS, PLC* and *SCADA systems*. All these systems can send data over a network but do so using a wide variety of communication protocols. The PI Interface can be thought of as a translator. It reads data from a data source and translate what it reads into a language the Data Archive can understand.

Note: With a few exceptions, the Data Source is not designed or published by AVEVA.

No matter which PI Interface is used, here are the steps it will perform when collecting data:

Step 1 : Reads from the data source
Step 2 : Timestamps the data (or ensures the data received is stamped at the source)
Step 3 : Formats the data
Step 4 : Applies Exception filtering
Step 5 : Sends the data to the Data Archive

Note: We will continue our discussion of Exception filtering in the section “Understanding Exception and Compression”

2.5 Solo Exercise – Choose a PI Interface



This is a solo exercise that is designed to maximize learning in a specific topic area. Your instructor will have instructions and will coach you if you need assistance during the exercise.

Exercise Objectives

- Select the correct PI Interface for a given data source

Problem Description

You are a PI System administrator of a brand-new PI System. The plant manager gives you a list of data sources from which he wants to collect data. For each data source, find the appropriate PI Interface.

Data Source	PI Interface
Siemens PLC 412-2	
Werum Pas-X	
Schneider PML 3710ACM	
Johnson Controls Metasys System	
A web page	
Text files (HINT: what is the most common type of text file encoding?)	

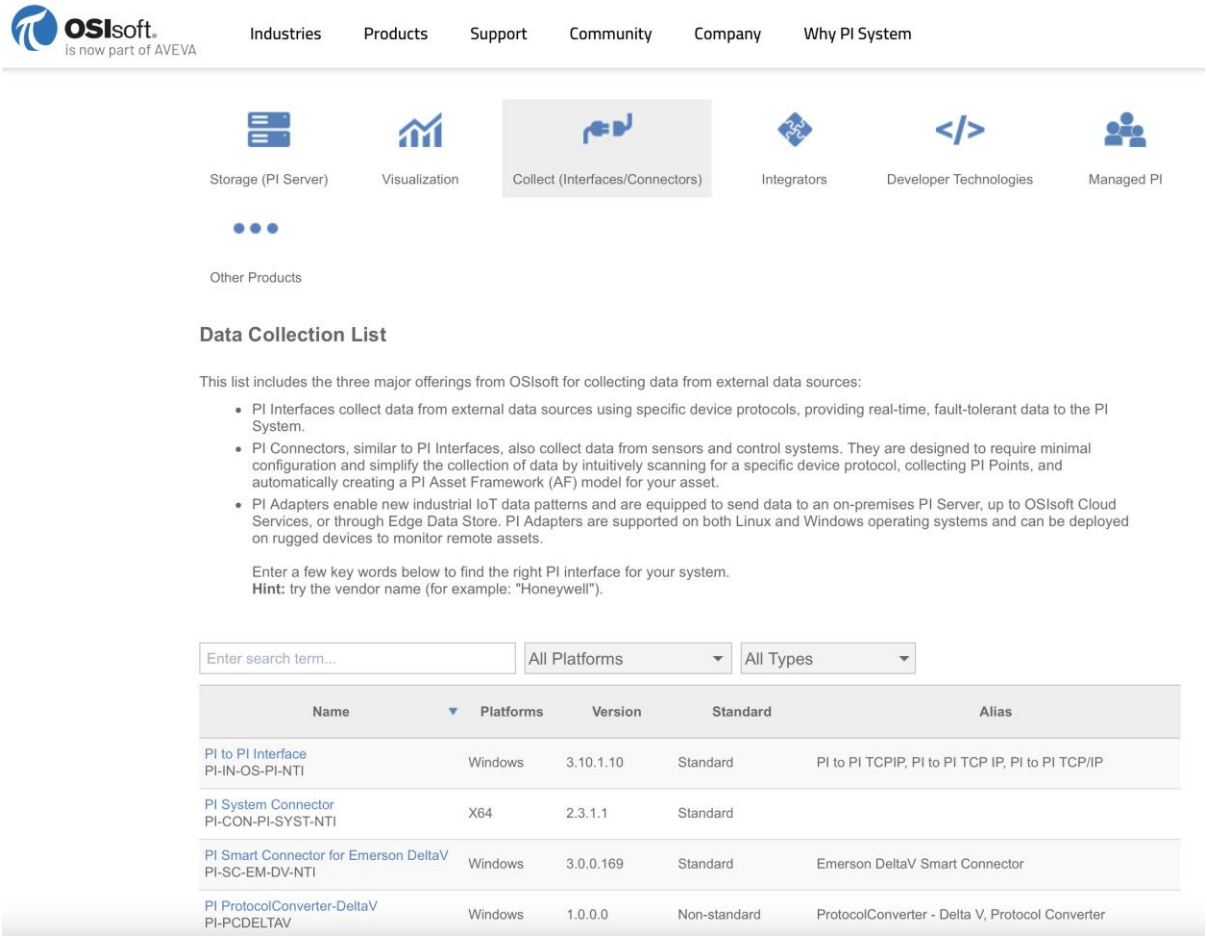
Approach

With over 300 active PI Interfaces to choose from, and countless possible data sources in your plant, selecting the correct PI Interface can be a daunting task. AVEVA provides a tool on its Technical Support website to help PI System administrators make their choice.

Step 1 : Browse to https://techsupport.osisoft.com/Products/

Step 2 : Click on Collect (interfaces/Connectors)

Step 3 : Use the search bar to look up the above data sources.



Note: Sometimes, this tool will not find any PI Interface for the data source entered. However, this does not mean that it is impossible to collect data from this source. Often, connecting to and reading from the data source requires knowledge of the data structure and format. You will need to use the data source manufacturer's documentation. If you need help deciding on a PI Interface, contact our Technical Support team through the customer portal: my.osisoft.com

2.6 Common PI Interfaces

As we saw in the previous exercise, some of our PI Interfaces are designed for a specific data source, while others are built using standard communication protocols. Below is a list of our most popular PI Interfaces.

1. **PI Interface for OPC DA**

   Collects real-time data from OPC Servers using the OPC DA standard, arguably the most common communication protocol in the industrial automation industry.

2. **PI Interface for Universal File and Stream Loading (UFL)**

   Collects real-time, historic or future data from ASCII files (e.g. txt, csv xml files, etc.), serial ports and POP3 email servers. The PI Interface can be configured to collect data no matter the format of the data in the source file, making it one of our most versatile interfaces.

3. **PI Interface for RDBMS**

   Collects real-time, historic or future data from any Relational Database Management System that supports ODBC drivers (e.g. Microsoft SQL Server, Oracle Database, IBM Informix, etc.)

4. **PI Interface for Modbus Ethernet PLC**

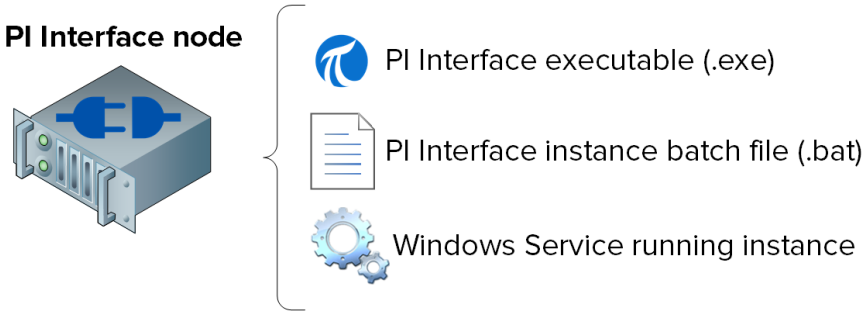
   Collects real-time data from PLCs that use the Modbus communication protocol

5. **PI to PI Interface**

   This PI Interface is used to send real-time, historic or future data from one Data Archive to another. A typical application of this PI Interface is to collect data from plant-level Data Archives to a centralized corporate Data Archive.

2.7 Define the components of a PI Interface

Once a PI Interface has been installed and configured on a computer, it consists of the following components:



- **PI Interface executable**: This is the executable file that will run and perform the operations to collect data from the data source.
- **PI Interface instance batch file**: Although the executable performs all of the operations, it needs instructions, such as (1) which data source to collect from, (2) which Data Archive to send to, etc. A batch file will contain all of these instructions.

Since you could have multiple data sources, you can create multiple batch files, and hence multiple *instances* of the PI interface running on a single node.

- **Windows Service running PI Interface instance**: For an instance of the PI Interface to run automatically when the computer starts, and run in the background, a Windows Service is created.



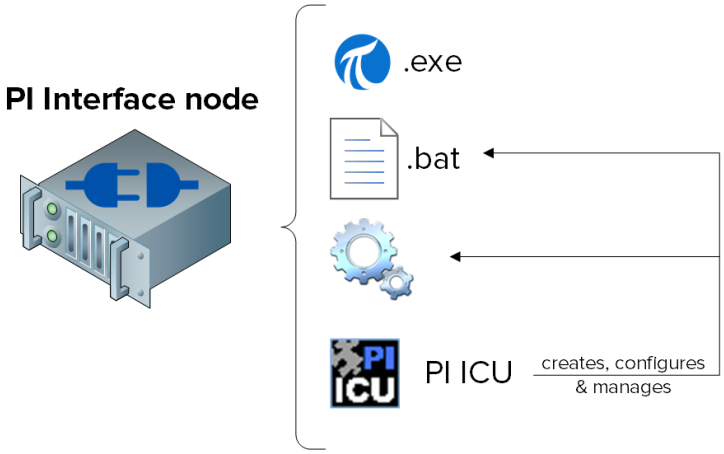
Tip

Checking the services snap-in (services.msc) is a great way of identifying all the PI Interfaces instances that are running on a single PI Interface node.

2.8 Define the PI Interface Configuration Utility

The PI Interface Configuration Utility (ICU) is a Graphical User Interface (GUI) that PI System administrators use to create and configure the PI Interface instance batch files and services.

The PI ICU will only configure batch files and services located on the computer where it is installed (it cannot be used to configure remote PI Interfaces).



Note: Once the PI ICU is used to configure a batch file, the contents of this file are written to a database on the Data Archive called the "Module Database" (MDB), which stores configuration information for the Data Archive. This allows you to recover the configuration of your PI Interface instance. However, if a batch file is manually edited, the PI ICU will throw a warning message.

2.9 PI Interface installation methodology

Every time a new PI Interface needs to be used for data collection, the following PI Interface installation methodology should be employed:

Step 1 : *Choose a PI Interface for the data source*

Step 2 : *Choose where to install the PI Interface*

Step 3 : *Install the PI Interface, the PI ICU and the PI API for Windows Integrated Security*

Step 4 : *Validate that the PI Interface can communicate with the Data Archive*

Step 5 : *Validate that data is available on the data source for the PI Interface to read*

Step 6 : *Configure security for the PI Interface on the Data Archive*

Step 7 : *Create and configure an instance of the PI Interface*

Step 8 : *Create PI Points for the PI Interface.*

These first 8 steps are the basic steps required to start collecting data. However, a few additional steps are needed to ensure reliable data collection in a production environment:

Step 9 : *Configure buffering with the PI Buffer Subsystem*

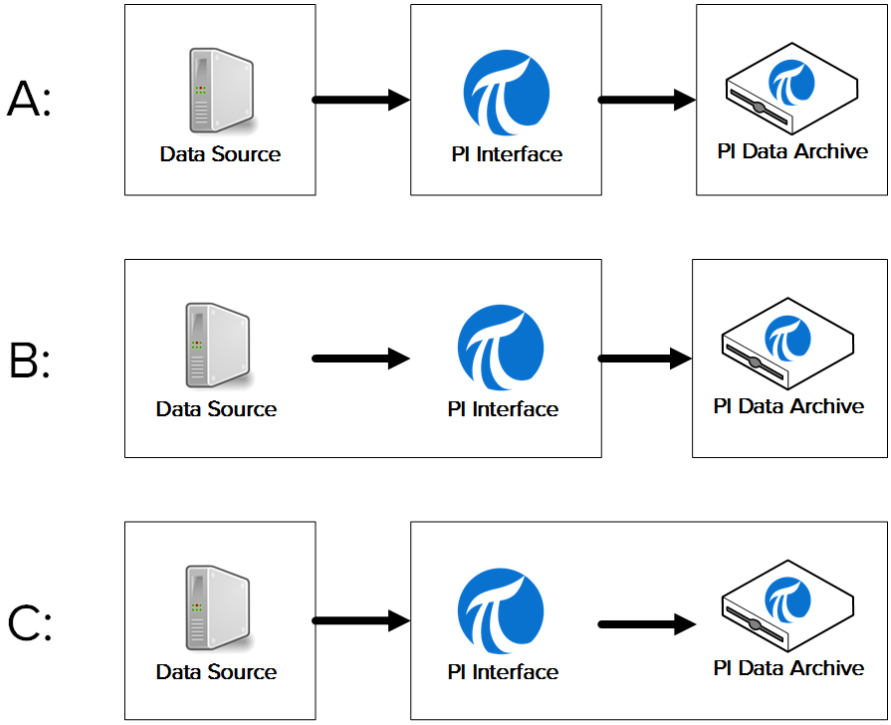
Step 10 : *Create PI Interface Health Points to monitor the health of the PI Interface*

2.10 Group Questions – PI Interface architecture



The following questions are intended to reinforce key information, or to discover a new insight. Your instructor may choose to have you try to answer the questions on your own or have the group answer them together aloud.

As we saw in the previous exercise, the data source is almost never on the same computer as the Data Archive. That being said, there are several possible PI Interface architectures:



- **Architecture A**: The data source, the PI Interface and the Data Archive are all installed on separate machines.
- **Architecture B**: The data source and the PI Interface are installed on the same machine.
- **Architecture C**: The PI Interface is installed on the Data Archive server.

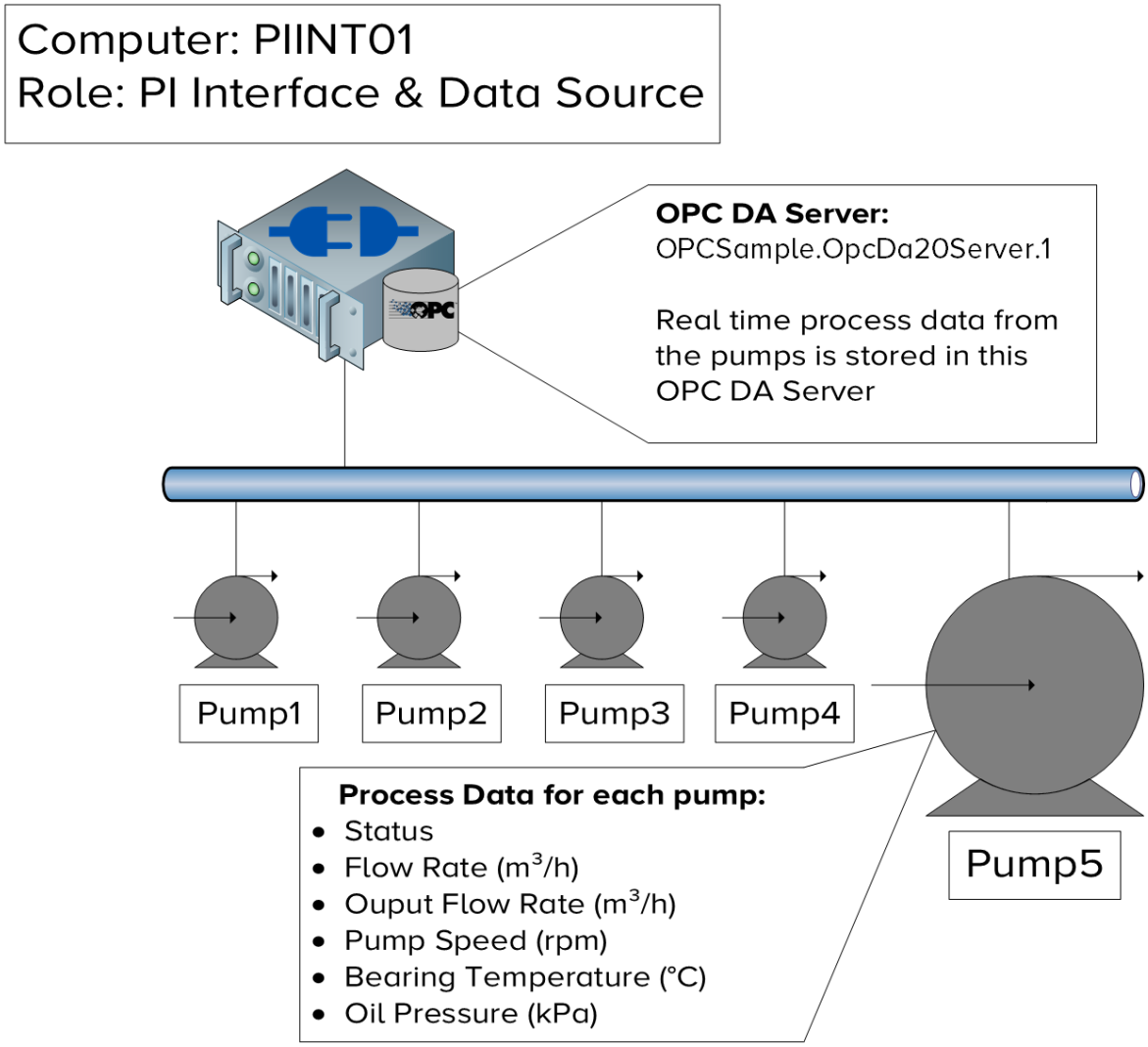
As a group, discuss the advantages, disadvantages and example application for each architecture:

Architecture	Advantages	Disadvantages	Example application
A			
B			
C			

2.11 Install and configure a PI Interface for OPC DA

Now that we are familiar with the PI Interface installation methodology, we can install and configure a new PI Interface to collect data in our Virtual Learning Environment. We will be doing so in the directed activities and exercises throughout the remaining sections of this chapter, by following the steps outlined in the section “PI Interface installation methodology”

Our data source is an OPC DA Server installed on PIINT01 (therefore, we are using architecture B from the previous group discussion). This OPC DA Server exposes real-time data from 5 pumps in our process. Our goal will be to collect this process data and store it in the Data Archive. We will be installing our PI Interface on PIINT01. Since we have already chosen our PI Interface and our architecture, we have completed steps 1 and 2 of the installation methodology.

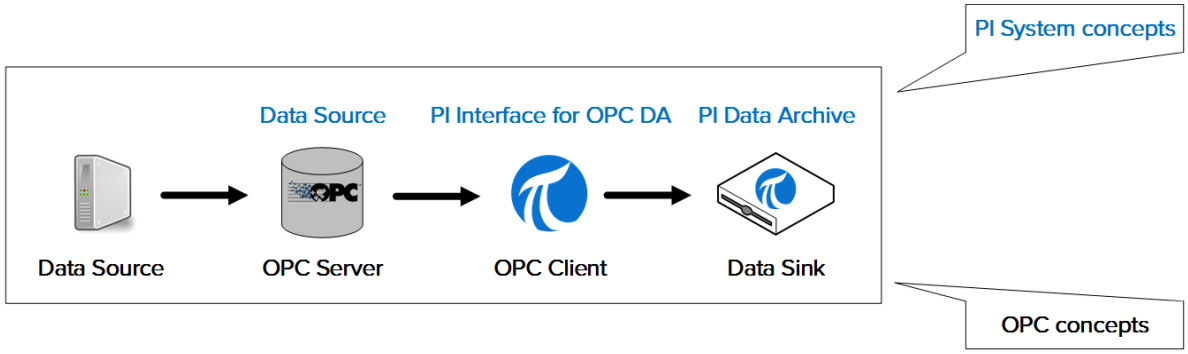


2.11.1 What is an OPC DA Server?

We chose an OPC DA Server as the data source for this class because it is the most common data source among our customers, making the PI Interface for OPC DA our most widely used interface.

OPC DA is a standard communication protocol developed for the industrial automation industry. As previously discussed, automation systems communicate using a wide-variety of different, often proprietary protocols. This makes communication between different systems very difficult. To solve this problem, multiple vendors got together and developed a series of platform-independent standards called OPC (Open Platform Communication). OPC DA is the standard for real-time data collection.

There are two required software components when communicating using an OPC standard: the OPC Server and the OPC Client. The OPC Server is a software application that exposes data from a data source in the OPC standard. The OPC Client is a software application that consumes data from an OPC Server and translates it to a different format. *The PI Interface for OPC DA is an OPC Client*. The OPC Server is a non-AVEVA application developed by another company.



2.11.2 Directed Activity – Install the PI Interface for OPC DA and the PI ICU



In this part of the class, you will perform a learning activity to explore the different concepts presented in this chapter or section. You may be invited to watch what the instructor is doing or perform the same steps at the same time. You may play a game or hold a quiz. Your instructor will have directions.

Activity Objectives

Complete steps 3 and 4 from the PI Interface installation methodology we outlined in this chapter.

> ***Step 3 :*** *Install the PI Interface and the PI ICU*
>
> ***Step 4 :*** *Validate that the PI Interface can communicate with the Data Archive*

Approach

Part 1 – Install the PI ICU and the PI Interface for OPC DA

Step 1 : On PIINT01, navigate to the folder C:\Class Folder\Install Kits

Step 2 : Right-click on the installation kit "PIICU_x.x.xx.xx_.exe" and select "Run as administrator".

Step 3 : Complete the steps in the installation wizard.

Step 4 : Repeat steps 2 and 3 with the following installation kits:

a. OPCInt_ReadOnly_x.x.x.xx_.exe

b. PIAPIxxxx _x.x.x.xx_.exe

Note: The first install kit installs the Read-only version of the PI Interface for OPC DA. This version does not have the ability to write data back to the OPC Server. The read-only version is strongly recommended as an inherently safer technology that also simplifies compliance with security policy.

The second install kit installs the PI API for Windows Integrated Security. Although the PI OPC Interface comes with the PI API, this version is more secure. We will continue our discussion of PI API security in the section "PI System Security Management".

Part 2 – Validate that the PI Interface node can communicate with the Data Archive server over the network

Step 1 : We will first test to see whether network packets can travel from the PI Interface node to the Data Archive Server. On PIINT01, run the command prompt and use the **ping** command to test the connectivity to PISRV01.

Step 2 : Next, we will test to see whether network packets can travel from the Data Archive Server to the PI Interface node. On PISRV01, run the command prompt and use the **ping** command to test the connectivity to PIINT01.

Step 3 : Data sent to the Data Archive uses TCP port 5450. The final test is to verify whether that ping port is open on the Data Archive Server. On PIINT01:

a. Run the Windows Powershell application

b. Run the following command:

tnc pisrv01 -port 5450

If port 5450 is open, you will see the message

```
PS C:\Users\student01.PISCHOOL> tnc pisrv01 -port 5450


ComputerName     : pisrv01
RemoteAddress    : 192.168.0.6
RemotePort       : 5450
InterfaceAlias   : Ethernet 10
SourceAddress    : 192.168.0.8
TcpTestSucceeded : True
```

If port 5450 is blocked, you will receive an error message:

```
PS C:\Users\student01.PISCHOOL> tnc pisrv01 -port 5450
WARNING: TCP connect to (192.168.0.6 : 5450) failed
WARNING: Ping to 192.168.0.6 failed with status: TimedOut


ComputerName            : pisrv01
RemoteAddress           : 192.168.0.6
RemotePort              : 5450
InterfaceAlias          : Ethernet 10
SourceAddress           : 192.168.0.8
PingSucceeded           : False
PingReplyDetails (RTT)  : 0 ms
TcpTestSucceeded        : False
```

Part 3 – Testing the two PI System connection protocols

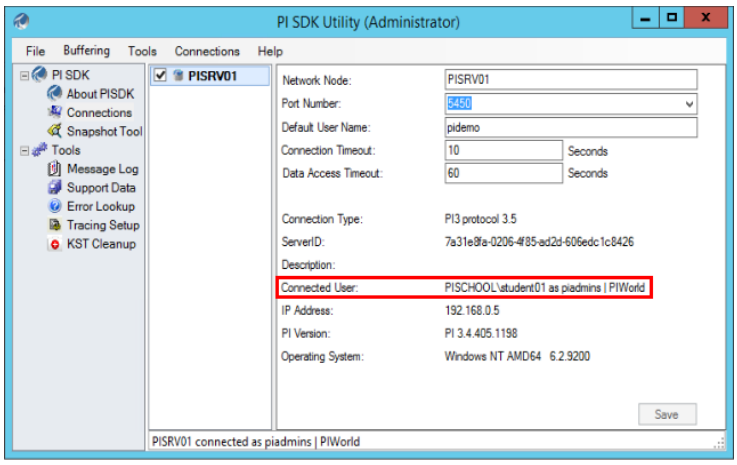
There are two connection protocols that can be used to connect to the Data Archive: The older PI API and the newer PI SDK. PI Interfaces are typically built to use the PI API when sending data. Newer PI System software, such as the PI ICU, are built to use the PI SDK. Therefore, both need to work correctly on a PI Interface node. We will now verify whether we can connect to the Data Archive from the PI Interface using these protocols

Step 1 : First, we will test a connection using the PI SDK.

a. Run the PI SDK Utility (AboutPI-SDK)

b. In the pane on the left side of the window, select “Connections”

c. In the second pane on the left, you should see the Data Archive name “PISRV01”. Click the checkbox next to the name. If the connection is successful, you should see your username and “connected as” in the “Connected User” Field.



Step 2 : Finally, we will test the PI API protocol

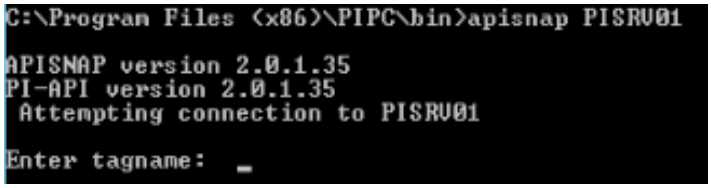
a. Run the command prompt

b. Navigate to the directory C:\Program Files (x86)\PIPC\bin

Hint: type “cd %pihome%\bin”

c. Run the command **apisnap PISRV01**

d. If the connection is successful, you should see the message:



e. Enter the tagname “sinusoid”. Do you get a value?

Note: We will continue our discussion of the PI API and PI SDK protocols in the section “PI System Security Management”.

2.11.3 Validating the availability of data on the OPC DA Server

The PI System is responsible for reliably collecting and storing data. However, there is not much the PI System can do if data is not available at the data source. This is one of the most common issues with newly installed PI Interfaces, so it is important to validate data availability before moving on to PI Interface configuration.

When the data source is an OPC DA Server, AVEVA provides a tool for this step, called the PI OPC Client Tool, which is installed with the PI Interface for OPC DA. In the section “What is an OPC DA Server?” we explained the concepts “OPC Server” and “OPC Client”. The PI OPC Client Tool is an OPC client published by AVEVA, designed to allow users to *view* data on the OPC Server, without collecting it.

Of course, the PI OPC Client Tool is not the only OPC Client that can be used to view data. Most OPC Server vendors include an OPC Client with the installation of the OPC Server. It is also recommended to test the availability of data on the OPC Server using this vendor-specific OPC Client.

2.11.4 Solo Exercise – Using the PI OPC Client Tool



This is a solo exercise that is designed to maximize learning in a specific topic area. Your instructor will have instructions and will coach you if you need assistance during the exercise.

Activity Objectives

Complete step 5 from the PI Interface installation methodology we outlined in this chapter:

> ***Step 5 :*** *Validate that data is available on the data source for the PI Interface to read*

Approach

Part 1 – Validate that you can connect to the OPC Server

Step 1 : From PIINT01, run the program "PI OPC Client Tool" 

Step 2 : In the top left corner, the field with "Localhost" is reserved for the address of the computer node where the OPC Server is installed. Since the OPC Server is installed locally, we will keep "Localhost" and connect by pressing the "Connect to node" button. 

Step 3 : A list of OPC Servers will appear under the "OPC Servers" field. Select OPCSample.OpcDa20Server.1 and click on the "Connect to OPC Server" button. 

Step 4 : If the connection is succesful, you should see a good server status under the "Server Status" field. The Server Current State should be "RUNNING"

Part 2 – Validate that OPC Tags that are available on the OPC Server

Step 5 : Click the "Add Group" button . In the "Add Group" dialog, click "Create".

Step 6 : Click the "Browse OPC Server, Add Tags" button 

Step 7 : The Add Item window will appear. This is the window that allows you to see what data is available on the OPC Server. Click the "List" button at the top-left of the window.

Step 8 : We are now browsing the server, and see the hierarchy of data available on the OPC Server. The data has been organized under 5 pumps. Select one of the pumps.

Step 9 : We now see the OPC Tags available for the selected pump on the right hand side. We must now verify whether the OPC Tags have good data. Click "Select All" and "Add Selected" under the OPC Tags. They should now be in the "Added Tags" field. Click "OK" on the lower right corner of the Window.

Note: This step does NOT add PI Points to the PI Interface, it simply adds OPC Items to the OPC client window so we can view the data.

Step 10 : You should now have returned to the main PI OPC Client window, with the tags you selected listed under Group1. To verify the current value of these tags, click the “Polling on Group” button. 

Step 11 : This should open the “Polling Group: Group1” window. Here, you will see a list of the pump tags you selected, with the current value, timestamp and quality. We must ensure that the quality is good, and that the value makes sense.

Note: We will use the PI OPC Client Tool again when configuring PI Points for the PI Interface for OPC DA

2.11.5 Ensuring proper authentication and authorization of the PI Interface on the Data Archive

In the previous sections, we ensured that:

- The PI Interface node can communicate over the network with the Data Archive *server*
- Data is available on the data source

The last step before configuring our PI Interface instance will be to ensure that the PI Interface:

- Has permission to connect to the Data Archive *application*
- Has permission to carry out its task once connected, which is to write data to the appropriate PI Points on the Data Archive.

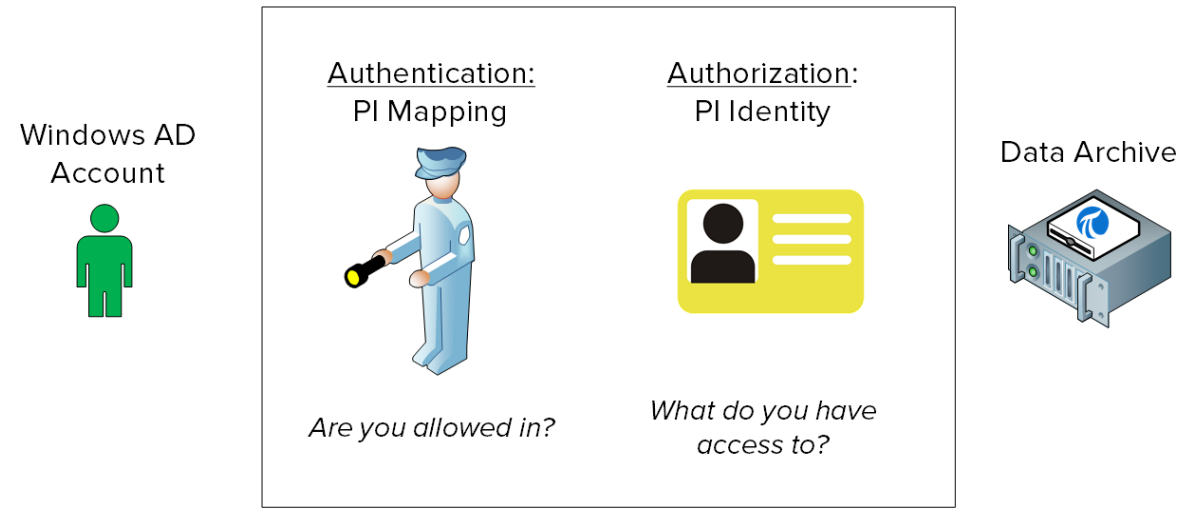
Although we will review PI System security in depth in a later chapter, it is important to perform a brief overview now, in order to properly configure our PI Interface.

Authentication vs. Authorization

In the context of the PI System:

- Authentication is the process that verifies the identity of a user or process, before allowing it to connect to the Data Archive
- Authorization is the process that determines what an application can do once connected to the Data Archive.

Typically, when software connects to the Data Archive, their Windows Active Directory account is authenticated by a **PI Mapping**. The PI Mapping assigns them a **PI Identity**, which grants them specific rights (authorization) on the PI System. The PI Mappings are like security guards at the entrance of a facility. They allow you to enter the facility, and give you an access badge (PI Identity) that gives you access to specific rooms once inside the facility.



2.11.6 Directed Activity – Create a PI Mapping for the PI Interface for OPC DA



In this part of the class, you will perform a learning activity to explore the different concepts presented in this chapter or section. You may be invited to watch what the instructor is doing or perform the same steps at the same time. You may play a game or hold a quiz. Your instructor will have directions.

Exercise Objectives

Complete step 6 from the PI Interface installation methodology we outlined earlier in this chapter:

> ***Step 6 :*** *Configure security for the PI Interface on the Data Archive*

Approach

We will be creating two PI Mappings that will allow your PI Interface to connect to the Data Archive.

AVEVA recommends using designated Windows Service Accounts to run PI System services that communicate over a network. Before starting, you made a request to your IT department to create a Service Account:

- PISCHOOL\svc-PIInterface (password: student)

Step 1 : Before we begin, let's see what happens when a PI API connection is made without the proper security.

a. From PIINT01, run the command prompt as the user "svc-PIInterface".

   i. On the taskbar, hold the "Shift" key then right-click the command prompt and select "Run as different user".

   ii. Enter the username "PISCHOOL\svc-PIInterface" and the password "student"

b. Navigate to the directory C:\Program Files (x86)\PIPC\bin
Hint: type "cd %pihome%\bin"

c. Run the command **apisnap PISRV01**. What response do you get?

Step 2 : First, we will create the PI Identity, the "access badge" our PI Interface needs to connect to the Data Archive. Log on to PISRV01, open SMT, and navigate to Security > Identities, Users, & Groups

a. You should be on the "PI Identities" tab. Click the "New…" button  at the top left.

b. In the "Identity" field, enter the name "PI Interfaces & PI Buffers". Click "Create"

Step 3 : Now, we will assign permissions to the PI Identity. Navigate to Security > Database Security.

a. Double-click on the "PIPOINT" table

b. Click “Add” and select the identity you just created, then click “OK”

c. Under “Permissions”, select both “Read” and “Write”, then click “OK”

Step 4 : Finally, we need to associate the service accounts IT created to the PI Identity we just created. Navigate to Security > Mappings & Trusts. You should arrive on the “Mappings” tab.

a. Click the “New Mapping” button 

b. Click on the ellipses  next to the “Windows Account” field. “From this location” should be PISCHOOL.INT. Enter the name svc-PIInterface, and click OK.

c. Click on the ellipses  next to the “PI Identity: field. Select the PI Identity “PI Interfaces & PI Buffers”

d. Click Create

Step 5 : Close PI SMT.

Step 6 : Validate that the new PI Mapping is working

a. From PIINT01, run the command prompt as the user “svc-PIInterface”.

   i. On the taskbar, hold the “Shift” key then right-click the command prompt and select “Run as different user”.

   ii. Enter the username “PISCHOOL\svc-PIInterface” and the password “student”

b. Navigate to the directory C:\Program Files (x86)\PIPC\bin
Hint: type “cd %pihome%\bin”

c. Run the command **apisnap PISRV01**

d. Back on PISRV01, in SMT, navigate to Operation > Network Manager Statistics. This utility shows you all the active connections to the Data Archive

e. Scroll to the bottom of the list and look for a connection with name “snapE”.

   i. What does this connection represent?

   ii. Which user made this connection?

   iii. Which identity was assigned to this connection?

2.11.7 Directed Activity – Configure a new instance of the PI Interface for OPC DA



In this part of the class, you will perform a learning activity to explore the different concepts presented in this chapter or section. You may be invited to watch what the instructor is doing or perform the same steps at the same time. You may play a game or hold a quiz. Your instructor will have directions.

Activity Objectives

Complete step 7 from the PI Interface installation methodology we outlined earlier in this chapter:

> ***Step 7 :*** *Create and configure an instance of the PI Interface*

Approach

Step 1 : From PIINT01, Run the PI ICU

Step 2 : Select the "Create new Interface Instance from .EXE" button  on the upper-left corner of the window.

Note: Every PI Interface installation comes with an example batch file called XXX.bat_new. New PI Interface instances can also be created by loading this example file in the PI ICU using the "Create new Interface Instance from .BAT file" button . This default batch file includes a common configuration, which can speed up the PI Interface configuration process. Choosing the .EXE method allows you to fully configure the PI Interface from scratch.

---

a. Browse to the PI Interface executable, located at *C:\Program Files (x86)\PIPC\Interfaces\OPCInt_ReadOnly,* and select the executable *OPCInt_ReadOnly.exe*

b. Select the host PI Data server "PISRV01"

c. Under the Optional Settings, set a Point Source of "OPC-PIINT01".

As we learned in earlier in this chapter, the combination of Point Source + Interface ID for each interface must be unique. When creating PI Interfaces, AVEVA recommends choosing a unique Point Source. This will facilitate PI Interface management and improve the performance of the PI Interface instances.

d. Click "Add" then "OK".

Step 3 : On the "General" tab

a. Set the Interface ID to 1

b. Click the "Add a scan class" button  and create a scan class with a 5 second frequency.

As we learned earlier in the chapter, the Location4 attribute of a PI Point assigns the point to one of the PI Interface scan classes. The scan frequency of the scan

class determines the rate at which the data is updated. The format of the scan frequency is as follows:

hh:mm:ss.##,

hh:mm:ss.##

where the following rules apply:

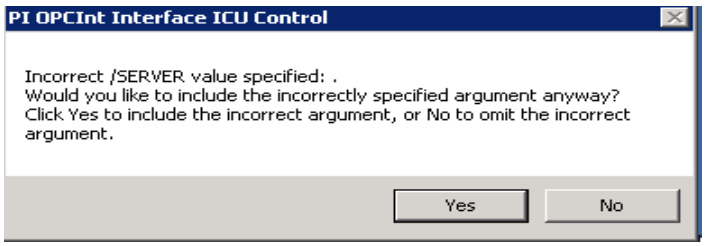
- the time before the comma represents the frequency
- the time after the comma represents an offset from midnight
- hh is hours
- mm is minutes
- ss is seconds
- ## is hundredths of seconds (01 to 99).
- If hh and mm are omitted, the scan period is assumed to be in seconds. For example, a scan frequency of **00:01:00,00:00:05** is equivalent to **60,5**.

The following tables show some example of scan classes and their result:

Scan class	Result
00:00:05	The PI Interface collects data every 5 seconds, and starts collecting immediately after starting up. Example: 12:24:02 12:24:07 12:24:12
00:00:05,00:00:00	The PI Interface collects data every 5 seconds, and starts collecting so that there is a 0 second offset from midnight. Example: 12:24:05 12:24:10 12:24:15
5,0	Same result as example above
01:00:00, 00:30:00	The PI Interface collects data every hour, and starts collecting so that there is a 30 minute offset from midnight. Example: 12:30:00 13:30:00 14:30:00

Step 4 : On the OPCInt tab

a. When you switch to this tab, you will get prompted with the following message:



Since we have not finished our configuration yet, click “No”.

b. Configure this PI Interface to collect data from our OPC Server, **OPCSample.OpcDa20Server.1**. This is the same server we connected to using the PI OPC Client Tool in directed activity 0 (page 36).

i. The “OPC Server Node Name” should be the IP address of the OPC Server node. Since we are connecting to a local OPC Server, we can leave this field as “localhost”

ii. Press the button “List Available Servers”

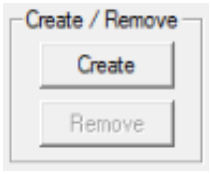
iii. Under the “OPC Server Name” field, select the OPC Server OPCSample.OpcDa20Server.1.

Step 5 : On the Service tab

a. Under “Log on as:” select “[Domain\]UserName”. Enter the following account information:

- UserName: PISCHOOL\svc-PIInterface
- Password: student

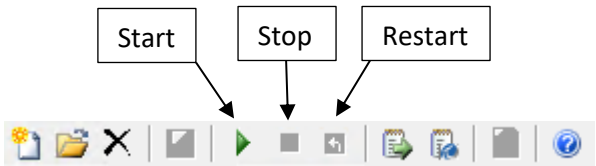
b. Press the “Create” button to create the service



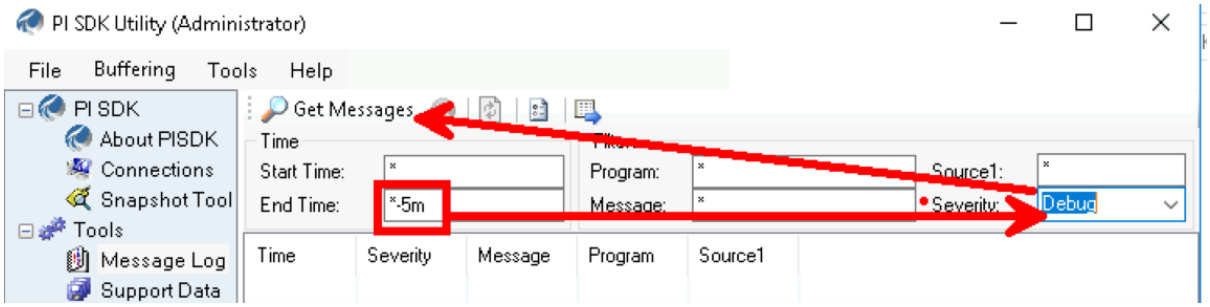
Step 6 : Save your changes by clicking the save button 

Step 7 : Start the service and watch the PI Message Log

a. After creating the service, you should now have access to the start, stop and restart Windows Service buttons at the top of the PI ICU Window.



b. Open PI SDK Utility to view logs. Enter End Time as *-5m and change Severity to Debug and click on Get Messages.



c. In the PI Message Logs, you should see the following messages:

`PIAPI successfully connected to PISRV01.`

This message means that interface is able to connect to PI Data Archive.

`Connected to OPC Server PIINT01:: OPCSample.OpcDa20Server.1 in thread ID XXXX`

This message means that the PI Interface was successfully able to connect to the OPC Server.

`OPC Server current state = RUNNING`

This message means that the OPC Server is in a good state.

`Total Number of points matching pointsource 'OPC-PIINT01' is 0`

This message means that no PI Points have been created with the Point Source of the PI Interface, so no data will be collected. This will soon change when we add PI Points in the upcoming exercises.

You can leave PI SDK open for the next exercise. When you want to view new logs, you just need to click on Get Messages.

2.11.8 Definition of PI Point types for the PI Interface for OPC DA

The last step required for data collection is the creation of PI Points for the PI Interface. As previously discussed, the PI Point configuration is unique to each PI Interface. This is due to the diversity of data sources that PI Interfaces can collect from.

Often, data can be requested in multiple different ways from a single data source. This is the case for OPC DA Servers. PI System administrators may choose to collect data in different ways for different PI Points from the same OPC DA Server. There are four different ways to define PI Points for the PI Interface for OPC DA:

Polled

For polled points, the PI Interface polls the OPC Server at regular intervals defined by the scan class frequency.

Advise

For advise points (referred to as read on change in the OPC Standard), the PI Interface requests that the OPC Server send it a new value whenever the OPC Server itself receives a new value and updates its cache. This way, the PI Interface does not have to constantly poll the OPC Server (less network traffic) and does not collect duplicate values from the OPC Server.


Tip

Often the Advise method of reading data is the most efficient and best performing.

Event (Trigger)

When an event point is created, it is associated with a trigger PI Point on the Data Archive (this trigger can be any PI Point). Whenever the value of the trigger point changes, the Data Archive informs the PI Interface, which requests that the OPC Server read directly from its data source and send it a new value.

Output

Output points read a separate PI Point and ***write*** the value out to the data source (the PI Interface is not being used for data collection in this case). The intention of this feature is not to take over the control system. Customers often use the results they get from input points to perform calculations which are written back to output points. As of version 2.6.3.5, there is a Read-Only version of the PI Interface for OPC DA that prevents the use of output points. It is also possible to disable this feature in the PI interface for OPC DA version earlier than 2.6.3.5

When creating PI Points for the PI Interface for OPC DA, the following rules should be applied:

1. The Location3 PI Point attribute determines the PI Point Type:

Location3	Type
0	Polled or Event
1	Advise
2	Output

2. Location4 determines the scan class
3. Only Advise points can go into scan class 1
4. PI Points of different types **cannot** belong to the same scan class!

2.11.9 Solo Exercise – Create a PI Point for the PI Interface for OPC DA using SMT



This is a solo exercise that is designed to maximize learning in a specific topic area. Your instructor will have instructions and will coach you if you need assistance during the exercise.

Exercise Objectives

Begin step 8 from the PI Interface installation methodology we outlined earlier in this chapter:

> ***Step 8 :*** *Create PI Points for the PI Interface.*

You will create your first PI Point using the Point Builder tool of SMT.

Approach

Step 1 : On PISRV01, open SMT and navigate to the tool Points > Point Builder.

Step 2 : Create your first PI Point, which will store the Bearing Temperature of Pump1. You will need to fill out the following attributes:

Attribute	Value
Name	Pump1.BearingTemp
Descriptor [optional]	
Eng Units [optional] (check the beginning of the chapter)	
Point type	
Point source	
Location1	
Location2	
Location3	
Location4	
Location5	
Instrument Tag	

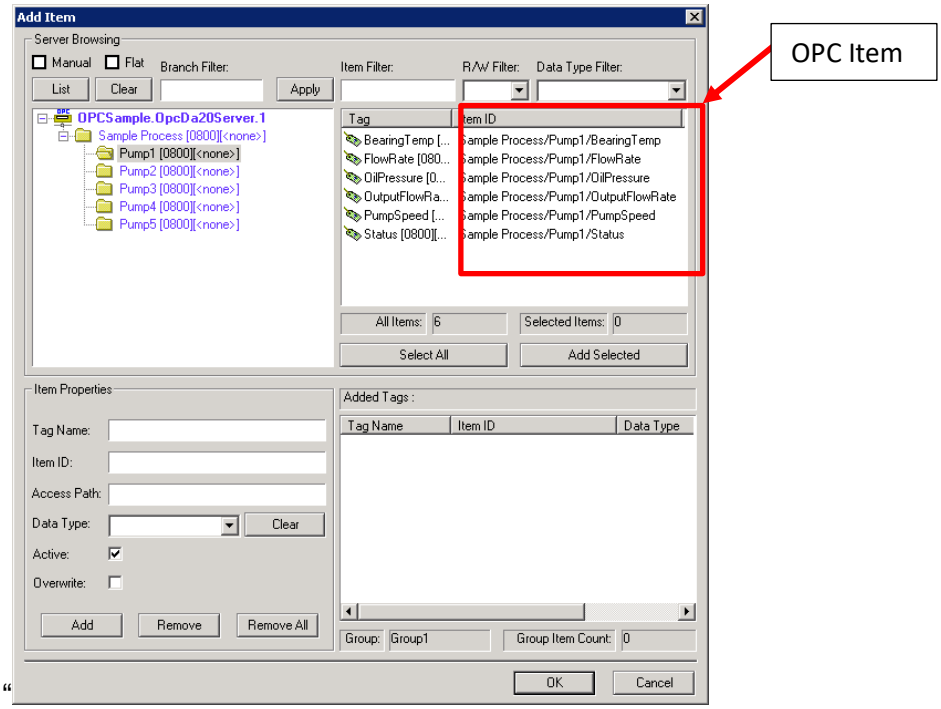
Step 3 : Verify whether your newly created PI Point is getting data. It can take up to 2 minutes for a running PI Interface to detect the new PI Point. While you are waiting, watch the PI Message Log window on PIINT01. When the PI Point is created, you should see the following messages:

tag Pump1.BearingTemp (XX) is added to the Interface

Hint 1: To find more information on how to configure the PI Point attributes, see "PI point configuration for PI Interface for OPC DA" in the PI Interface for OPC DA User Guide. You can access the documentation at these locations:

- Customer Portal: https://my.osisoft.com
- Documentation: https://docs.aveva.com

Hint 2: The Instrument Tag corresponds with the OPC Item ID. These can be viewed using the PI OPC Client Tool, using the same procedure outlined in the directed activity “Using the PI OPC Client Tool”.



“

2.11.10 Directed Activity – Create remaining PI Points for the PI Interface for OPC DA using PI Builder



In this part of the class, you will perform a learning activity to explore the different concepts presented in this chapter or section. You may be invited to watch what the instructor is doing or perform the same steps at the same time. You may play a game or hold a quiz. Your instructor will have directions.

Activity Objectives

Complete step 8 from the PI Interface installation methodology we outlined in this chapter:

> ***Step 8 :*** *Create PI Points for the PI Interface.*

You will create your remaining PI Points using PI Builder

Approach

When creating PI Points for the PI Interface for OPC DA, a functionality of the PI OPC Client Tool can be used to facilitate the PI Point creation process. OPC Items can be added to groups on the PI OPC Client Tool. A csv file can then be generated which is specifically designed to be used to export PI Points using PI Builder.

Part 1 – Generating the CSV file

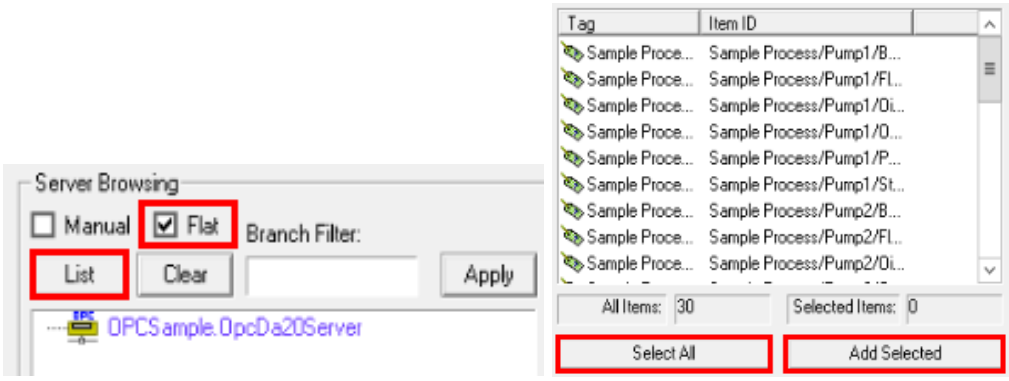
Step 1 : From PIINT01, run the PI OPC Client Tool

Step 2 : Connect to OPCSample.OpcDa20Server.1

Step 3 : Click the “Add Group” button . In the “Add Group” dialog, click “Create”.

Step 4 : Click the “Browse OPC Server, Add Tags” button 

Step 5 : The Add Item window will appear. View the data available on the OPC Server by checking the “Flat” option and press “List” on the left side. This lists all the items available on the OPC Server. On the right side of the window, click the “Select All” button, then “Add Selected”, then “OK”.

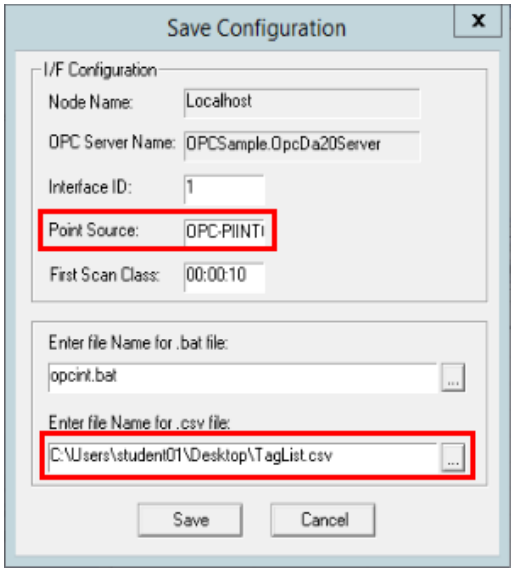


Note: A maximum of 500 OPC Items can be added at once using this method. We will see how to add even more items in bulk later in this section.

Step 6 : You should now have returned to the main PI OPC Client window, with the tags you selected listed under Group1. On the toolbar at the top of the window, select File > Save As

Step 7 : In the Save Configuration window, click on the ellipses [...] button next to the "Enter file Name for .csv file" field and chose the desktop as the file location

Step 8 : Change the Point Source to "OPC-PIINT01", then select "Save"



Part 2 – Creating the PI Points from Point Builder

Step 1 : Open the shortcut on the desktop called "TagsList". Copy-paste the csv file to this folder. This is a shared folder on PISRV01

Step 2 : Open the file with Excel. You should see the following columns in Excel:

Select(x)
Tag
instrumenttag
pointtype
location1
location2
location3
location4
location5
pointsource

The PI Point attributes instrumenttag, pointtype, location2, location5 and pointsource have all been set correctly by the PI OPC Client Tool according to the OPC Item that was added to the group. We now only have to make minor modifications to this spreadsheet before publishing the PI Points to the Data Archive.

Step 3 : We want all of our PI Points to be Advise points in scan class 1. Change "location3" to 1 and "location4" to 1 for all PI Points

Step 4 : Change the "Tag" column to give the PI Points appropriate names.

Hint: You can use the following procedure:

i. Copy-paste the "instrumenttag" column into the "Tag" column.

ii. Select the "tag" column.

iii. Use the shortcut Ctrl+H to open the "Find and Replace" dialog.

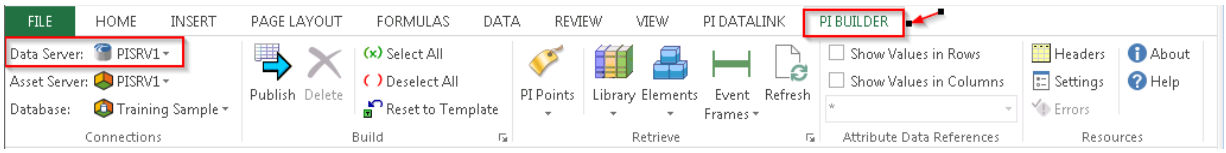
iv. Replace the slash characters (/) with periods (.).

v. Replace the "Sample Process." string with an empty field.

This should give you the naming convention **PumpX.DataName**

Step 5 : The Pump1.BearingTemp PI Point was already created. Therefore, simply remove the "x" in the "Select(x)" column, since we will not be publishing this PI Point.

Step 6 : Browse to the PI Builder tab in the Excel ribbon. . Notice we are connected to the default Data Archive "PISRV01"





Step 7 : Select the Publish Button

Step 8 : Select edit Mode "Create Only" and select "OK". Verify we see the following message at the bottom of the publish window:

The requested action is complete.

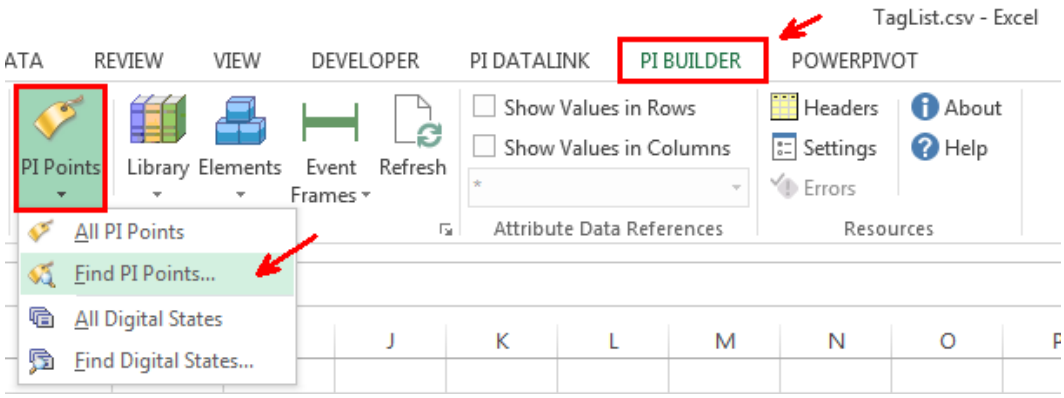
Step 9 : In SMT > Data > Current Values, verify whether your new PI Points are getting data. **Hint:** you can search by point source "OPC-PIINT01"

Part 3 – Managing PI Points using PI Builder

PI Builder can be used to create, edit and delete PI Points. Now that our pump PI Points have been created, let's edit them to make them more user-friendly.

Step 10 : Open a new Excel workbook. Browse to the PI Builder tab in the Excel ribbon.

Step 11 : On the PI Builder tab, select PI Points > Find PI Points



Step 12 : Search for all of your Pump PI Points, select them and his "OK"

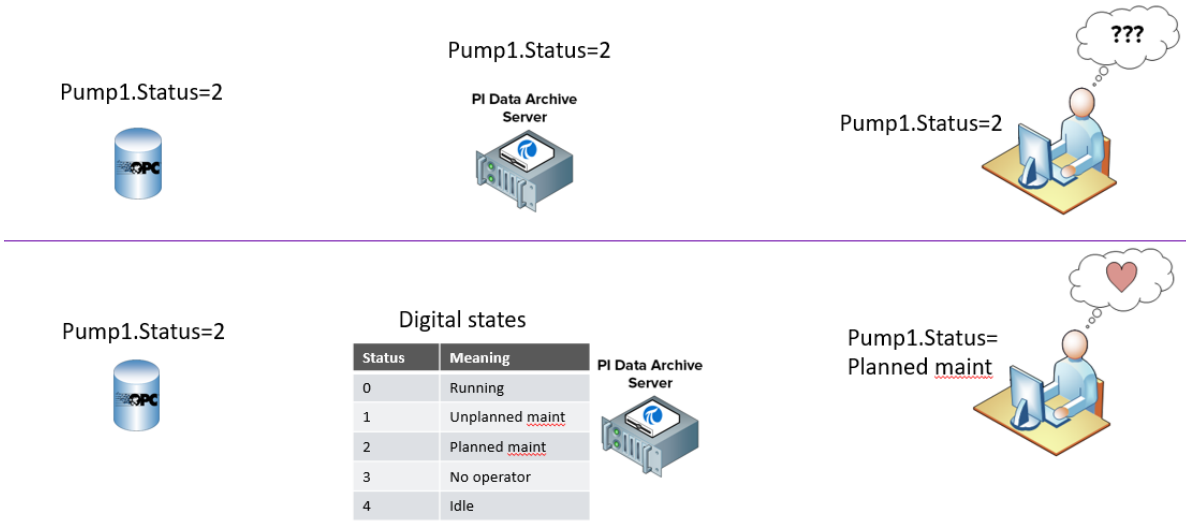
Step 13 : On the "Select Object Types and Column Headers", select the "Required Columns", the "Description" and "engunits" columns, then "OK"

Step 14 : Edit the description and engunits column for each of the PI Points. **Hint:** Use copy-paste and find-replace (Ctrl+H) to speed up the process!

Step 15 : Publish your changes. This time, select edit mode "Edit Only".

2.12 Digital tags

Sometimes the value that we receive from the data source is analog value or has another meaning. For example, when we receive 0 it means that the device is running and when we receive 1 it means device is under maintenance. In such cases, digital states can be utilized to make the values more user-friendly.



2.12.1 Directed activity – Create Digital States and assign tags to it



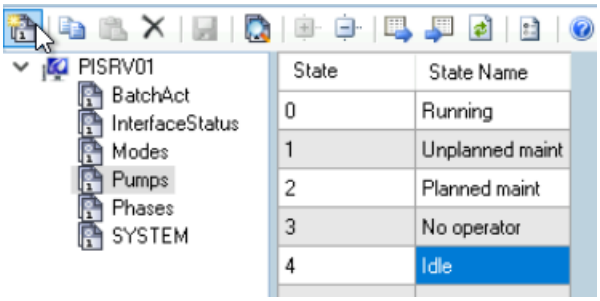
In this part of the class, you will perform a learning activity to explore the different concepts presented in this chapter or section. You may be invited to watch what the instructor is doing or perform the same steps at the same time. You may play a game or hold a quiz. Your instructor will have directions.

Activity Objectives

We have created 5 Status tags. The meaning of these are as shown in the table above. Create a digital state and assign these tags to it.

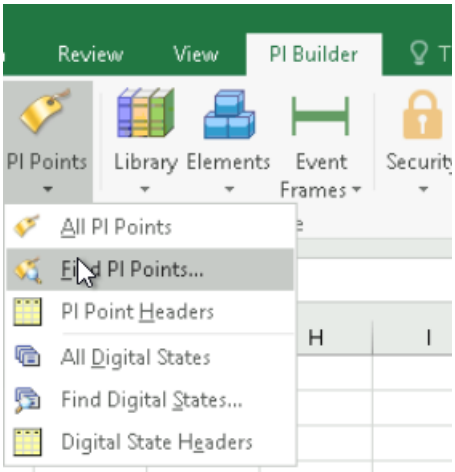
Approach

Step 1 : On PI System Management Tools, go to Points > Digital States. Click on the Add button:



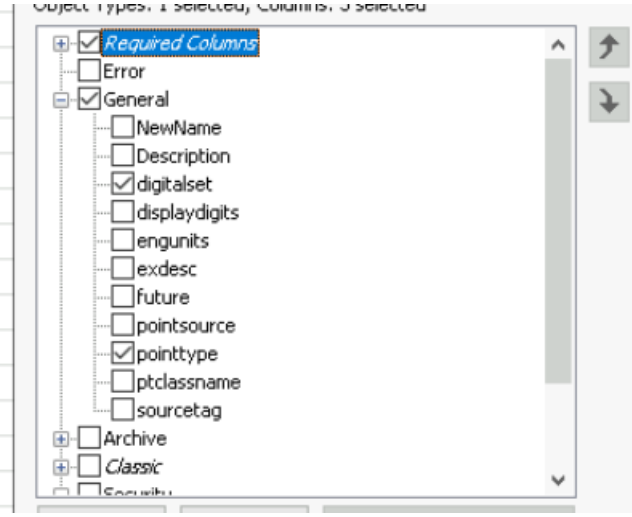
Step 2 : Create a new set called Pumps and assign values as shown above. Save the set. You do not have to type anything into State column, it will be populated automatically.

Step 3 : Open Excel, go to PI Builder > PI Points and click on Find PI Points.



Step 4 : Search for Pump status tags, select all of them and click OK. How can you find all of them using a single query?

Step 5 : We would like to change the point type and assign a new set to these tags. Therefore, select only these columns and click OK.



Step 6 : Change the cells as shown below and publish your changes using “Edit only” mode.

	A	B	C	D	E
1	**Selected(x)**	**Name**	**ObjectType**	**digitalset**	**pointtype**
2	x	Pump1.Status	PIPoint	Pumps	Digital
3	x	Pump2.Status	PIPoint	Pumps	Digital
4	x	Pump3.Status	PIPoint	Pumps	Digital
5	x	Pump4.Status	PIPoint	Pumps	Digital
6	x	Pump5.Status	PIPoint	Pumps	Digital

Step 7 : Check the values of your status tags using Data>Current Values.

2.13 Troubleshooting Exercise



The following group exercise is intended to reinforce the concepts we have learned so far. The instructor will act as an end user with an issue and you will resolve their issue as a group.

2.14 Configuring a reliable PI Interface

In the previous section, we have gone through all the steps required to collect data from an OPC DA Server. We now have process data from 5 pumps in our Data Archive, which PI System users can view in real-time, and use in their analyses.

However, the data from this PI Interface is not yet reliable. Multiple issues can arise which would lead to PI System users losing access to the data

2.15 Group Questions – Preventing data loss

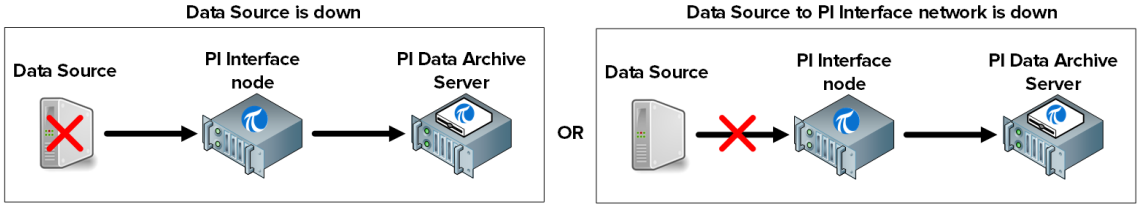


The following questions are intended to reinforce key information, or to discover a new insight. Your instructor may choose to have you try to answer the questions on your own or have the group answer them together aloud.

Questions

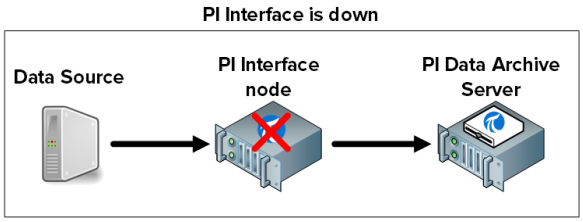
For each of the scenarios below, determine whether it is possible to avoid data loss, and determine which AVEVA functionality can be used to prepare for each.

A:



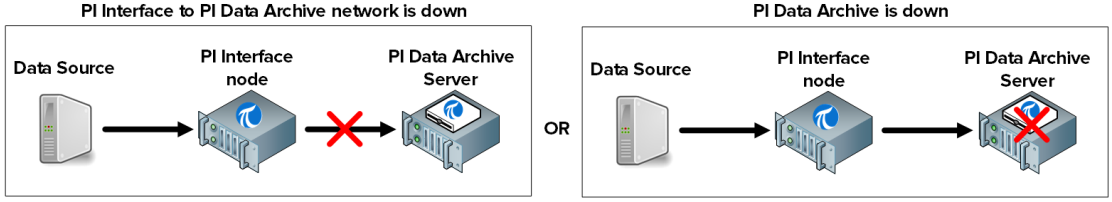
Is it possible to prevent data loss: ☐ Yes ☐ No Which steps can be taken to prepare for this scenario:_____

B:



Is it possible to prevent data loss: ☐ Yes ☐ No Which steps can be taken to prepare for this scenario:_____

C:

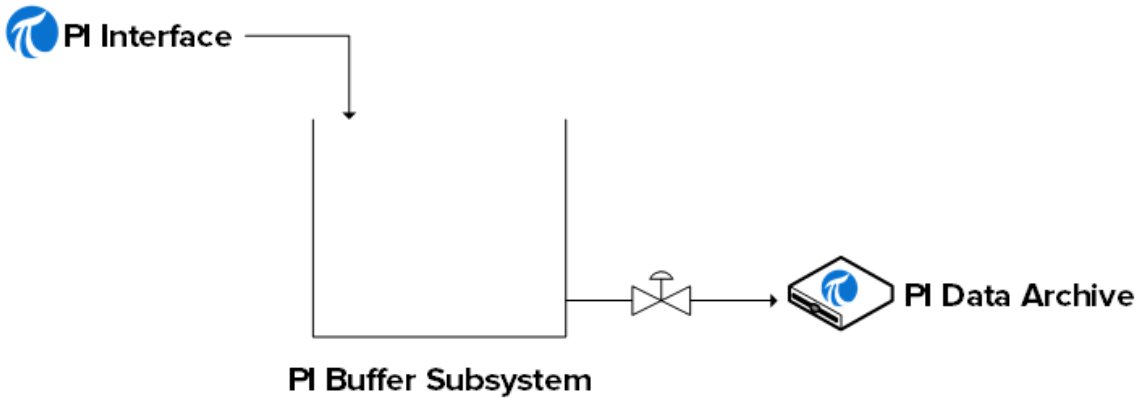


Is it possible to prevent data loss: ☐ Yes ☐ No Which steps can be taken to prepare for this scenario:_____

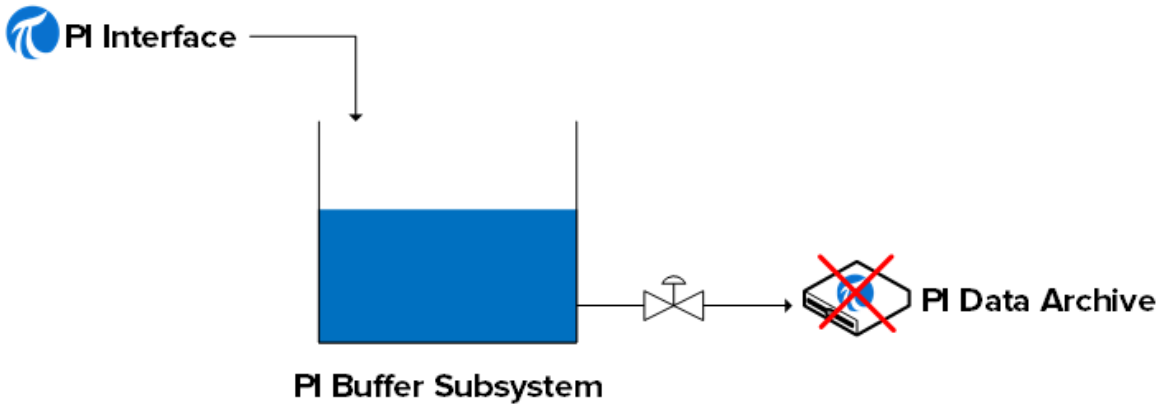
2.16 Defining the PI Buffer Subsystem

2.16.1 What is the PI Buffer Subsystem?

The PI Buffer subsystem is an AVEVA application installed with every PI Interface. Once configured, it buffers data on the PI Interface node. A good analogy for the PI Buffer Subsystem is a reservoir.



Under normal operation, when the valve to the Data Archive is open, data simply passes through the reservoir. When the valve closes (i.e. The Data Archive is down, or the network is down), as the PI Interface continues to collect data, this data will start accumulating in the reservoir.



When the valve opens again (the Data Archive, or network, are restored), the data will flush out of the reservoir and be sent to the Data Archive.

Note: There is another, older AVEVA buffering service called the API Buffer Server. The PI Buffer Subsystem is the best option for most environments. The API Buffer Server should only be used if (1) the PI Server receiving the buffered data is older than version 3.4.375 and (2) the PI Interface runs on a non-Windows platform.

2.16.2 How does the PI Buffer Subsystem work?

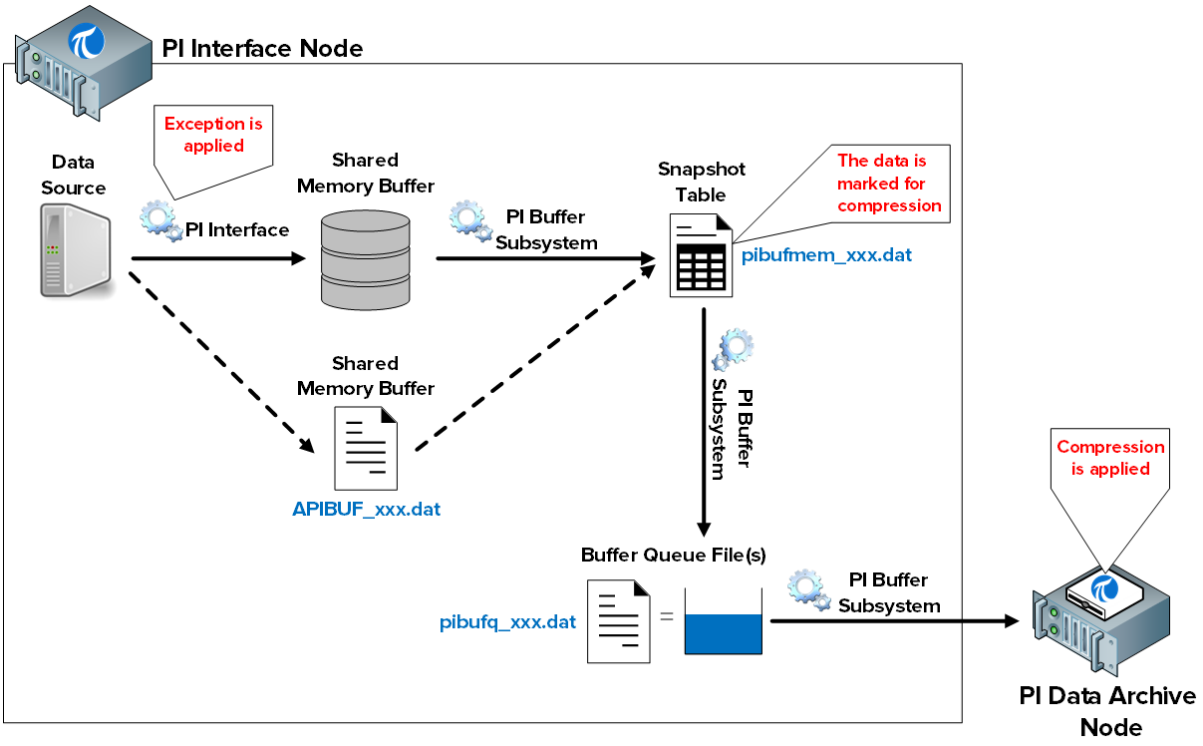
The PI Buffer Subsystem runs as a Windows Service.

The PI Buffer Subsystem does not just buffer data from a PI Interface. It can actually buffer data from any application (AVEVA or custom-built) which is writing data to a Data Archive.

When the PI Buffer Subsystem has been configured, instead of writing data directly to the Data Archive, PI API applications (such as PI Interfaces) write the data to a “shared memory buffer”.

The PI Buffer Subsystem performs the following steps:

1) Reads data from the shared memory buffer and transfers it to its “snapshot table”
2) From the snapshot table, marks the data for compression
3) Writes the data to a memory-mapped buffer queue file
4) Reads data from the buffer queue and sends it to the Data Archive



Note: Exception and Compression are mechanisms by which data is filtered, so that only meaningful data is kept in the Data Archive. The PI Interface is responsible for Exception. The PI Buffer Subsystem marks events as either **Snapshot Only** (i.e. get rid of this value when a new one comes in) or **To Be Archived** (store this value). The Data Archive then treats the data accordingly. We will continue our discussion of the compression algorithm in the following chapter.

The files involved in this process are:

1) Shared memory buffer: The location in memory where PI Interfaces write data. When this location is full, data is written to a file on disk called APIBUF_<Data Archive name>.dat
2) Snapshot table (pibufmem_<GUID>.dat): This table holds the most recent value received for all buffered PI Points.
3) Buffer queue file (pibufq_<GUID>.dat): This is the file that acts as our “tank”. The default size of this file is 32 MB. When the file becomes full, a second file is created. When this file becomes fill, a third file is created. This will continue until the PI Interface node runs out of disk space.

Note: PI SDK and AFSDK applications write data directly into the PI Buffer subsystem’s snapshot table, and so the first step above is skipped.

2.16.3 Directed Activity – Configure Buffering



In this part of the class you will perform a learning activity to explore the different concepts presented in this chapter or section. You may be invited to watch what the instructor is doing or perform the same steps at the same time. You may play a game or hold a quiz. Your instructor will have directions.

Activity Objectives

Complete step 9 from the PI Interface installation methodology we outlined in this chapter:

Step 9 : *Configure buffering with the PI Buffer Subsystem*

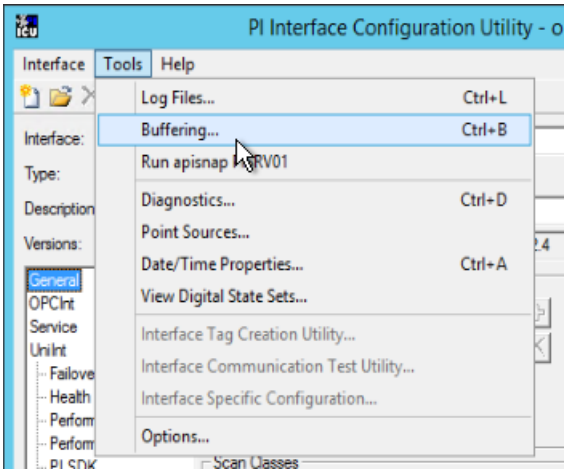
AVEVA recommends using designated Windows Service Accounts to run PI System services that communicate over the network. Before starting, you made a request to your IT department to create a Service Account:

- PISCHOOL\svc-PIBuffer (password: student)

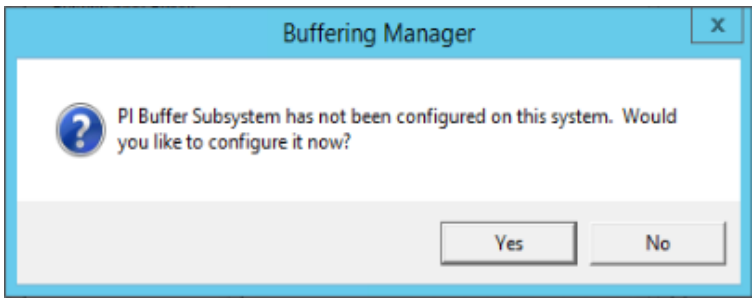
Approach

<u>Part 1 – Configure the PI Buffer Subsystem</u>

Step 1 : From PIINT01, run the PI ICU. Navigate to Tools > Buffering.

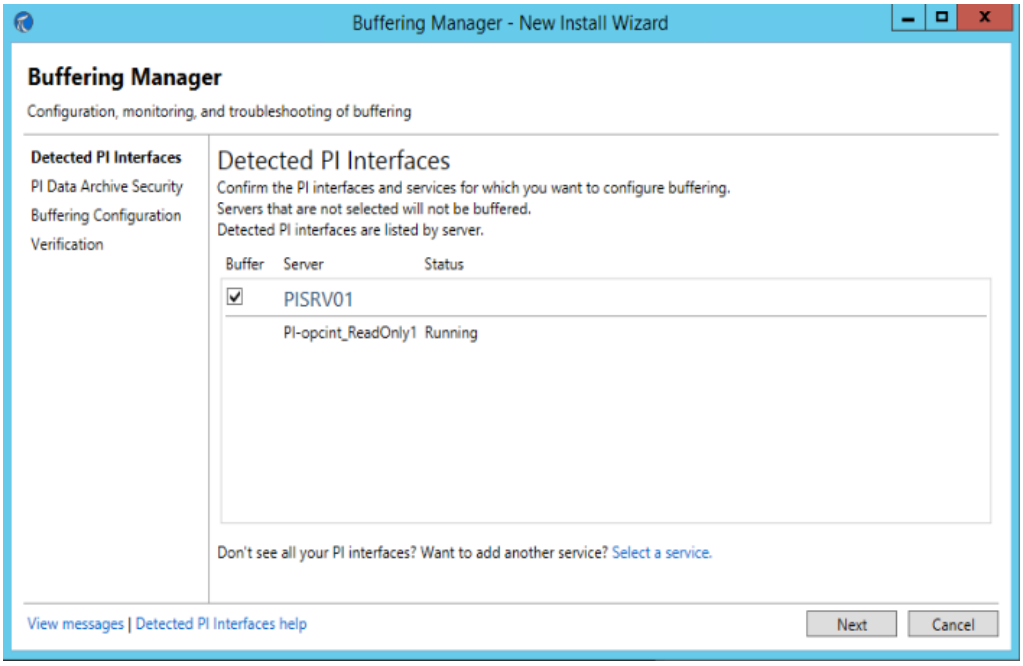


Step 2 : Select “Yes” to continue with the PI buffer subsystem configuration wizard



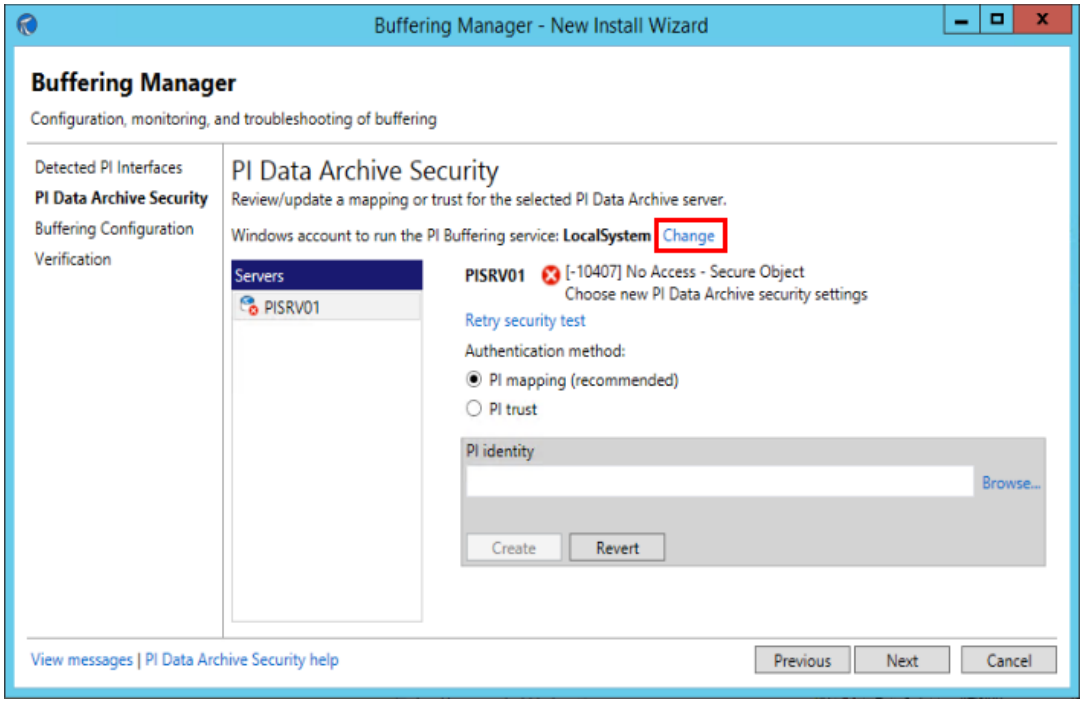
Step 3 : Select “Continue with configuration”.

Step 4 : You should see the Data Archive “PISRV01” and the PI Interface name we configured earlier. Check the box and click “Next”.

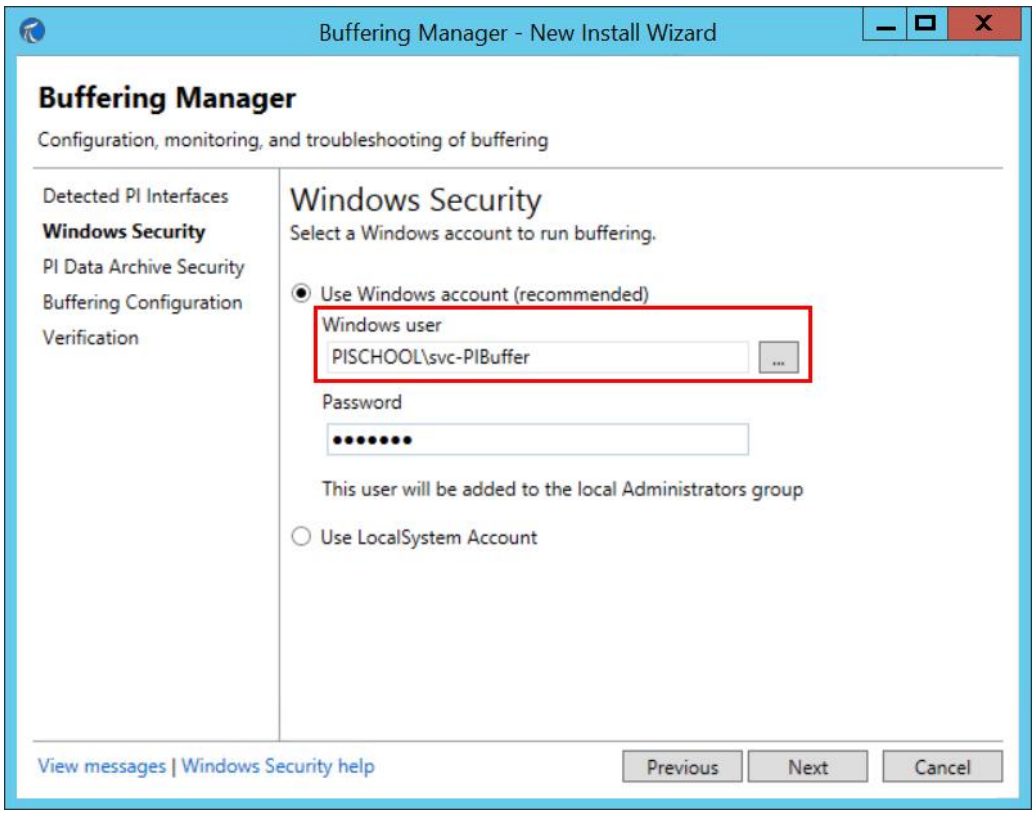


Step 5 : This step ensures that the PI Buffer Subsystem will the granted the correct security on the Data Archive.

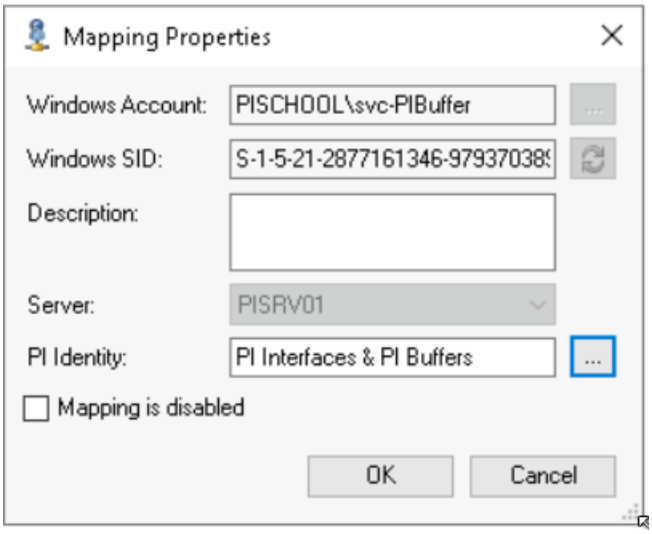
i. The default service account for the PI Buffer Subsystem is NT Service\pibufss. This account has more privilege on the local computer than is needed. Before proceeding, we will change this service account to our dedicated domain account. Click on “Change” next to “NT Service\pibufss”



ii. Select “Use Windows account”. Enter the account “PISCHOOL\svc-PIBuffer” with password “student”. Click on “Next”.



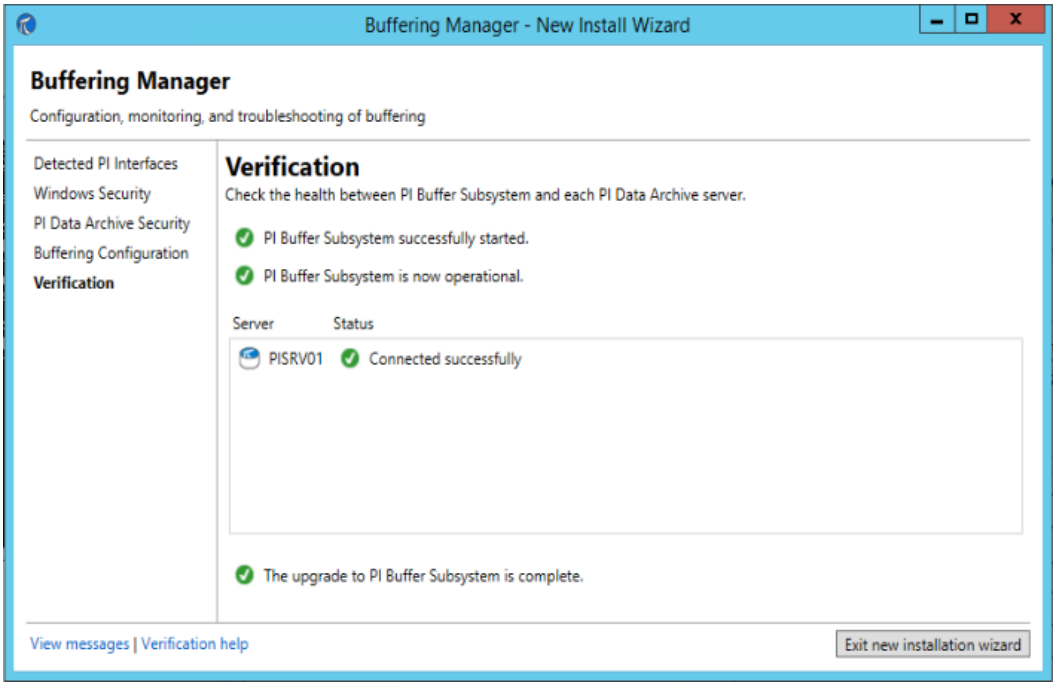
iii. When you return to the security test window, the test will fail again. Why? Because we did not create a PI Mapping. This means PI Data Archive is not expecting the user PISCHOOL\svc-PIBuffer. We will create the PI Mapping we need to authenticate on the Data Archive. Go to PISRV01 and create a new PI Mapping for PISCHOOL\svc-PIBuffer



iv. Once you return to the Data Archive Security window, click “Retry”. This time it should work because PI Data Archive is expecting this user.

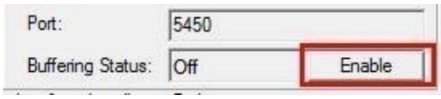
Step 6 : You can now select the Buffer Queues location. For this environment **set the location to E:\OSIsoft\Buffering**. If possible, it is strongly recommended to separate the buffer queue from the OS drive in order to avoid failure on the interface node by filling the main drive.

Step 7 : The last window will run a verification on the status of the PI Buffer Subsystem. If there is no errors, you can **exit the installation wizard**.



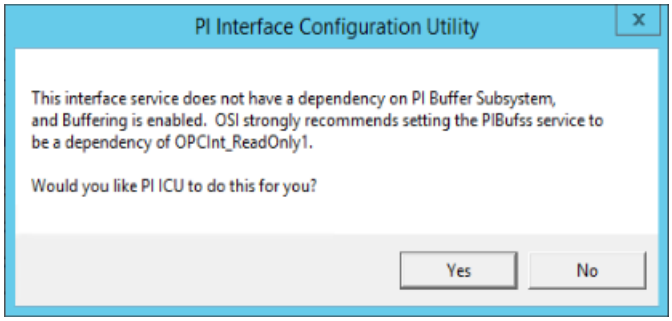
Step 8 : Once the installation wizard completed and closed the Buffering Manager window will open showing the status and statistics of the PI Buffer Subsystem. To re-open this window from the PI ICU, navigate to Tools > Buffering.

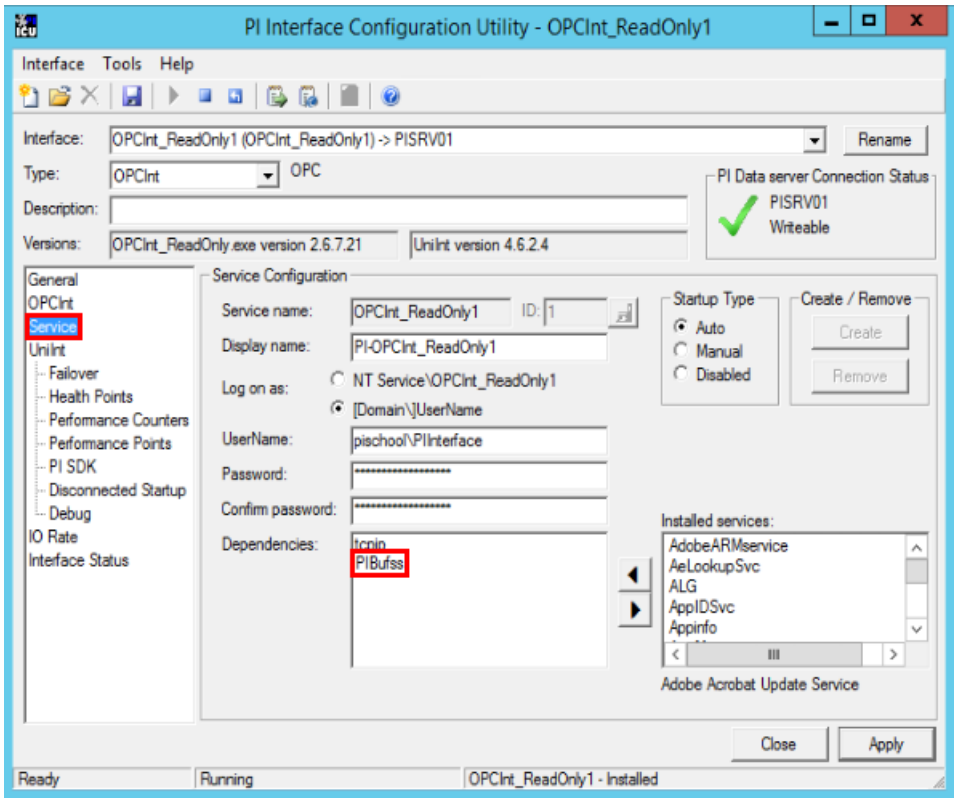
Step 9 : Interface should automatically show that the buffering status is on. If not, click on the butten below to enable buffering from the PI ICU general tab.



To buffer data from a PI Interface, the PI Buffer Subsystem must start before all the PI Interfaces on the machine. Since both are run using Windows Services, it is possible to configure a dependency on the PI Buffer Subsystem.

You can verify the configuration by navigating to the PI ICU > Service tab and verifying the “Dependencies” field. The PI ICU will automatically detect if a dependency on PIBufss is missing. Notice PIBufss is added to the interface service dependencies when we select “yes.”





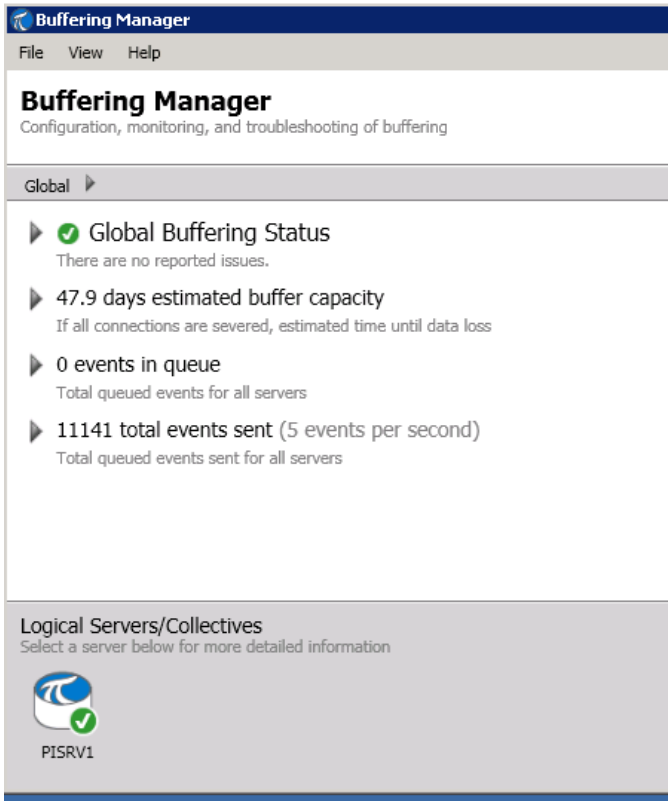
Part 2 – Validate buffering

A very important troubleshooting tool is how to verify whether data is truly buffering on our PI Interface node. Since version 4.3 of the PI Buffer Subsystem, this task has been made much easier with the introduction of the “Buffering Manager” GUI. Since this tool is not available with older versions, we will demonstrate how to verify the buffering status in both scenarios

Method 1: Buffering Manager

Step 1 : From PIINT01, Run the PI ICU. From the top of the window, select Tools > Buffering

Step 2 : In the Buffering Manager, the buffering statistics update in real-time to show the Global Buffering Status, the estimated buffer capacity, the events in queue, and the total events sent in real time:



To confirm that events are flowing through the buffer, make sure that the "total events sent" is increasing. The buffering manager will also report issues such as low disk space, important error messages, etc.

Method 2: Command line utility pibufss (optional)

Step 1 : From PIINT01, run a command prompt, navigate to the directory C:\Program Files\PIPC\bin
Hint: type "cd %pihome%\bin"

Step 2 : Run the command **pibufss -cfg**

This command will show you the overall buffering status.

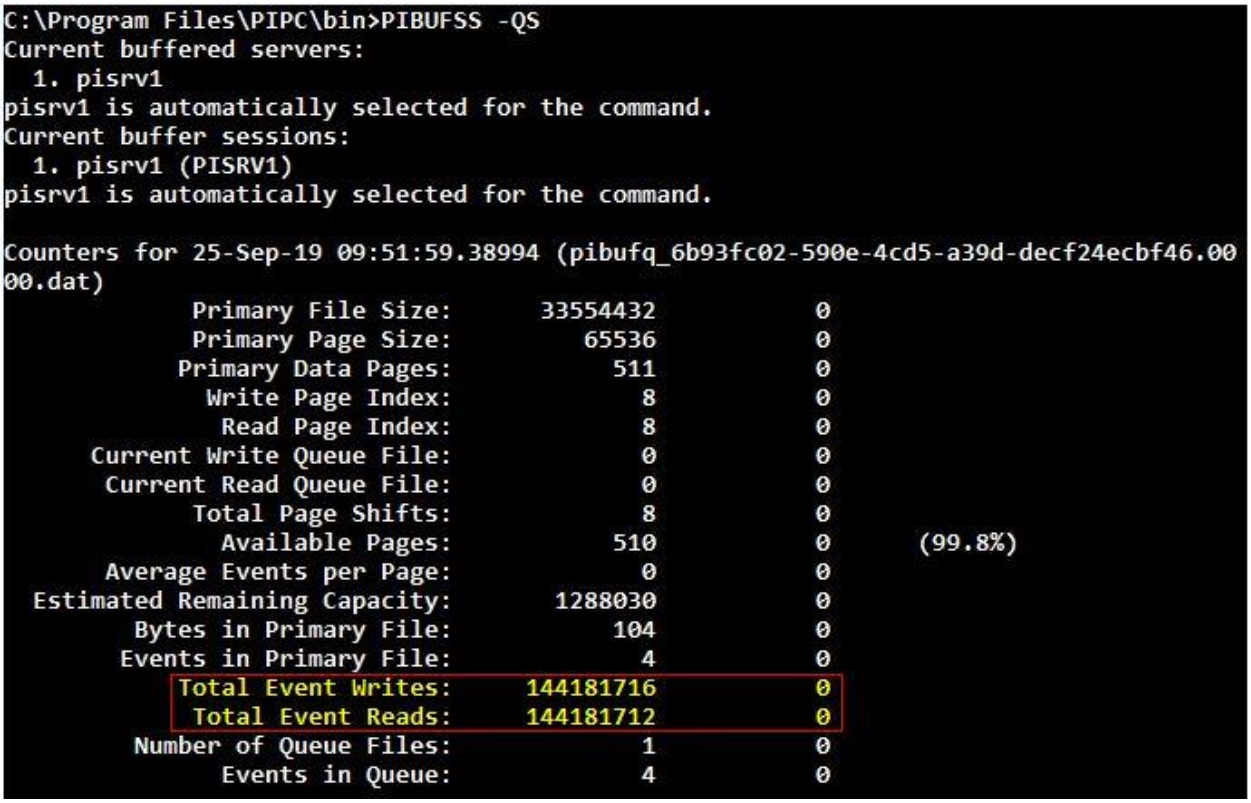
```
C:\Program Files\PIPC\bin>pibufss -cfg

*** Configuration:
Buffering: On (API data buffered)
Loaded physical server global parameters: queuePath=C:\ProgramData\OSIsoft\Buffering
authenticationOptions=SSPI;TRUST

*** Buffer Sessions:
  1 non-HA server, name: pisrv1, session count: 1
    1 [pisrv1] state: SendingData, successful connections: 6
      PI identities: PIBuffers, auth type: SSPI
      firstcon: 16-Jul-19 02:17:53, lastreg: 25-Sep-19 09:14:02, regid: 4
      total events sent: 144184296, snapshot posts: 17764427, queued events: 5
```

Step 3 : Run the command **pibufss -qs**

This command will show you the statistics of the buffer queue file. A healthy buffer queue will have reads and writes from this file.



Step 4 : To stop the statistics, press Ctrl+C

2.16.4 Solo Exercise – Test the PI Buffer Subsystem (Optional)



This solo or group activity is designed to maximize learning in a specific topic area. Your instructor will have instructions and will coach you if you need assistance during the activity.

Exercise Objectives

- See the PI Buffer Subsystem in action

Problem Description

Now that you have configured the PI Buffer Subsystem, you will test it and see its mechanism in action

Approach

Step 1 : From PISRV01, create a PI Vision display showing the data from one of your Pumps for the last 15 minutes. Rename your display “Pump Data”.

Step 2 : From PIINT01, open the Buffering Manager and validate that the PI Buffer Subsystem is healthy. (You can also use the **pibufss -qs** command from the previous directed activity)

Step 3 : In this step, we will simulate a network disruption. On PIINT01, open command prompt as administrator and run the command **pibufss -bc stop** (this manually tells the PI Buffer Subsystem to stop sending data).

```
C:\Program Files\PIPC\bin>pibufss -bc stop
Current buffered servers:
  1. PISRV01
PISRV01 is automatically selected for the command.
Current buffer sessions:
  1. PISRV01 (PISRV01)
PISRV01 is automatically selected for the command.

Control command "stop" successfully initiated on session PISRV01
Please check the PI Message Log for errors.
```

Step 4 : Watch the “events in queue” grow in the Buffering Manager or the first command prompt dialog you opened.

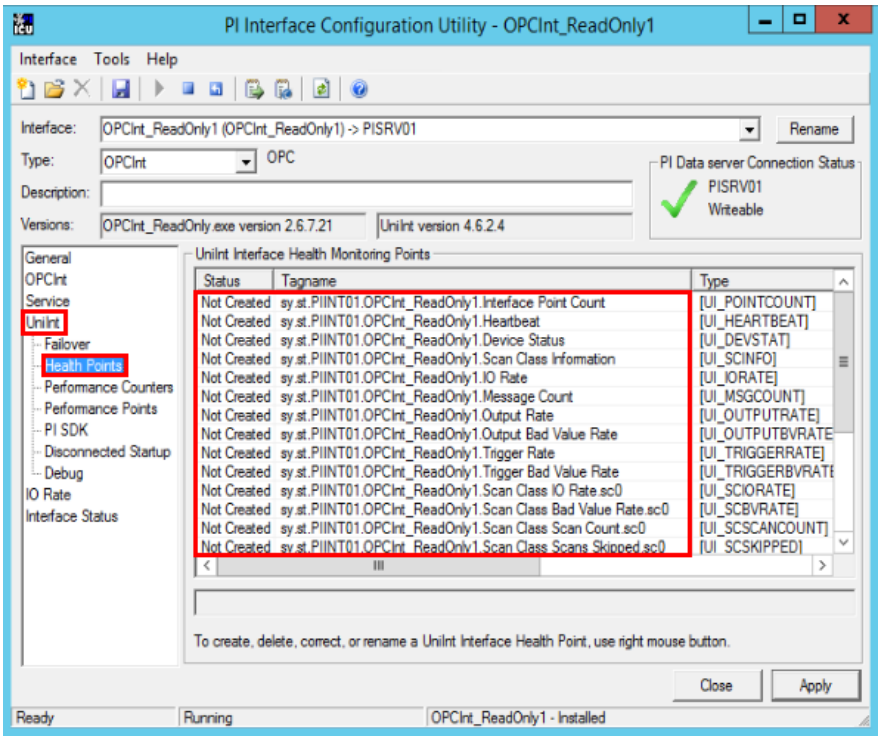
Step 5 : Watch what happens in your PI Vision display.

Step 6 : Once a few minutes have gone by, run the command **pibufss –bc start** to re-establish the connection.

Step 7 : Check the Buffering Manager and your PI Vision display once again.

2.17 Monitor the health of the PI Interface

UniInt Health Points are PI Points that collect information about the health of the interface. They are created from the PI ICU, in the UniInt > Health Points section:



AVEVA recommends that, at minimum, the following UniInt Health Points be created:

1. **Heartbeat**: This PI Point indicates whether or not the interface is running. The heartbeat point is updated continuously unless the interface is shut down or in a deadlock situation. As long as the interface is running, the value of the point cycles incrementally from 1 to 15. The heartbeat point does not indicate whether the interface is connected to or collecting data from a data source.
2. **Device Status**: This PI Point contains information about communication between the interface and the data source. During normal operation, it contains the value GOOD, indicating that the interface is communicating properly with the data source. Otherwise, the point contains a string indicating status, formatted as follows:

   Status code | Description | Interface-specific text.

   Example: 95 | Device(s) in error

   This device status means that the PI Interface cannot communicate with the data source.
3. **IO Rate**: This PI Point keeps a count of all point values (inputs, outputs, triggered inputs) being sent to the Data Archive. If the value stops updating, the interface has stopped collecting data.
4. **Scan Class Scans Skipped**: This PI Point counts, for a specific scan class, “skipped scans”, i.e the number of scans that were not performed before the scan time elapsed and the next scheduled scan was executed, for a defined reporting period (8 hours by default)

2.17.1 Directed Activity – Configure UniInt features: Health Points



In this part of the class, you will perform a learning activity to explore the different concepts presented in this chapter or section. You may be invited to watch what the instructor is doing or perform the same steps at the same time. You may play a game or hold a quiz. Your instructor will have directions.

Activity Objectives

Complete steps 10 from the PI Interface installation methodology outlined in this chapter.

> ***Step 10 :*** *Create PI Interface Health Points to monitor the health of the PI Interface*

Approach

Step 1 : On PIINT01, run PI ICU and browse to UniInt > Health Points

Step 2 : Right-click on the following Health Points and select “Create”:

a. Heartbeat
b. Device Status
c. IORate
d. Scan Class Scans Skipped.sc1

Step 3 : On PISRV01, using SMT, verify whether these Health Points are receiving data

Note: We will continue our discussion of UniInt Health Points in the chapter “Monitoring a PI System”.

2.18 Troubleshooting Exercise



The following group exercise is intended to reinforce the concepts we have learned so far. The instructor will act as an end user with an issue, and you will resolve their issue as a group.

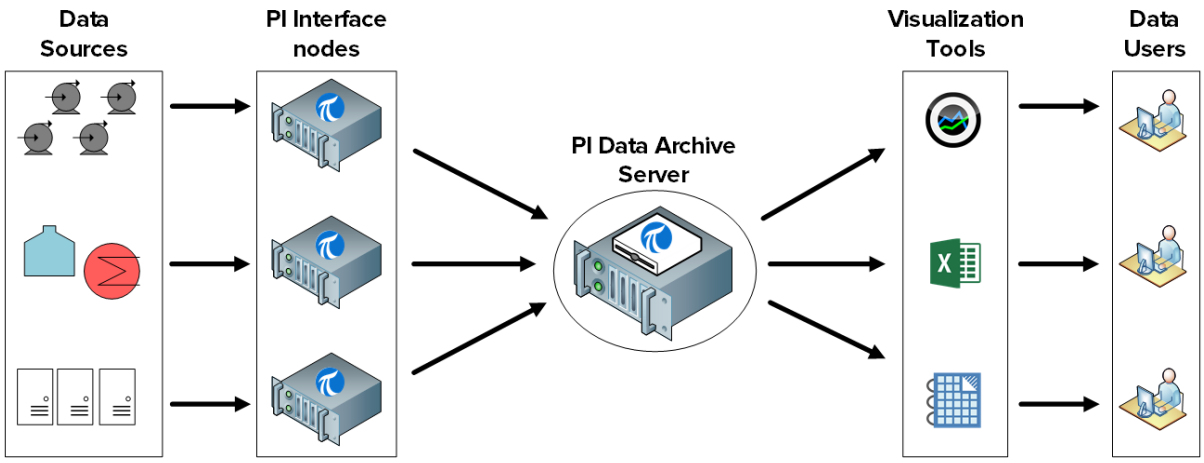
3. Data Archive Management

Objectives

- Describe the components of the Data Archive
- Describe the functions of the basic Data Archive subsystems
- Describe data flow through the Data Archive
- Inspect snapshot, event queue and archive statistics
- Describe exception and compression
- Define a strategy for exception and compression
- Identify the directory structure of the PI folder
- Start and stop the Data Archive
- Describe best practices for archive file size and location
- Change the location and size of archive files
- Describe PI System backup methodology
- Configure a local backup of the PI System
- Describe how to restore a Data Archive from a backup

3.1 Define the role of the Data Archive

In Chapter 1, we learned that the Data Archive is the PI System component responsible for storing time series data, organized into individual streams called PI Points. It receives the data for these PI Points from PI Interfaces, and allows users to access PI Point data through the use of visualization tools like PI Vision.



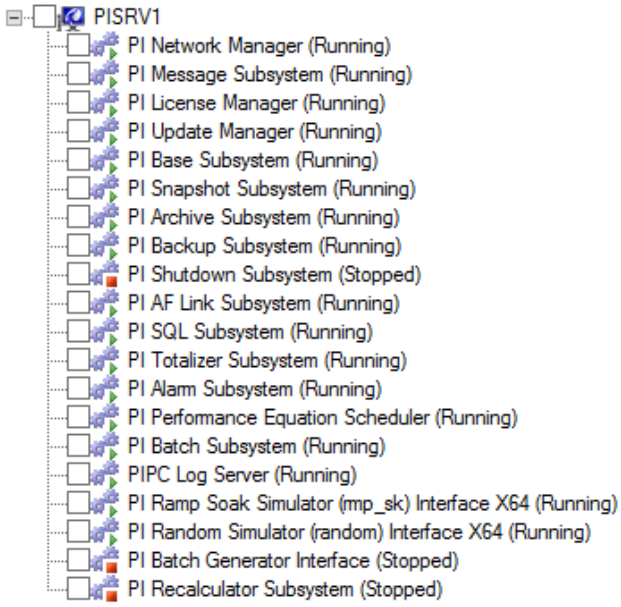
The Data Archive has multiple other roles, including security, licensing, and backup management.

3.2 Describe the Data Archive subsystems

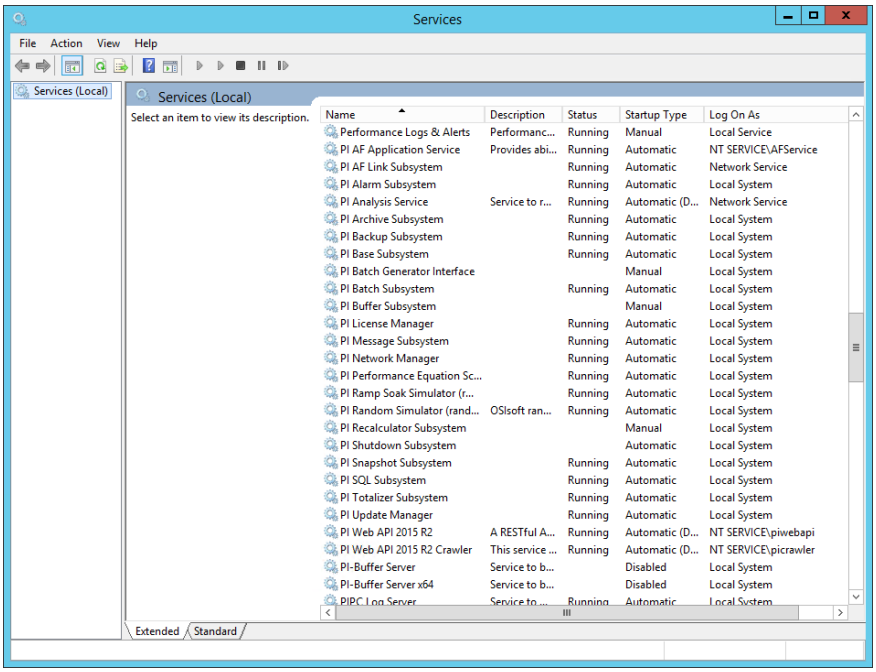
The Data Archive is composed of multiple “subsystems” which manage different tasks. These subsystems are Windows Services.

There are two ways of verifying the status of Data Archive subsystems on PISRV01:

- Using System Management Tools:
    a. Run SMT
    b. Navigate to Operation > PI Services



- Using the services snap-in
    a. Run the application services.msc
    b. Look for Windows Services that start with PI



3.2.1 Group Questions – Identify the role of Data Archive subsystems



The following questions are intended to reinforce key information, or to discover a new insight. Your instructor may choose to have you try to answer the questions on your own or have the group answer them together aloud.

Question

Associate the following Data Archive subsystems with their role, listed in the table below.

PI Network Manager: _____

PI Message Subsystem: _____

PI License Manager: _____

PI Update Manager: _____

PI Base Subsystem: _____

PI Snapshot Subsystem: _____

PI Archive Subsystem: _____

PI Backup Subsystem: _____

1	Stores and serves the data after it comes out of the snapshot subsystem. Data consists of multiple time-stamped measurements for each data point. Values represent on/off, pressures, flows, temperatures, set points, and so on.
2	Maintains the Point Database, Digital State Table, and configuration databases for authentication. Hosts the PI Module Database.
3	Maintains license information for Data Archive and all connected applications.
4	Records status and error messages for Data Archive in a log file.
5	Manages communication between Data Archive subsystems, interfaces and client applications. Also validates clients at time of connection. Clients may be standard products such as PI ProcessBook, or they may be custom PI API or PI SDK programs.
6	Manages backups of the Data Archive
7	Stores the most recent event for each point, applies compression, sends data to the event queue, serves snapshot events, and sends updates for client applications to PI Update Manager.
8	Queues notifications of changes in data values, point attributes, modules, and so on to any interface or client application that is signed up for notification.

3.3 Data Flow through the Data Archive

As we learned in the previous group activity, the PI Snapshot Subsystem and the PI Archive Subsystem are the two services involved in archiving PI System data.

In the previous chapter, we learned how data flows through the PI Interface node when the PI Buffer subsystem is configured. The PI Buffer Subsystem is actually very similar to the PI Snapshot Subsystem and thus performs very similar tasks.

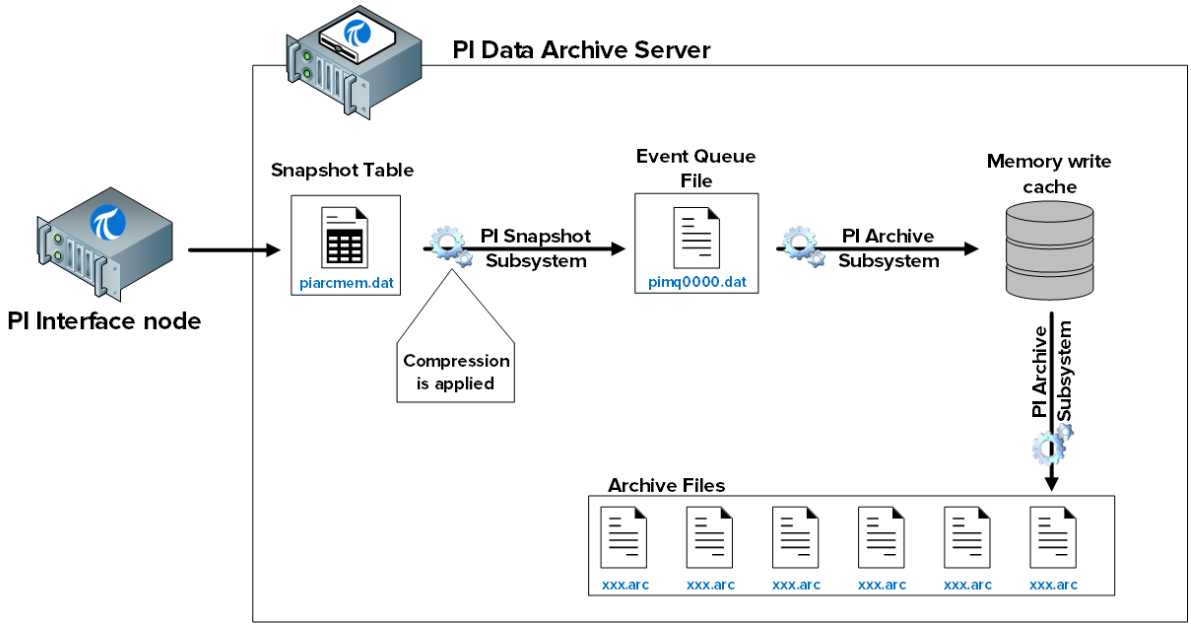
Whenever new data is sent to the Data Archive, the PI Snapshot Subsystem:

1) Reads data from the snapshot table on the Data Archive
2) Applies compression
3) Writes the data to the event queue

At this point, the PI Archive subsystem takes over. It:

1) Reads data from the event queue
2) Writes the data into a “write cache” in memory
3) Periodically, it will write the data from the write cache onto disk, in the archive files.

When PI System visualization tools (e.g., PI Vision) request “snapshot” data, the data they receive comes directly from the snapshot table, before compression is applied.



The files involved in this process are:

1) Snapshot table (piarcmem.dat): This table holds the most recent value received for all PI Points.
2) Event queue (pimq0000.dat): This file is very similar to the buffer queue file on the PI Interface node, discussed in the previous chapter. During normal operation, it simply acts as a tank through which data flows from the PI Snapshot Subsystem to the PI

Archive Subsystem. However, if there is a problem with the PI Archive subsystem (e.g. it is too busy responding to other requests), data will start accumulating in this tank.

3) The memory write cache: This cache, which is stored in memory rather than the hard drive, was designed to minimize the number of writes to disk, thus improving performance. By default, the PI Archive subsystem flushes the cache every 5 minutes defined by the tuning parameter ‘Archive_SecondsBetweenFlush’.

4) Archive files (xxx.arc): These are the files on disk where the archive data for each PI Point is stored. We will continue our discussion of archive file management in later in this chapter.

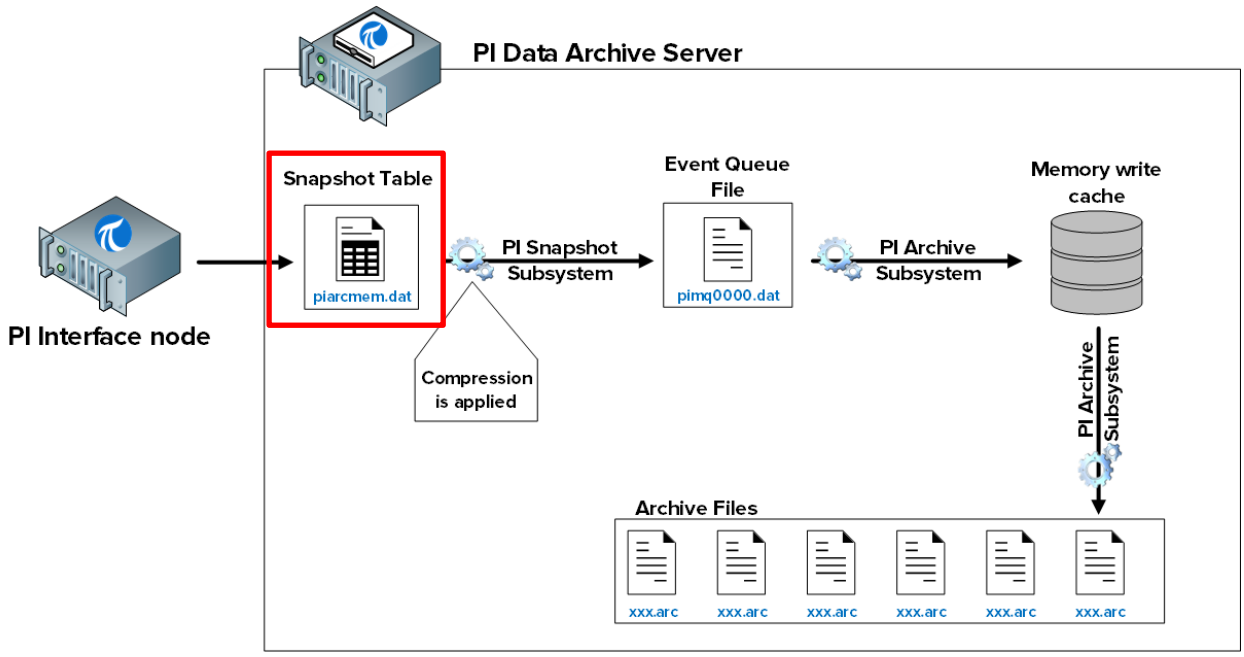
3.3.1 Directed Activity – Inspect Snapshot table statistics



In this part of the class, you will perform a learning activity to explore the different concepts presented in this chapter or section. You may be invited to watch what the instructor is doing or perform the same steps at the same time. You may play a game or hold a quiz. Your instructor will have directions.

Activity Objectives

Learn how to monitor the snapshot table:



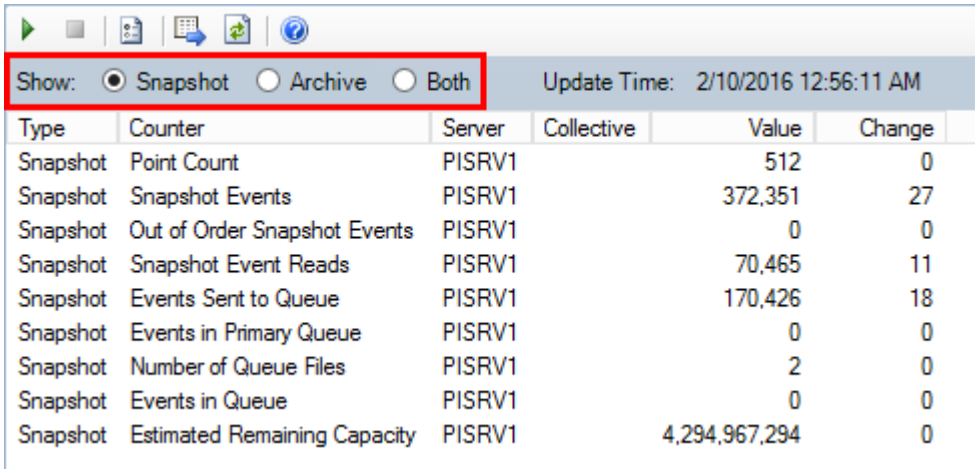
Approach

There are two ways of inspecting the snapshot table statistics:

Method 1 – Using System Management Tools

Step 1 : On PISRV01, run SMT, and navigate to Operation > Snapshot and Archive Statistics

Step 2 : At the top of the page, change the radio button to show only Snapshot statistics



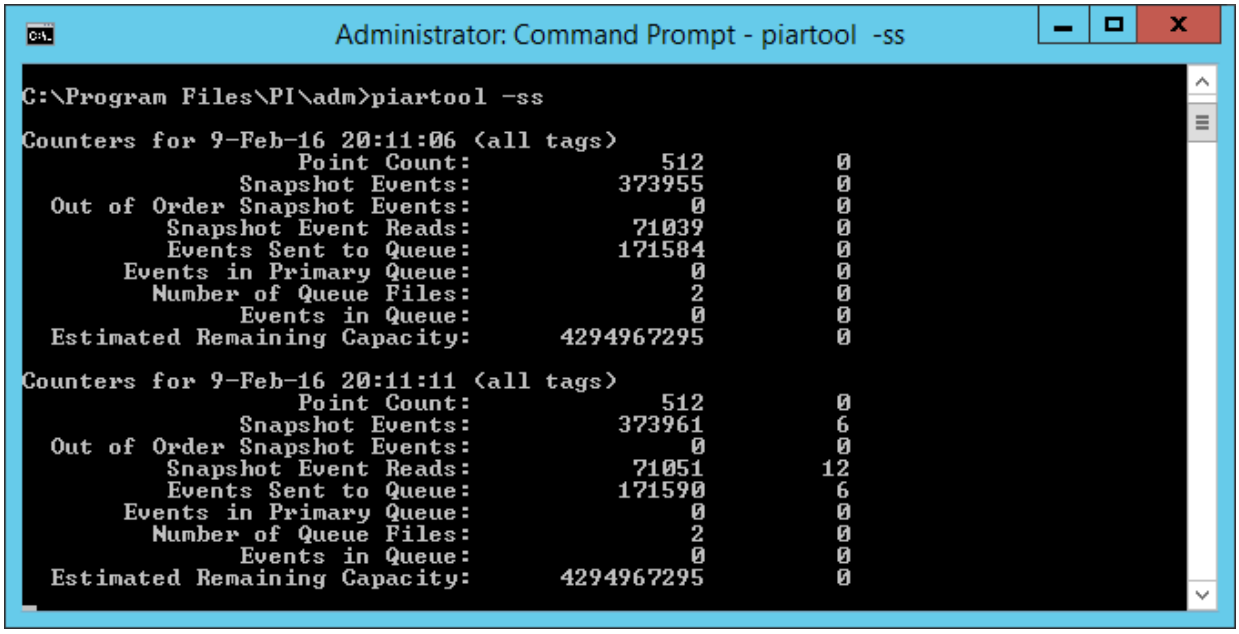
Show: Snapshot Archive Both Update Time: 2/10/2016 12:56:11 AM

Type	Counter	Server	Collective	Value	Change
Snapshot	Point Count	PISRV1		512	0
Snapshot	Snapshot Events	PISRV1		372,351	27
Snapshot	Out of Order Snapshot Events	PISRV1		0	0
Snapshot	Snapshot Event Reads	PISRV1		70,465	11
Snapshot	Events Sent to Queue	PISRV1		170,426	18
Snapshot	Events in Primary Queue	PISRV1		0	0
Snapshot	Number of Queue Files	PISRV1		2	0
Snapshot	Events in Queue	PISRV1		0	0
Snapshot	Estimated Remaining Capacity	PISRV1		4,294,967,294	0

Method 2 – Using the command line

Step 1 : From PISRV01, run the command prompt, and navigate to the folder C:\Program Files\PI\adm
Hint: type “cd %piserver%\adm”

Step 2 : Run the command **piartool –ss**



This command will post the snapshot table statistics every 5 seconds. The left column displays the current statistics, and right column indicates the change in statistics since the last refresh. Press “Ctrl + C” to exit from the snapshot statistics. Some important statistics are:

- **Snapshot events**: Number of events being held in the snapshot table
- **Out of Order Snapshot Events**: Events that have passed through the snapshot table which were older than the current snapshot. A large number of OOO events can cause performance issues

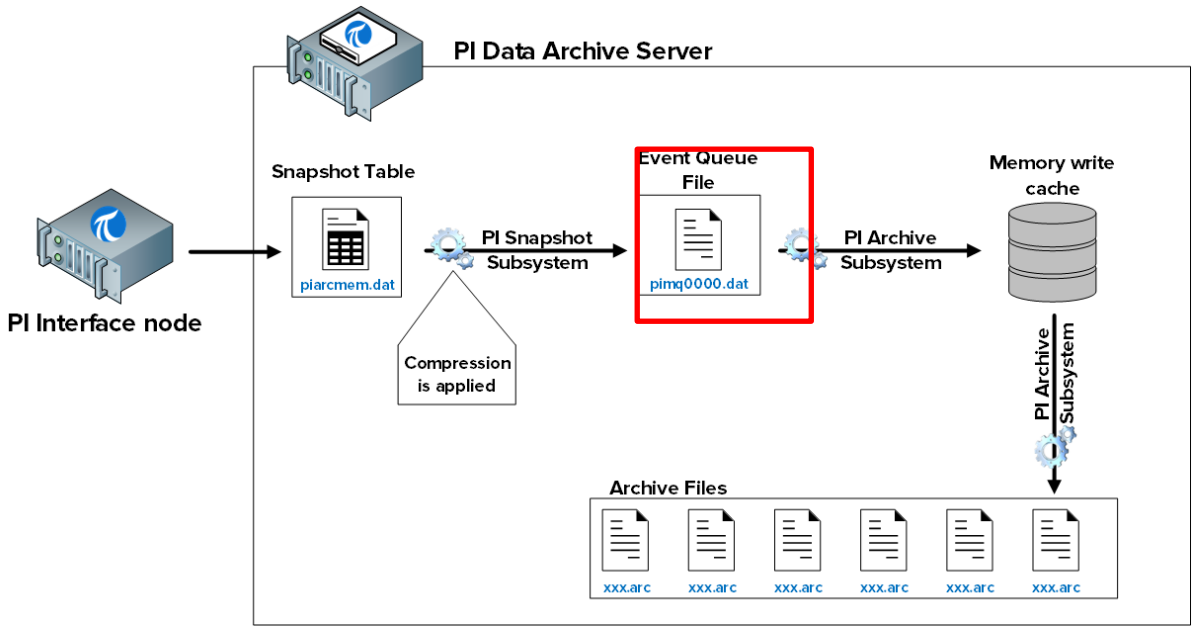
3.3.2 Directed Activity – Inspect Event Queue Statistics



In this part of the class, you will perform a learning activity to explore the different concepts presented in this chapter or section. You may be invited to watch what the instructor is doing or perform the same steps at the same time. You may play a game or hold a quiz. Your instructor will have directions.

Activity Objectives

Learn how to monitor the event queue:



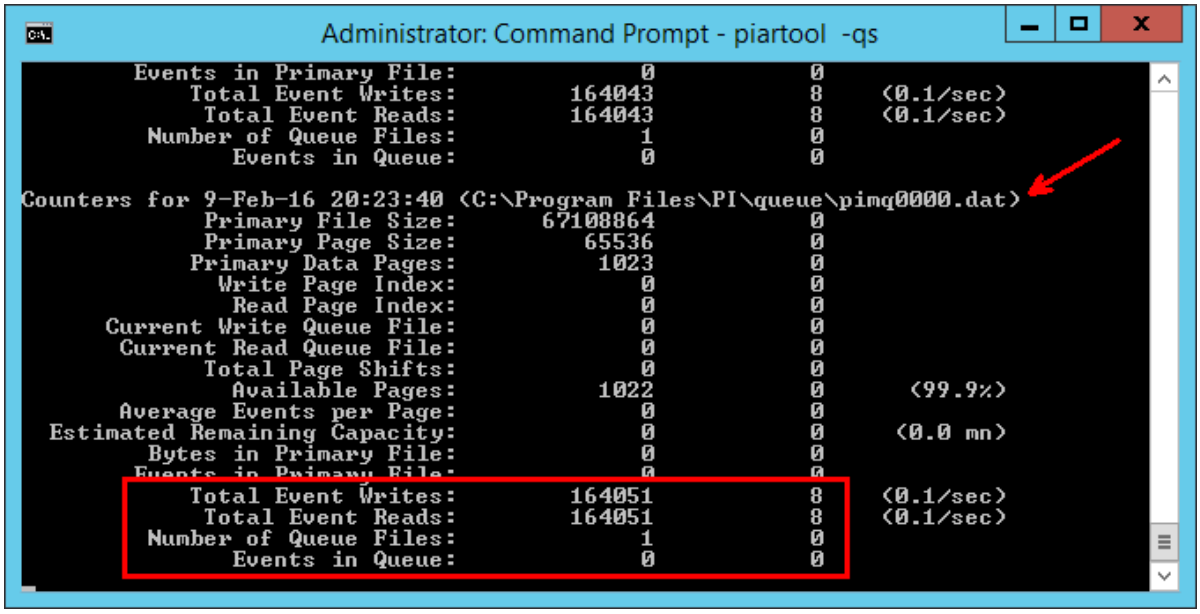
Approach

Step 1 : From PISRV01, run the command prompt, and navigate to the folder C:\Program Files\PI\adm
Hint: type “cd %piserver%\adm”

Step 2 : Run the command **piartool –qs**

This command will post the Event Queue statistics every 5 seconds. Important statistics include:

- The current event queue file name and location are printed in the first line.
- **Total Event Reads and Total Event Writes**: Reads and writes from the event queue file. These numbers should be increasing by the same amount. If the reads are increasing and the writes are not, this could indicate a problem. Contact AVEVA technical support.
- **Number of event queue files**: Under normal operation, this should be 1. If the number of reads exceeds the number of writes, the event queue will fill up, and a new event queue file will be created. Again, this indicates a potential issue.


Administrator: Command Prompt - piartool -qs
Events in Primary File: 0 0
Total Event Writes: 164043 8 (0.1/sec)
Total Event Reads: 164043 8 (0.1/sec)
Number of Queue Files: 1 0
Events in Queue: 0 0
Counters for 9-Feb-16 20:23:40 (C:\Program Files\PI\queue\pimq0000.dat)
Primary File Size: 67108864 0
Primary Page Size: 65536 0
Primary Data Pages: 1023 0
Write Page Index: 0 0
Read Page Index: 0 0
Current Write Queue File: 0 0
Current Read Queue File: 0 0
Total Page Shifts: 0 0
Available Pages: 1022 0 (99.9%)
Average Events per Page: 0 0
Estimated Remaining Capacity: 0 0 (0.0 mn)
Bytes in Primary File: 0 0
Events in Primary File: 0 0
Total Event Writes: 164051 8 (0.1/sec)
Total Event Reads: 164051 8 (0.1/sec)
Number of Queue Files: 1 0
Events in Queue: 0 0

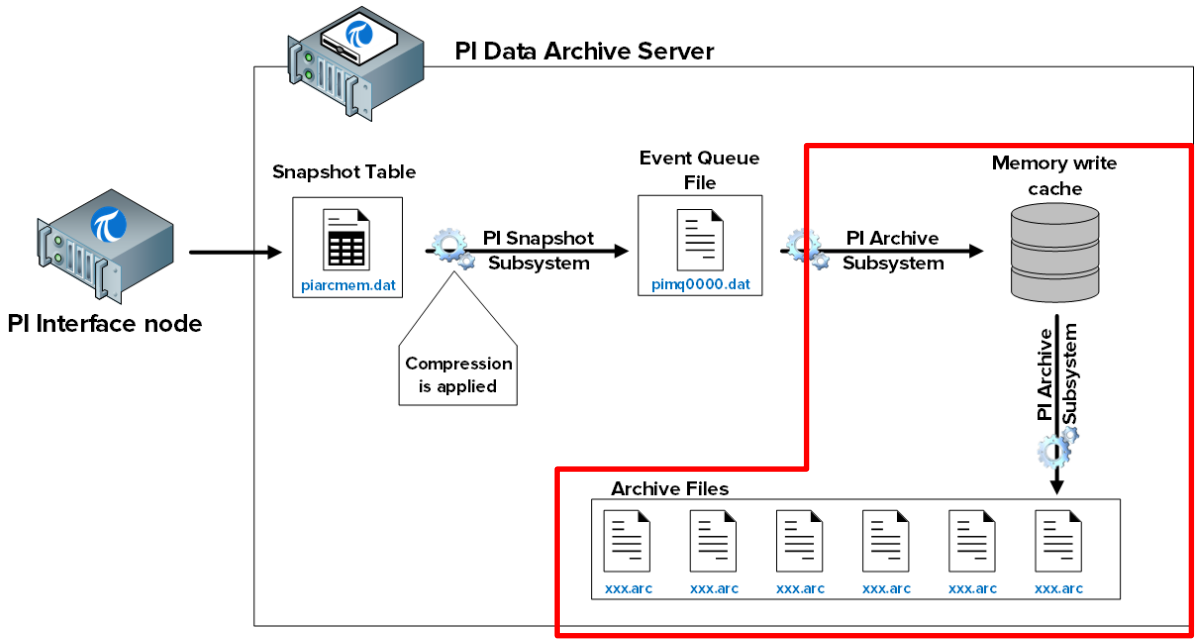
3.3.3 Directed Activity – Inspect the archive statistics



In this part of the class, you will perform a learning activity to explore the different concepts presented in this chapter or section. You may be invited to watch what the instructor is doing or perform the same steps at the same time. You may play a game or hold a quiz. Your instructor will have directions.

Activity Objectives

Learn how to monitor archiving:



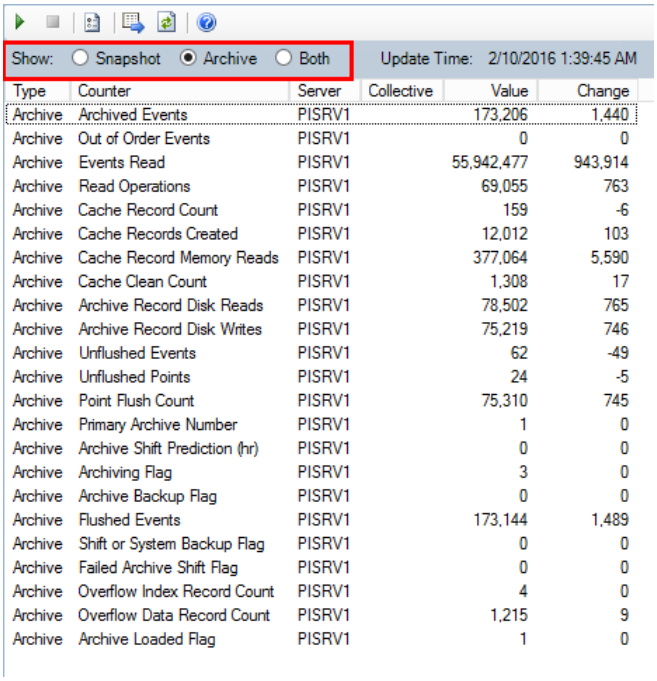
Approach

There are two ways of inspecting the archive table statistics:

Method 1 – Using System Management Tools

Step 1 : On PISRV01, run SMT, and navigate to Operation > Snapshot and Archive Statistics

Step 2 : At the top of the page, change the radio button to show only Archive statistics



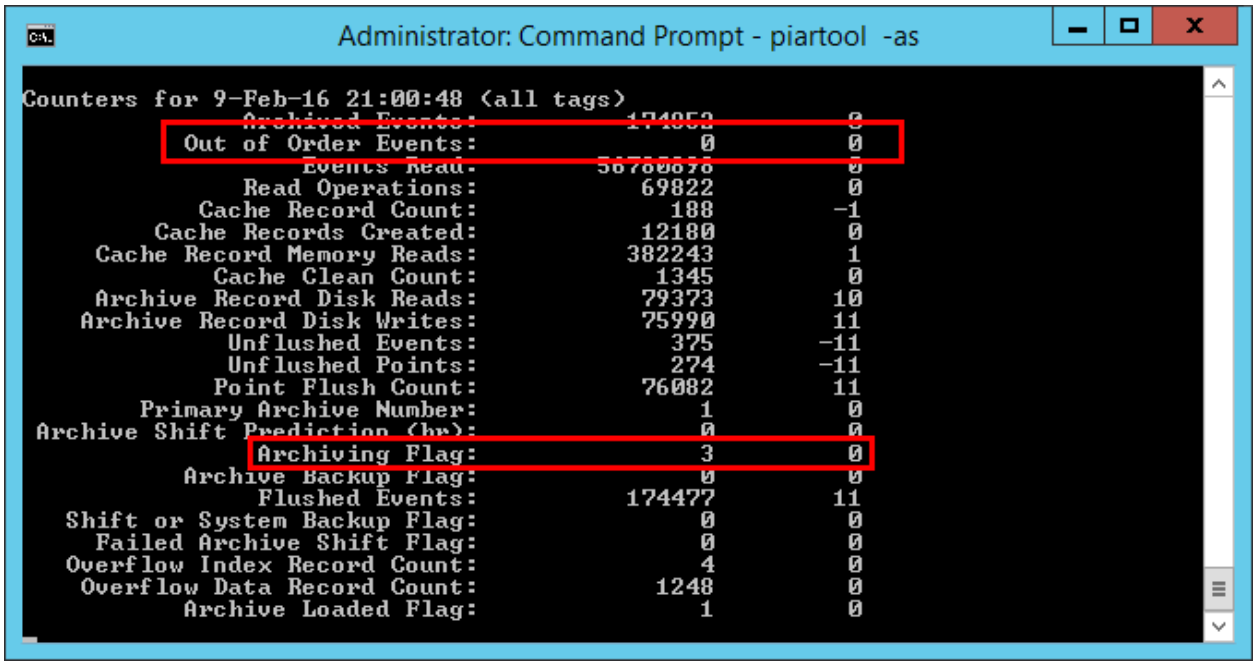
Show: Snapshot Archive Both Update Time: 2/10/2016 1:39:45 AM

Type	Counter	Server	Collective	Value	Change
Archive	Archived Events	PISRV1		173,206	1,440
Archive	Out of Order Events	PISRV1		0	0
Archive	Events Read	PISRV1		55,942,477	943,914
Archive	Read Operations	PISRV1		69,055	763
Archive	Cache Record Count	PISRV1		159	-6
Archive	Cache Records Created	PISRV1		12,012	103
Archive	Cache Record Memory Reads	PISRV1		377,064	5,590
Archive	Cache Clean Count	PISRV1		1,308	17
Archive	Archive Record Disk Reads	PISRV1		78,502	765
Archive	Archive Record Disk Writes	PISRV1		75,219	746
Archive	Unflushed Events	PISRV1		62	-49
Archive	Unflushed Points	PISRV1		24	-5
Archive	Point Flush Count	PISRV1		75,310	745
Archive	Primary Archive Number	PISRV1		1	0
Archive	Archive Shift Prediction (hr)	PISRV1		0	0
Archive	Archiving Flag	PISRV1		3	0
Archive	Archive Backup Flag	PISRV1		0	0
Archive	Flushed Events	PISRV1		173,144	1,489
Archive	Shift or System Backup Flag	PISRV1		0	0
Archive	Failed Archive Shift Flag	PISRV1		0	0
Archive	Overflow Index Record Count	PISRV1		4	0
Archive	Overflow Data Record Count	PISRV1		1,215	9
Archive	Archive Loaded Flag	PISRV1		1	0

Method 2 – Using the command line

Step 1 : From PISRV01, run the command prompt, and navigate to the folder C:\Program Files\PI\adm
Hint: type “cd %piserver%\adm”

Step 2 : Run the command **piartool –as**



This command will post the Archive statistics every 5 seconds. Important statistics include:

- **Archiving Flag**: This flag indicates whether data is being archived:

  0: Data is not being archived

  1: Only historic data is being archived

2: Only future data is being archived

3: Historic and future data is being archived

For Data Archive 2012 and earlier, a value of 1 is healthy. For Data Archive 2015 and later, a value of 3 is healthy. An unhealthy Archiving flag indicates an issue. Contact AVEVA Technical Support.

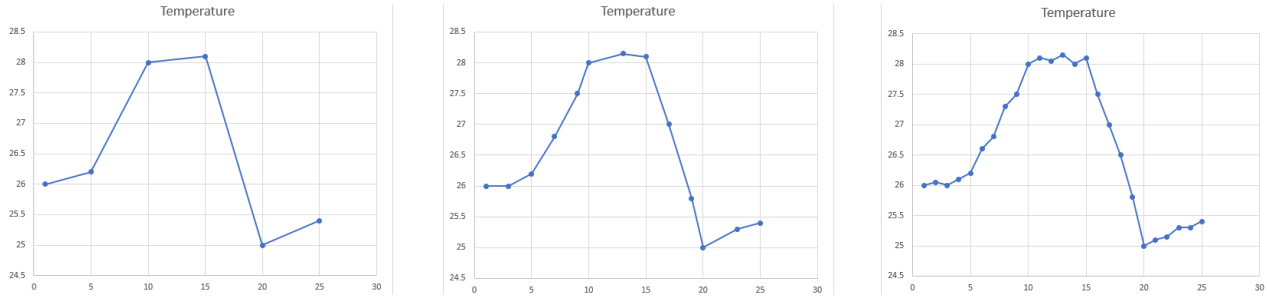
- **Out of Order Events**: Events which are older than the last value written to the archive. A large number of OOO events can cause performance issues for the PI Archive Subsystem.

3.4 Understanding Exception and Compression

3.4.1 Why use Exception and Compression?

As we've discussed in the previous sections, Exception and Compression are mechanisms by which data is filtered, so that only meaningful data is kept in the Data Archive. Exception is applied by the PI Interface, and Compression is applied by the Snapshot Subsystem.

So why do we bother with them? Why not keep all the unfiltered data in the Data Archive? Look at the three graphs below. They show very similar information with different level of details. What could impact our decision about the level of detail that we want to see?



Exception and Compression have the following benefits:

1. **Storage**: They reduce the amount of space needed to store data, freeing up valuable space on a hard drive. Some historic data can be reduced to 90 percent or more of their original size.
2. **Transmission Speeds**: The amount of time it takes to send data sets over the network depends on the size of the transmitted data set. Reducing data sets minimizes the amount of time it takes to move PI System Data in the network by a considerable margin. This also reduces the financial cost of running a network, since less equipment and bandwidth is needed to transmit the data sets.
3. **Archiving and Backup**: Reducing your data allows other processes to be faster and more efficient such as archiving and backups. The PI Archive subsystem can respond to requests more quickly if it is not busy handling large amounts of unfiltered data.
4. **PI System Performance**: the ultimate result of efficient data processing is an increase in performance that allows the PI System to store more data for a given disk size, with faster data retrieval time for display and in a more secure way with backup features that prevent data loss.

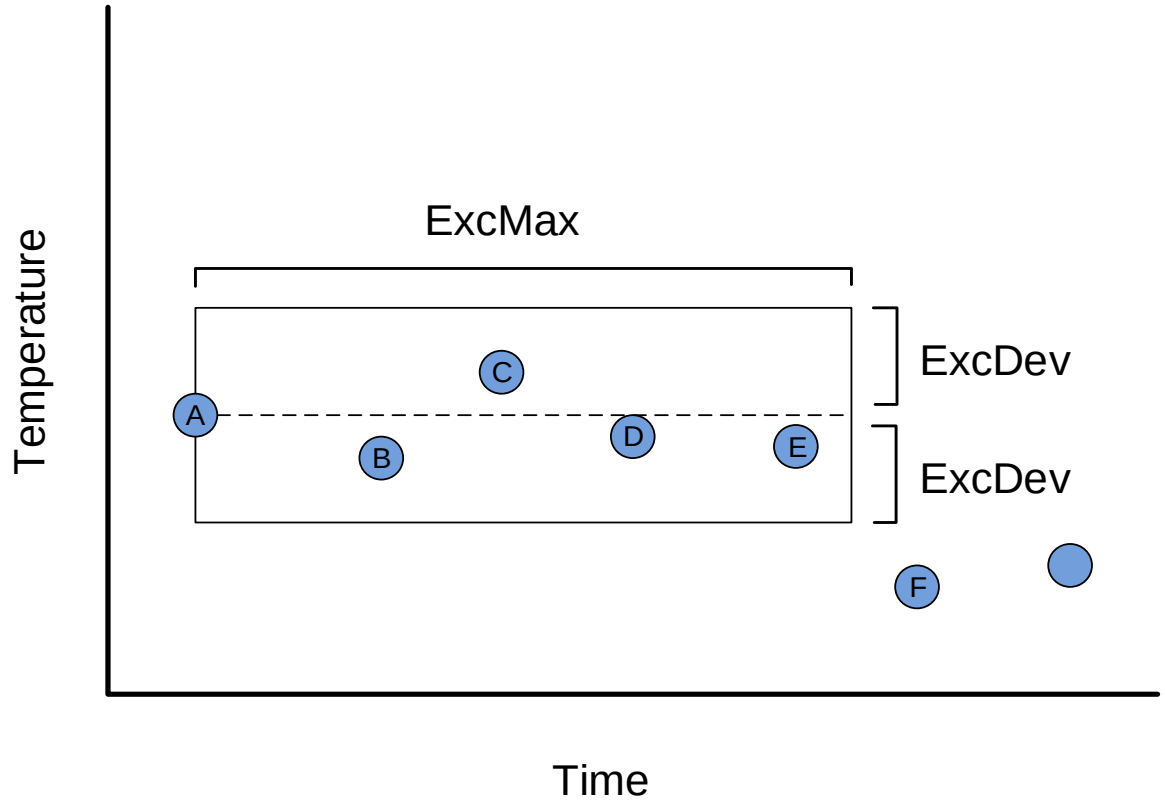
You can go to our YouTube learning channel and watch the video about Exception and Compression.

3.4.2 How does Exception work?

Exception works by removing values that are not changing with time or values whose change is insignificant and below the instrument's accuracy threshold. For example an interface that reads an instrument accurate to within ±0.5, and receives the following values: 1.5, 1.7, 1.6, 1.5, will only store the 1.5 value and show a straight line after that, because all changes were below the instrument's accuracy threshold, and can therefore be considered **noise**.

The exception mechanism uses a simple deadband algorithm to determine whether to send events to the Data Archive. For each PI Point, the following PI Point attributes determine the deadband:

1) **ExcDev** (or **ExcDevPercent**) determines how much a point's value needs to change before the PI Interface sends it to the Data Archive
2) **ExcMax** sets a limit on how long the PI Interface can go without reporting a value to Data Archive. After the ExcMax time period, the PI Interface sends the next new value to Data Archive, regardless of whether the new value is different from the last reported value
3) **ExcMin** sets a limit on how frequently the interface can report values. For example, if you want the interface to wait a full ten minutes before reporting a new value to Data Archive, then you would set the ExcMin attribute to 600 seconds.



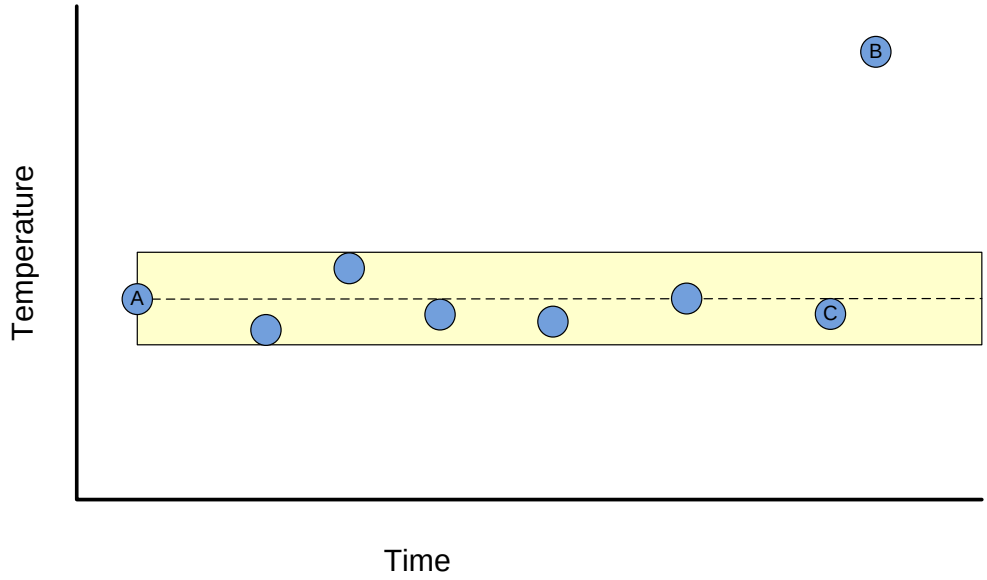
In the illustration above, which values will be sent to the Data Archive?

Answer:

Why do we need the previous value?

We send the value before the exception for a very basic reason: without the previous value, it would not be possible to correctly draw the historic trend.

Consider the series of points below. Draw one trend using just the initial value (Value A) and the value that jumps out of the deadband (Value B). Then draw a trend line that includes not only those two points but the value previous (Value C).



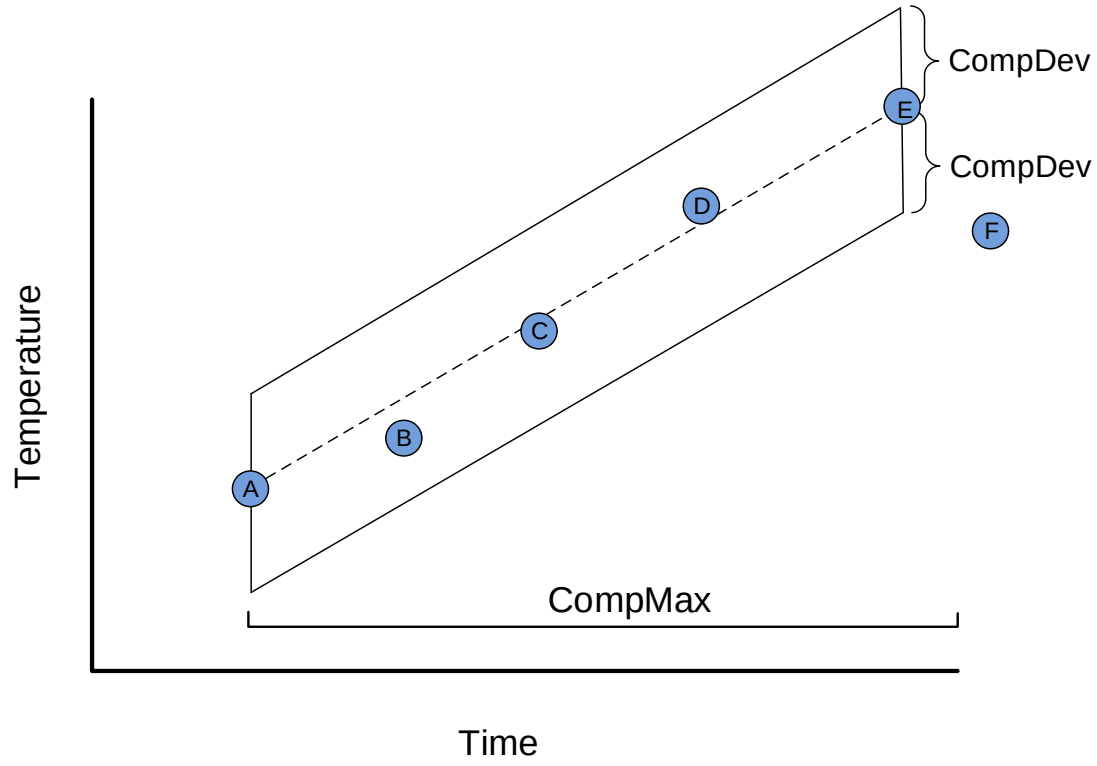
Of the two trend lines you just drew, which trend is more accurate?

3.4.3 How does compression work?

Compression works by removing data that is not meaningful, i.e. data that is not required to accurately reproduce the original data from the data source in a trend.

But isn't everything meaningful?

Not necessarily. For example, consider the following simplified illustration. Which values would you require to accurately represent the data over time?



In the above illustration, which value(s) will be sent to the PI Server?

Answer:

Compression is determined by the following PI Point attributes:

CompDev or **CompDevPercent** determine how much a point's value needs to change for Data Archive to save it.

CompMin and **CompMax** control how often Data Archive saves a new value for a particular point. (This is similar to the ExcMin and ExcMax attributes in exception reporting.)

Note: Additional details on compression can be found in *KB00699 – Compression Explained*.

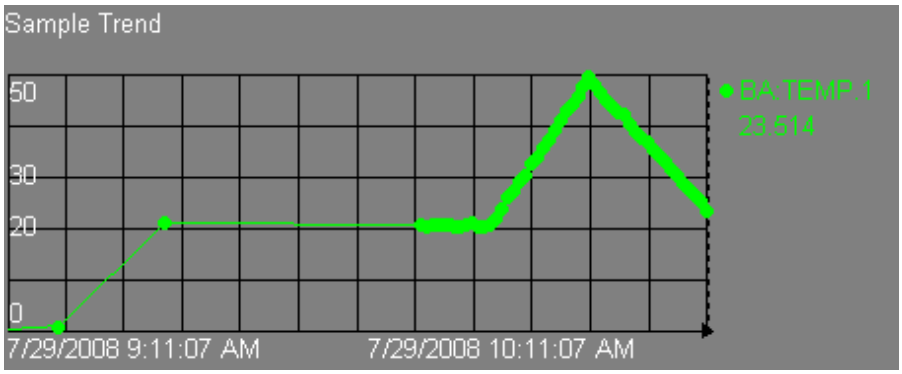
https://customers.osisoft.com/s/knowledgearticle?knowledgeArticleUrl=KB00699

3.4.4 Effects of Exception and Compression on Displayed Data

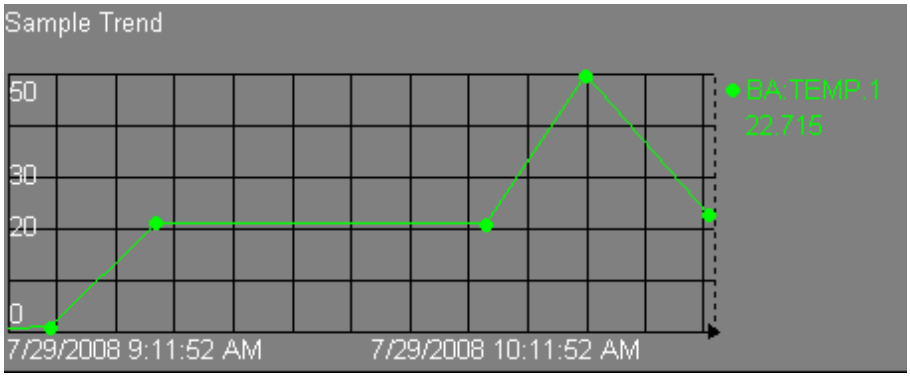
Often you will experience a situation where one of the following conditions will be true:

You are watching a trend and you can see many values, but when the trend is, refreshed most of them disappear.

Before:



After:



This is completely normal behavior and what you are witnessing is the results of compression being applied. In this case, the ProcessBook trend is receiving updates from the snapshot table. However, it only keeps these snapshot values in its local cache for a short period of time. When the trend is refreshed, ProcessBook must query the Data Archive again, and receives data directly from the archive files, where compression has been applied.

3.4.5 Default Values for Exception and Compression

The default values for exception and compression are as follows:

ExcDevPercent = 0.1 (% of span);

ExcMax = 600 seconds (10 minutes);

CompDevPercent = 0.2 (% of span);

CompMax = 28800 seconds (8 hours);

Zero = 0;

Span = 100.

Why are the default values important to me?

Because a deadband that is too wide will filter too much of your data and a deadband that is too narrow may give you a lot of unnecessary data.

On the other hand, there are instances where you would want to capture everything you collect with no exception or compression. You might be performing a calculation and want to capture every result, or you may have regulatory requirements that require you to store every reading.

A very important aspect of being a PI System Administrator is determining a strategy for setting Exception and Compression.

3.4.6 Group Questions – Determine a strategy for setting Exception and Compression



The following questions are intended to reinforce key information, or to discover a new insight. Your instructor may choose to have you try to answer the questions on your own or have the group answer them together aloud.

Question

Determine a strategy for setting Exception and Compression settings for your PI Points.

What would be appropriate for your PI System?

3.5 Data Archive Files

You may have noticed that, when using PI System command line utilities, and when opening PI System files, we have used two directories:

- **PIPC (environment variable %pihome%):** This is the directory where all PI System clients are installed. PI System clients are applications that connect to the Data Archive (PI Interfaces, visualization tools, etc.). There is a 32-bit PIPC folder for 32-bit applications, and a 64-bit PIPC folder for 64-bit applications (**%pihome64%**)
- **PI (environment variable %piserver%):** This is the directory where the Data Archive is installed, and where all Data Archive files and utilities reside.

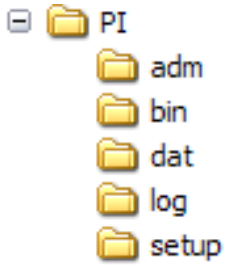
3.5.1 Group Questions – Explore the Data Archive Directory



The following questions are intended to reinforce key information, or to discover a new insight. Your instructor may choose to have you try to answer the questions on your own or have the group answer them together aloud.

Approach

Examine the directories under the C:\Program Files\PI folder:



ADM - administration tools

BIN - binaries

DAT - data files

LOG - message log files

SETUP -additional install kits

Questions

1. Where are the start and stop files for the Data Archive? _____
2. Where is the license file? _____
3. Where is piartool.exe? _____

3.5.2 Directed Activity – Starting and Stopping the Data Archive



In this part of the class, you will perform a learning activity to explore the different concepts presented in this chapter or section. You may be invited to watch what the instructor is doing or perform the same steps at the same time. You may play a game or hold a quiz. Your instructor will have directions.

Activity Objectives

Learn how to properly start and stop the Data Archive

Approach

Part 1 – Stop the Data Archive

Step 1 : On PISRV01, run Windows Explorer.

Step 2 : Navigate to the C:\Program Files\PI\adm folder.

Step 3 : Find the file **pisrvstop.bat**. Right-click on this file and select "Run as administrator".

Step 4 : In the command line window that has opened, observe how each subsystem is shut down in a specific order. If the server is restarted without running this file, the subsystems might not shut down in the right order. It is good practice to always use this file to stop the Data Archive before restarting the server.

Note: You might also notice that the file pisrvsitestop.bat is invoked at the beginning of the script. It is recommended never to edit the pisrvstop.bat file directly. You can add additional commands in the pisrvsitestop.bat file instead.

Step 5 : Go to PIINT01 and open Buffering Manager. Buffering should show a warning and the "events in queue" should increase.

Part 2 – Start the Data Archive

Step 6 : Go back to Windows Explorer.

Step 7 : In the same directory as before, find the file **pisrvstart.bat**. Right-click on this file and select "Run as administrator".



Tip

To speed up the process, create "Data Archive Start" and "Data Archive Stop" icons on your computer desktop pointing to the appropriate batch file.

3.6 Managing Archive Files

In this chapter, we learned how data flows through the Data Archive, and eventually ends up in files called “archive files”. One of the important tasks of a PI System Administrator is to properly manage these files.

3.6.1 Directed Activity – Explore your archive files



In this part of the class, you will perform a learning activity to explore the different concepts presented in this chapter or section. You may be invited to watch what the instructor is doing or perform the same steps at the same time. You may play a game or hold a quiz. Your instructor will have directions.

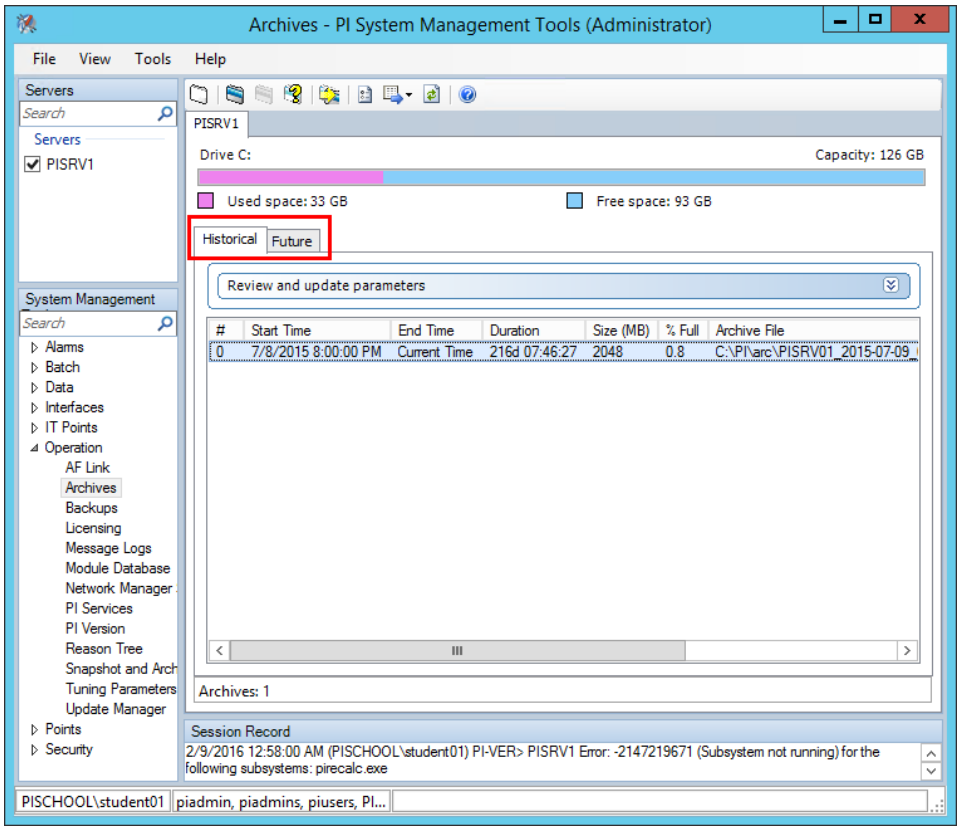
Activity Objectives

Learn how to use System Management Tools to browse your archive files

Approach

Step 1 : On PISRV01, run SMT

Step 2 : Browse to Operations > Archives. Notice that you have two tabs: Historic and Future



Historic vs Future

Since the introduction of Data Archive 2015, two different types of archive files are available: historic archives and future archives. Historic archives store data for non-future PI Points, while future archives store data for **future PI Points**. Only future archive files (and thus future PI Points) can accept data which is more than 10 minutes in the future.

Data in future archives is never mixed or interchanged with data in historic archives. When time passes and future data moves into the past, it is still stored in the same future archives.

Step 3 : Right-click on one of the historic archive files and select “Properties”. Fill out the properties below:

Type: _____

State: _____

Status: _____

Start Time: _____

End Time: _____

Shift Flag: _____

Step 4 : Right-click on one of the future archive files and select “Properties”. Fill out the properties below:

Type: _____

State: _____

Status: _____

Start Time: _____

End Time: _____

Shift Flag: _____

Fixed vs Dynamic

When you create historic archives, by default, they are created with a fixed size, and the memory is allocated upon creation to minimize the potential of fragmentation on disk.

You also have the option of creating dynamic archives. Dynamic archives are files that grow as they are filled. ***For historic archives, dynamic sizes should only be used for troubleshooting and reprocessing of archives.***

When you create future archives, they are created as fixed archives with an initial size of 1MB. If the data stored in that archives ever exceed 1MB the future archive grows dynamically to store the extra data.

Registered vs Unregistered

For the Data Archive to access the data in an archive, the archive must be registered (often referred to as "mounted" in other systems). Archive files can be registered and unregistered using System Management Tools. The registered archives can be located on any drive available to the Data Archive, if there is enough bandwidth to retrieve data.



Tip The Primary Archive should always live on the Data Archive. Older "full" archive files are used less frequently and can be migrated to a storage device.

The Primary Archive

The "primary" archive is the archive where current data is being written to. It has the same characteristics as the other archives with two exceptions:

1. The primary archive cannot be unregistered
2. The primary archive has no end time – it is labeled with "current time"

Historic archives are sequential in time

Each historic archive has a start and an end time. All the data between those two points in time is contained in that file. Historic archives do not overlap in time. When an archive is initialized the timestamp of the first value sets the start time. When it is approximately 98% full a new file is initialized (in case some data comes in late). So, the archives are separated by time seams. These seams are transparent to the user.

The process of initializing a new primary archive is called "**shifting**."

If automatic archive creation is enabled, the Data Archive will create a new historic archive file automatically and use it as the new primary historic archive. We enable it by setting tuning parameters called *Archive_AutoArchiveFileRoot* and *Archive_FutureAutoArchiveFileRoot*. As of Data Archive 2012, automatic archive creation is enabled by default.

When automatic historic archive creation is disabled:

- If there is an empty historic archive file, this file will be promoted as the primary archive file
- If no empty historic archive exists, then the oldest historic archive will become the primary and *its existing data will be overwritten*.



Tip

To prevent overwriting archive files, make sure to:

- Always have plenty of disk space in the archive file directory
- Create alerts within your IT organization to warn against low disk space
- Create at least 2 empty historic archive files

You can also make individual archive files “non-shiftable”. This means they can never become primary and so can never be overwritten.

Future archive and non-sequential data

Future archives are optimized for non-sequential data unlike real-time data stored in the historic archives. This way future archives are only created when necessary.

When a future PI Point receives a value, if a future archive does not yet exist for this timestamp, the PI Archive Subsystem creates new 1 MB fixed archive with a 1-month time span (from the 1st of the month to the1st of the next month). If more than 1 MB of data is sent to this archive, it will become dynamic and grow as needed. Future archives spanning longer time ranges can be created manually.

Step 5 : Take note of the archive file directories for the historic and future archives. In windows explorer, browse to the location of this file. You will notice that each archive file is accompanied by a second file with the file extension .ann.

Historic archive directory: _____

Future archive directory: _____

The Annotation File

Each archive file has an associated annotation file. Annotations allow you to associate arbitrary information, such as text comments and other binary data, with an archived value of a PI Point. It is important to always keep the annotation file in the same directory as the archive file.

3.6.2 Best practices for archive file management

When formulating an archiving strategy, the following best practices should be followed:

Archive sizing

Versions of the Data Archive prior to Data Archive 2012 had a default historic archive size of 256MB. Since the release of Data Archive 2012, the default historic archive size is determined automatically on installation. The following strategy is used to determine the archive size, and is the official recommendation

- (Physical Memory in MB) ÷ 3 OR 3 × (Licensed Point Count)/1024 MB (whichever is smaller)
- Rounded down to nearest power of 2
- No less than 256MB, and no more than 10,240MB

After the original installation, during automatic archive file creation, the size of the new primary archive file will be identical to the current primary archive. The tuning parameter ***Archive_AutoArchiveFileSize*** can be used to change the archive file size for the next archive shift.

Physical Memory (MB)	Historical Archive Size (MB)
0 to 1,535	256 (2^8)
1,536 to 3,071	512 (2^9)
3,072 to 6,143	1,024 (2^10)
6,144 to 12,287	2,048 (2^11)
12,288 to 24,575	4,096 (2^12)
24,576 to 30,719	8,192 (2^13)
30,720 or greater	10,240 (capped)

Archive file location

Ideally, the both the archive files and event queue should be located on separate, dedicated local volumes. Using separate drives allows data to be read from the event queue and written to the archive simultaneously, thereby optimizing data throughput.

During automatic archive file creation, the location of the archive file is determined by the tuning parameters ***Archive_AutoArchiveFileRoot*** and ***Archive_FutureAutoArchiveFileRoot.*** Clearing the value in these tuning parameters will disable automatic archive file creation. The name of the archive file is determined by the tuning parameters ***Archive_AutoArchiveFileExt*** and ***Archive_AutoArchiveFileFormat.***

Additional recommendations

AVEVA also recommends the creation of 2 empty historic archive files.

3.6.3 Solo Exercise – Change the archive file configuration



This is a solo exercise that is designed to maximize learning in a specific topic area. Your instructor will have instructions and will coach you if you need assistance during the exercise.

Exercise Objectives

Learn how to change the archiving parameters using SMT

Problem Description

You were just given the role of PI System Administrator at your company. The PI System at your company was installed a long time ago and was recently upgraded. After reviewing the current archiving configuration in place, you decide to make the following changes:

1. Currently, the historic archive files are stored on the C drive. You decide to move these to a new dedicated drive (E:\).
2. Currently, the archive file size is set to the default of older versions of the Data Archive, 256MB. For new archives, change this file size 512 MB
3. Create 2 empty archives for emergencies.

Try to devise a strategy to accomplish these tasks on your own, before using the Step-by-step approach below.

Approach

Part 1 – Changing the default auto-archive settings

Step 1 : Run SMT. Navigate to Operation > Tuning Parameters > Archive Tab

Step 2 : Change the value of ***Archive_AutoArchiveFileRoot*** to E:\PIArchives\PISRV01

Step 3 : Change the value of ***Archive_AutoArchiveFileSize*** to 512 MB

Step 4 : Force an archive shift. Navigate to Operation > Archives. Press the “Force an archive shift” button 

Step 5 : On PISRV01, open a command prompt and navigate to the “C:\Program Files\PI\adm” directory. Run piartool –al to see the progress of archive shift

Step 6 : Make sure your new archive was created automatically with the correct name, size and location

Part 2 – Move the existing archives to the new location

Step 7 : In SMT, navigate to the Operation > Archives. On the “Historic” tab, select all the archives that are in the C:\PI\arc directory

Step 8 : Unregister the archives using the “Unregister selected archive” button . **Note:** Unregistered archive will still show up in SMT until you click the refresh button .

Step 9 : Now that the archive files are unregistered, we can move them to the new location. Copy-paste the archive files to E:\PIArchives. Whenever moving archive files (.arc), ensure to move the corresponding annotation files (.ann)

Step 10 : Since we just copied the file, the account running the PI Archive Subsystem does not have correct access to it. We can give access using the file properties. But we need to do this for each file, and we need to know what the minimum access to provide is. Instead, we can use a command that will give correct access to all the accounts on all the relevant folders.

a. Run the command prompt as administrator. Navigate to the directory C:\Program Files\PI\adm. **Hint:** type “cd %piserver%\adm”

b. Run the following command:

pidiag -updateFolderSecurity

Step 11 : Back in SMT, press the “Register an archive” button . Select all the archives that you moved.

Part 3 – Create empty archives

Step 12 : In SMT, navigate to the Operation > Archives. Press the “Create a new archive” button .

Step 13 : Create two empty archives with no defined start time or end time

3.7 Manage Tuning Parameters

In the previous exercise, we used tuning parameters to change the behavior of the auto-archiving functionality of the Data Archive. There are many more tuning parameters that can be used to change the default behavior of the Data Archive.

The default values of these tuning parameters are set so that common Data Archive installations have the best possible configuration. However, each Data Archive is unique, and sometimes these tuning parameters need to be adjusted.

3.7.1 Group Questions – Tuning Parameters



The following questions are intended to reinforce key information, or to discover a new insight. Your instructor may choose to have you try to answer the questions on your own or have the group answer them together aloud.

Questions

For the tuning parameters below, answer the following questions:

- What is the purpose of this tuning parameter?
- What are the defaults?
- Under what conditions do you think you would have to alter the values?
- What considerations would you have in resetting this?

1. **EnableAudit**

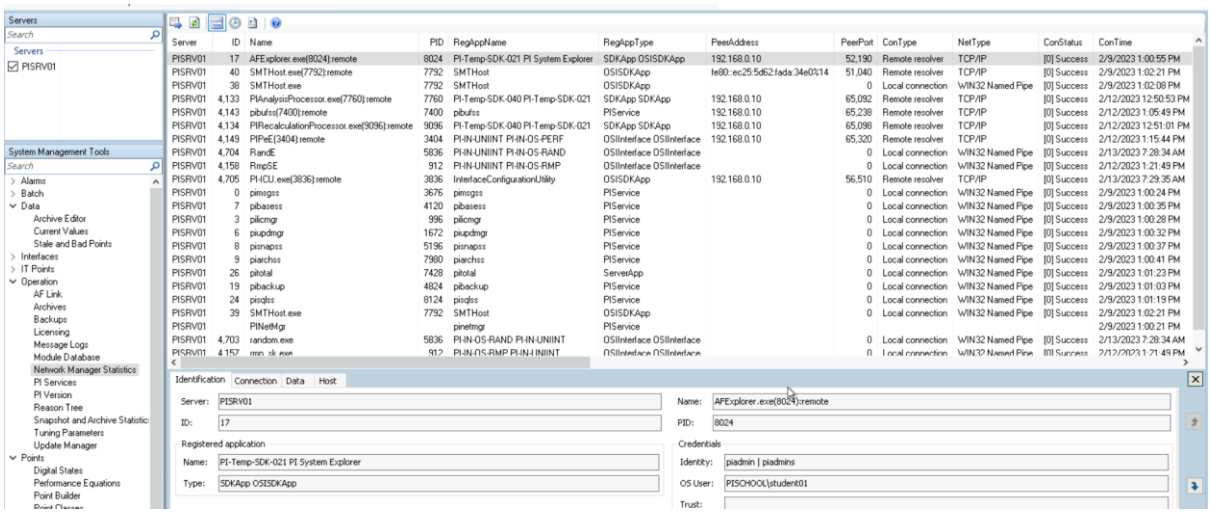
2. **Archive_LowDiskSpaceMB**

3. **Snapshot_EventQueuePath**

4. **TotalUpdateQueue and MaxUpdateQueue**

3.8 Connections to PI Data Archive

Connections to PI Data Archive are managed by PI Network Manager. As an administrator, you might want to know what applications are connected, how much data is being transferred and what are the IPs that are used in these. All this information can be found in Network Manager Statistics in PI Data Archive. You can see them in PI SMT > Operations > Network Manager Statistics.



Some useful columns:

- Name: Name of the application. Interfaces usually look in the format of xxxxE (for example, OPCpE for OPC DA Interface, PIPeE for PI PerfMon interface)
- PeerAddress: IP address of the application.
- Identity: shows list of PI Identities that this connection has. For PI Mappings there might be several PI Identities.
- OSUser: shows the username of the Windows user if PI Mapping is used for connection.
- Trust: shows the name of the PI Trust if PI Trust is used for connection.

3.8.1 Solo Exercise – Find the connections that use PI Trusts



This is a solo exercise that is designed to maximize learning in a specific topic area. Your instructor will have instructions and will coach you if you need assistance during the exercise.

Exercise Objectives

AVEVA recommends using PI Mappings instead of PI Trusts. Therefore, we would like to find all the application that use PI Trusts. We will see how to move from PI Trusts to PI Mappings in the PI Security chapter.

Approach

Step 1 : From PISRV01, open PI SMT and go to Operations > Network Manager Statistics.

Step 2 : Using the column Trusts, find all the applications that are using PI Trusts. Note them down

Application(s):_____

Trust name(s): _____

IP(s):_____

Two of these connections are from PISRV01. Do you know what are these applications? If not, ask your instructor.

3.9 Manage Data Archive Backups

Throughout this section, we have gone under the hood of the Data Archive. We have learned that the Data Archive is composed of the following:

- Subsystems (windows services) that perform tasks
- Files that hold data (snapshot table, event queues, archives)
- Files that hold configuration information (PI Point configuration, tuning parameters)
- The physical hardware (CPU, RAM, hard drive) on which all of these components depend.

Given our new insights, it is now possible to imagine all the ways our Data Archive might be in danger.

3.9.1 Group Activity – Why backup?



The following questions are intended to reinforce key information, or to discover a new insight. Your instructor may choose to have you try to answer the questions on your own or have the group answer them together aloud.

Activity Objectives

Describe the importance of data backups

Questions

Your instructor will give you a couple of minutes. Document below:

- Scenarios that you can think of where you would need a backup of your data
- The types of data you feel would be critical
- Some elements of a disaster recovery plan that are key

Your instructor will then lead a discussion on your answers.

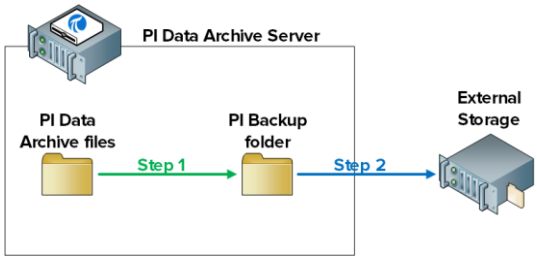
3.9.2 Your Backup Strategy

The Data Archive has a service called the PI Backup Subsystem. This service can be used to create a local "backup" of specific Data Archive files.

AVEVA recommends that you perform nightly backups of your Data Archive, using the two-step backup strategy:

The two-step backup

The Data Archive files are copied to a local "PI Backup" folder. This folder is then copied to an external storage device, usually using a 3rd party application.



Note: There are two other possible backup strategies which are not officially recommended or supported by AVEVA:

1. Taking a direct VSS backup of the Data Archive Server using 3rd party backup software
2. If the Data Archive is running on a Virtual Machine, taking a child VM snapshot

If your company choses to implement either of these two alternatives, you will be responsible for validating your backup and restore procedures. Neither method should be selected without conducting extremely thorough testing and verification. For more information on the potential pitfalls of these backup strategies, please consult https://customers.osisoft.com/s/knowledgearticle?knowledgeArticleUrl=KB00659.

3.9.3 How the Data Archive backup works

Which files get backed up?

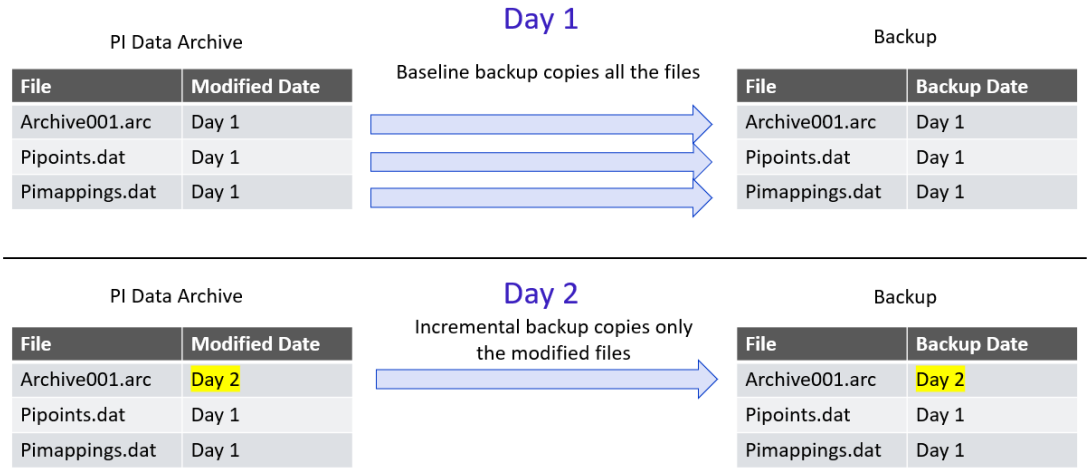
The PI Backup subsystem creates a copy of all the Data Archive files that have been created or edited since the original installation. In other words, everything that contains data or configuration information. These are the only files which are needed to restore a Data Archive.

The folders and their contents are listed below:

- **adm**: pisrvsitestart.bat, pisrvsitestop.bat, pisitestart.bat, pisitebackup.bat
- **archive file directory**: historic archive and annotation files
- **future archive file directory**: future archive and annotation files
- **bin**: pipeschd.bat
- **dat**: everything
- **log**: everything
- **PIPC (32-bit and 64-bit)**: all bat, log, ini, txt, and sql files, along with the ACE Executables and ACE Class Libraries (only when pisitebackup is invoked)

Note: If the AF Database is installed on the Data Archive, on a SQL Server Express edition, the AF Database (PIFD) is also backed up. We will continue our discussion of AF Server backups in the following chapter.

The PI Backup subsystem takes ***incremental*** backups of the Data Archive. This means that, when a backup is performed, only files which have been modified since the last backup are copied to the PI Backup directory. Thus, resources are not wasted on overwriting files which have not changed.



Can the Data Archive be accessed during the backup?

Since the Data Archive backup uses Microsoft's Volume Shadow Copy Services (VSS), the Data Archive remains online and can be accessed as usual during the backup.

However, to minimize the impact of a backup on users, AVEVA recommends the following:

1. The daily backup should be performed at an off-peak time. The default time is at 3:15 AM.
2. The PI Backup directory should be on a dedicated, physical drive.

How do I configure a daily backup?

The procedure is as follows:

1. Establish a baseline backup of the Data Archive to a PI Backup folder.
2. Set up a daily Windows Scheduled Task that runs the incremental Data Archive backup to the same PI Backup folder.
3. Either one of the following steps:
    a. Use a third-party backup tool to automate a regular backup of the PI Backup folder to external storage. Data Archive includes a script that you can use for this, if a third-party tool is not an option (pisitebackup.bat).
    b. Take a backup of the entire Data Archive server using a third-party software
    c. Take a snapshot of the Data Archive virtual machine

Note: For new installations of the Data Archive, the first incremental backup is a full backup. Therefore, step 1 is not necessary. For upgraded or moved Data Archives, step 1 is necessary.

3.9.4 Solo Exercise – Configure a daily backup



This is a solo exercise that is designed to maximize learning in a specific topic area. Your instructor will have instructions and will coach you if you need assistance during the exercise.

Exercise Objectives

Configure a daily backup of the Data Archive

Approach

Part 1 – Establish a baseline backup

Step 3 : From PISRV01, run the command prompt. Navigate to the directory C:\Program Files\PI\adm. **Hint:** type “cd %piserver%\adm”

Step 4 : Run the following command:

piartool -backup E:\PIBackup -type FULL -arcdir

Step 5 : Validate that the PI Backup worked.

a. Open the E:\PIBackup folder.

   i. Look at the files that were copied

b. Run SMT. Navigate to Operation > Backups. Take note of the type and status of the backup.

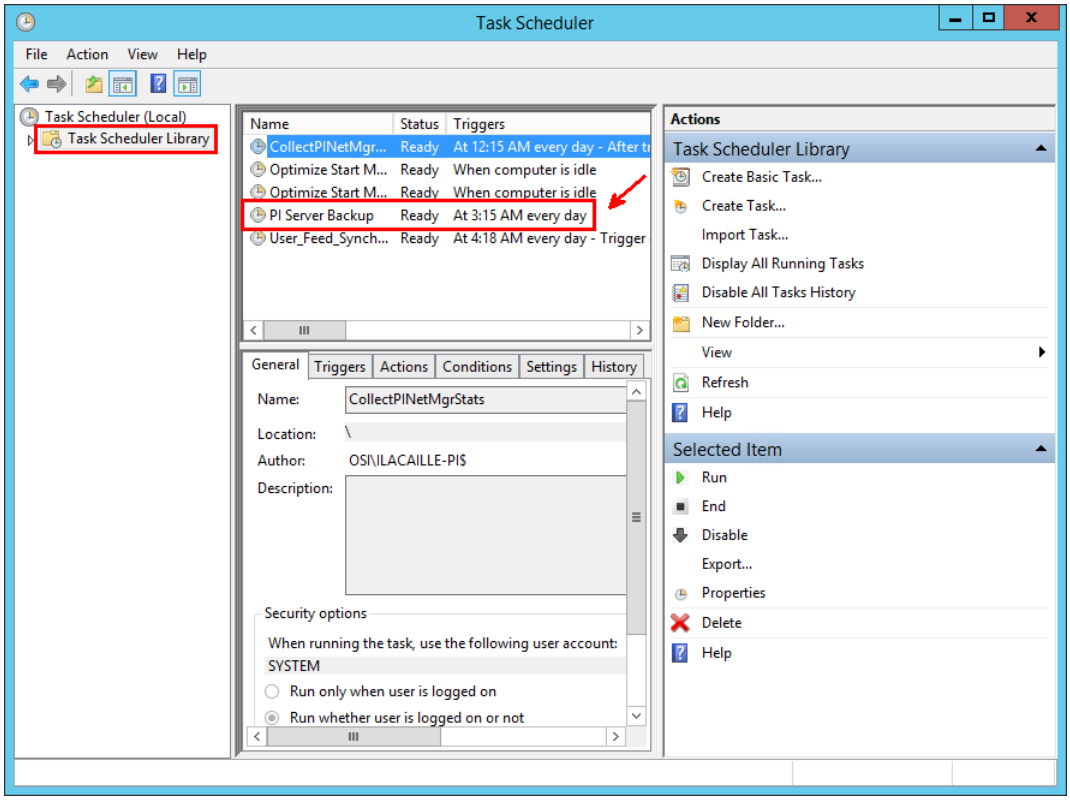
Part 2 – Set up a daily Window Scheduled task to run the backup

Step 1 : In the same command prompt, run the command:

pibackup E:\PIBackup –install

Step 2 : Run the Task Scheduler snap in (taskschd.msc)

Step 3 : In the left pane, select the Task Scheduler Library. You should have a scheduled task named “PI Server Backup”



Step 4 : Right-click on the task and select “Properties”. Go to the “Triggers” tab. This is where you can modify the default scheduling of the task, if needed.

Step 5 : Test your new scheduled task. Right-click on the task and select “Run”

Step 6 : Validate that the PI Backup worked.

a. Open the E:\PIBackup folder.

   i. Open the backup message log file “pibackup_<date>. Are there any errors?

b. Run SMT. Navigate to Operation > Backups. Take note of the type and status of the backup.

Note: It is also possible to track your backups using PI Performance Monitor points. AVEVA recommends that you monitor the following Windows performance counters for PI Backup Subsystem:

Last Backup Failed: Will be 1 if the last backup failed; otherwise, it will be 0.

Backups Started: Should increase by 1 every night if you have a nightly backup task installed.

Failed Backups: Will increase by 1 for every failed backup.

Note that if the pisitebackup.bat or a third-party backup of your backup directory fails, this will not be reflected in the Performance Counters.

3.9.5 Restoring the Data Archive from a backup

A Data Archive Backup can be restored to an existing Data Archive, or to a brand-new computer. Common reasons for restoring a backup include:

- Recovering from a server hardware failure
- Establishing a development server based on a production server
- Moving the Data Archive to a new server

Do not hesitate to contact AVEVA Technical Support for assistance during this process.



For specific steps on how to restore a Data Archive see "Restore a backup to an existing Data Archive" or “Restore a Data Archive backup to a new computer” in *PI Server 2024 User Guide.*

3.10 Troubleshooting Exercise



The following group exercise is intended to reinforce the concepts we have learned so far. The instructor will act as an end user with an issue, and you will resolve their issue as a group.

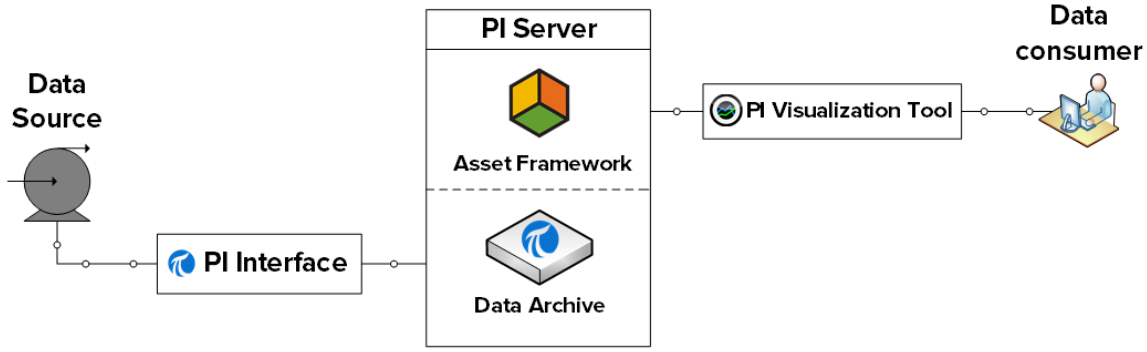
4. Asset Framework Management

Objectives

- Define the role of the Asset Framework
- Describe an asset/element
- Describe an attribute
- Describe PI System Explorer
- Define the four AF Element Attribute types
- Explain the relationship between an attribute and a PI Point
- Explain how data flows in the PI System when users are viewing an attribute
- View attribute data using PI System Explorer
- View attribute data using PI ProcessBook
- Describe the advantages of viewing attributes instead of PI Point
- Describe and create a AF Database
- Create an asset with attributes associated to existing PI Points in PSE
- Describe an template
- Explain the benefits of an AF Template
- Create a template
- Create assets in bulk using PI Builder
- Describe the components of the AF Server (service, SQL database)
- Configure a local backup of the AF Server
- Describe how to restore a backup of the AF Server

4.1 Define the role of the Asset Framework

In the first chapter, we learned that the Asset Framework is one of the crucial software components of a basic PI System. Together, the Data Archive and the Asset Framework make up the PI Server. While the Data Archive stores data, the Asset Framework *organizes* and *enhances* data. What exactly does this mean?

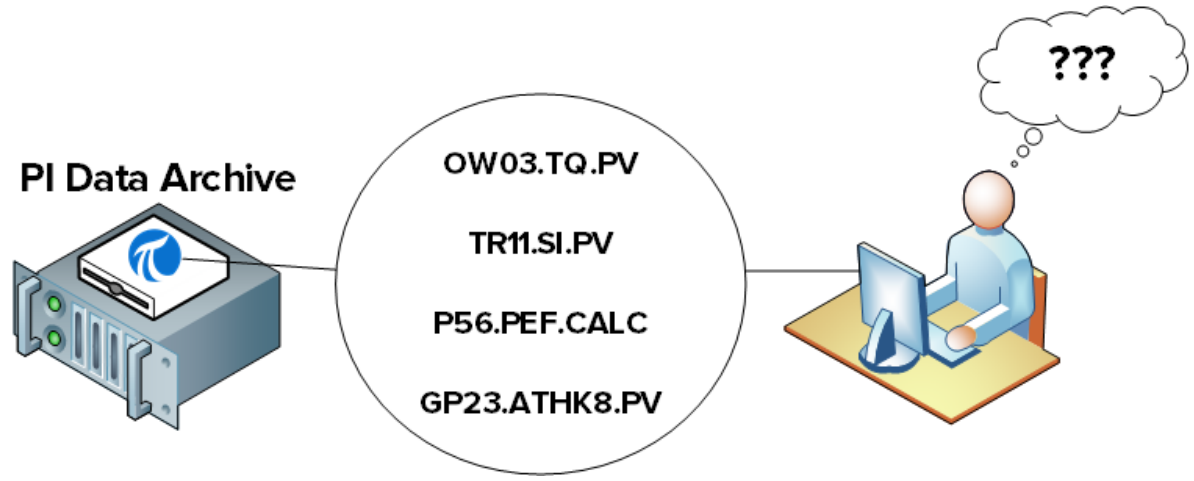


Organizing the data

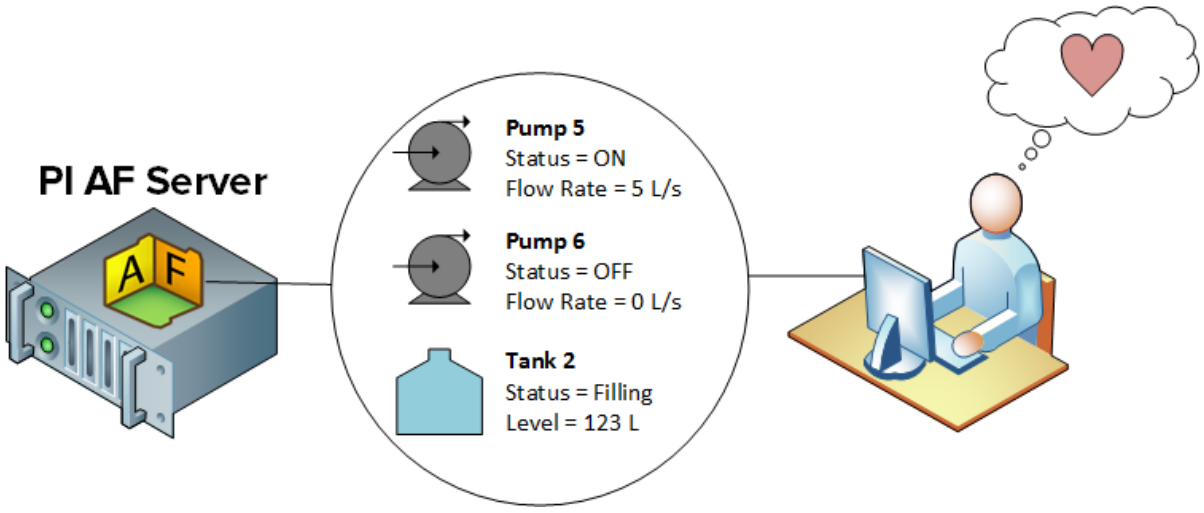
In chapter 1, we used PI Vision to visualize data from the PI Point "BA:TEMP.1", which collects our Reactor 1 temperature. The name of the PI Point was based off a naming convention unknown to us as the user.

This naming convention could come from anywhere and mean anything… It could have been decided by a PI System Administrator 10 years ago… It could reflect the name of the data stream at the data source (e.g. in the DCS)… Generally speaking, PI Point names are *machine friendly*, not *human friendly*.

In most cases, a new PI System user has no way of knowing which PI Point they need. They are doomed to rely on their colleagues or the PI System Administrator, and might not be all that excited about using the PI System.



This is where the Asset Framework comes in. It presents PI System users with an alternative, human friendly view of the data. The data is organized such a way that a brand new PI System user can instantly recognize their process, and easily find the data they are looking for.



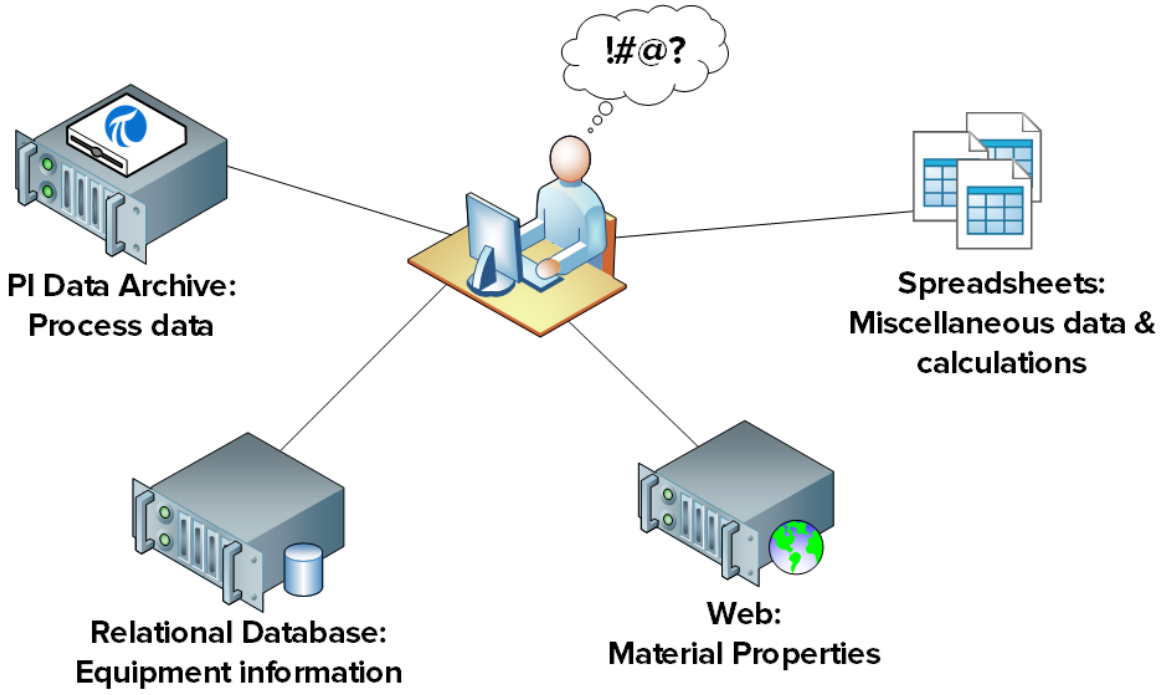
Enhancing the data

The only type of PI System data we've talked about so far is time-series data, i.e. data that changes often with time. However, there is a lot of static data that can be important for data consumers to access, such as:

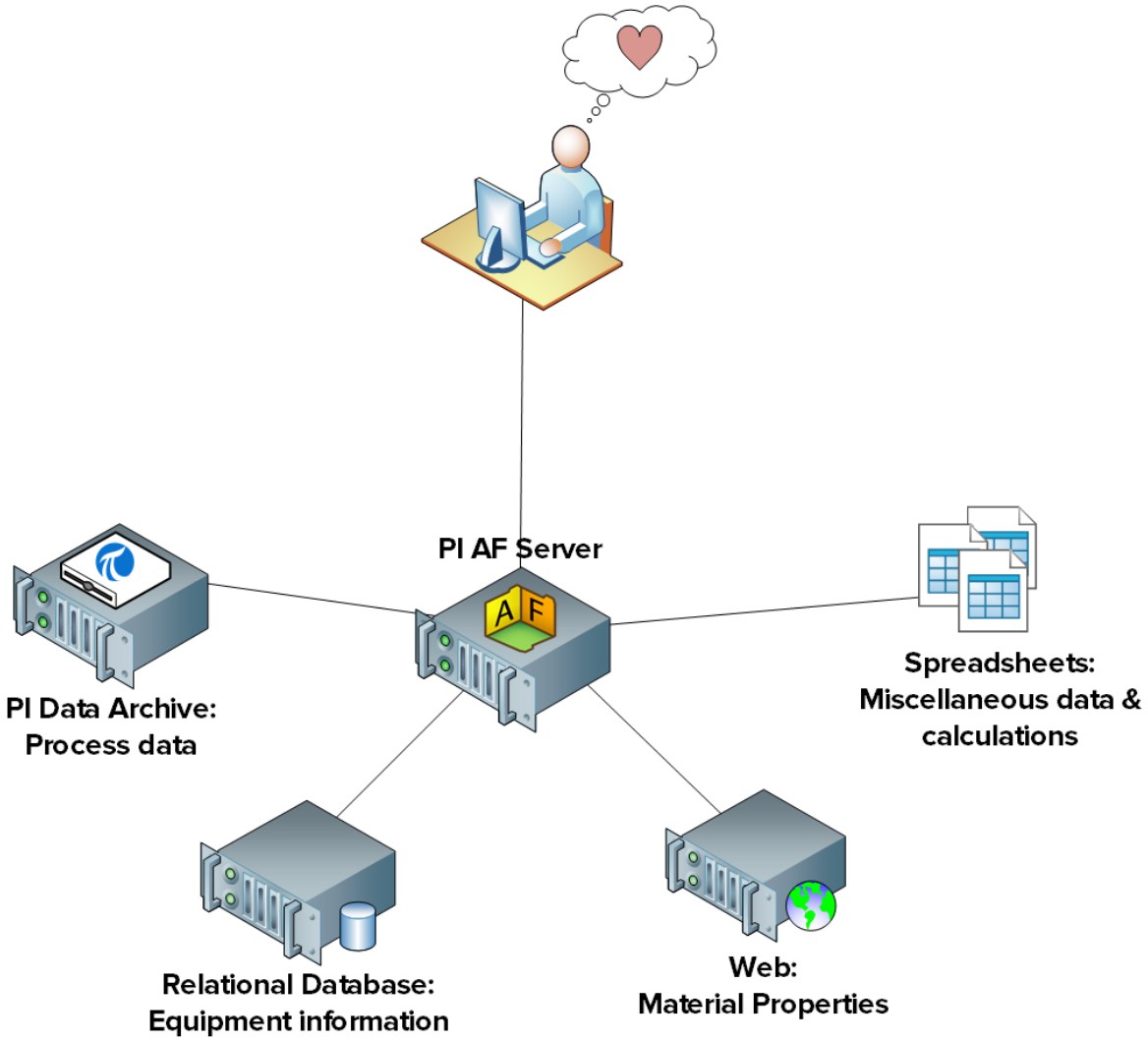
- Material properties

- Manufacturer information
- Geographical information

This static data might be sitting in various Excel spreadsheets, relational databases, websites, etc. Normally, it would be a pain for users to hunt down and gather this data in one place.



Once again, this is where the Asset Framework comes it. This static data can be made readily available to PI System users in an organized, human friendly view, by importing or linking the data to the Asset Framework.



Some users might not be satisfied with raw (static and time-series) data. Often, the raw data needs to be manipulated in order to make it meaningful. Here are a few examples:

- From raw voltage and current measurements, an engineer at a Transmission & Distribution company might want to view a list of all trips, disturbances and outages.
- From raw flowmeter and tank sensor data, a Chief Operations Officer might want to see the production totals for each of his plants, in real-time.
- From raw stack emission measurements, a plant manager might want to be notified automatically when environmental regulations are being broken.

Each of these requests can be met using PI System functionality build on top of the Asset Framework.

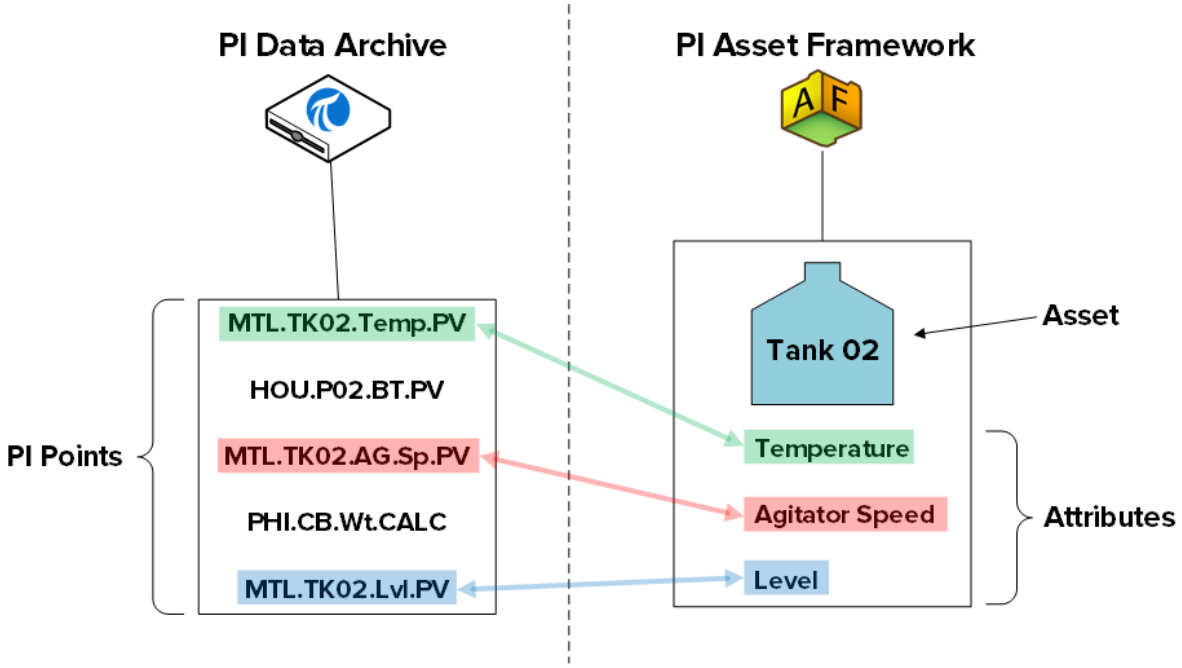
4.2 Define assets and attributes

As its name implies, the Asset Framework is made up of a framework of *assets*. So what is an asset?

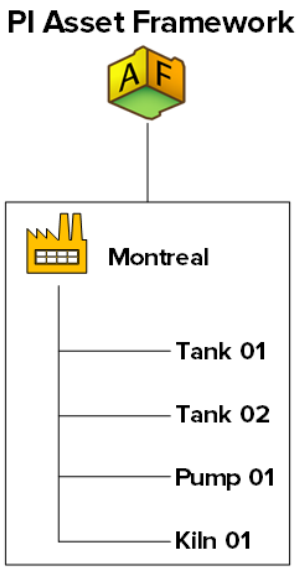
An asset is a logical or physical component of a process, under which data can be grouped. In the Data Archive, we have PI Points for which we are collecting streams of data. For example, we might be collecting:

- The temperature of a tank
- The speed of the tank's agitator
- The level of the tank

In the Data Archive, there is no relationship between these individual PI Points, even though, in reality, they are all related to the same piece of equipment. In the Asset Framework, these streams of data can be grouped under a "Tank" asset. These data streams are referred to as "attributes" of the tank asset, and are linked to the PI Point where the data is collected.



The assets in the Asset Framework are organized in a hierarchy. Continuing from our previous example, our tank asset might belong to one of our plants, the Montreal plant. We could therefore create an asset for the Montreal plant, and one of the Montreal plant's child assets would be the tank above. In this case, the "Montreal" asset is a logical component of our process.



So what benefit do we get from organizing our data into a hierarchy of assets?

1. **A human friendly view of the data**

   As we saw in the previous chapter, the data is now organized, which makes it easy for us to find, understand and consume.

2. **The relationships between assets provide additional opportunities**

   By defining the relationship between the Montreal asset and the equipment that belongs to the Montreal plant, we can extract additional information for the data. For example, if for each piece of equipment, we are collecting the power consumption, we can easily obtain the total power consumption of our Montreal plant.

3. **A way to compare similar assets**

   If I have multiple plant assets in my asset framework, I can compare the power consumption for each plant. If I create a report analyzing my tank performance, I can apply this same report to all of my tanks. As we will see later in this chapter, the use of templates when creating similar assets makes our work reusable and scalable.

In Asset Framework terminology, an “asset” and an “element” are synonyms.

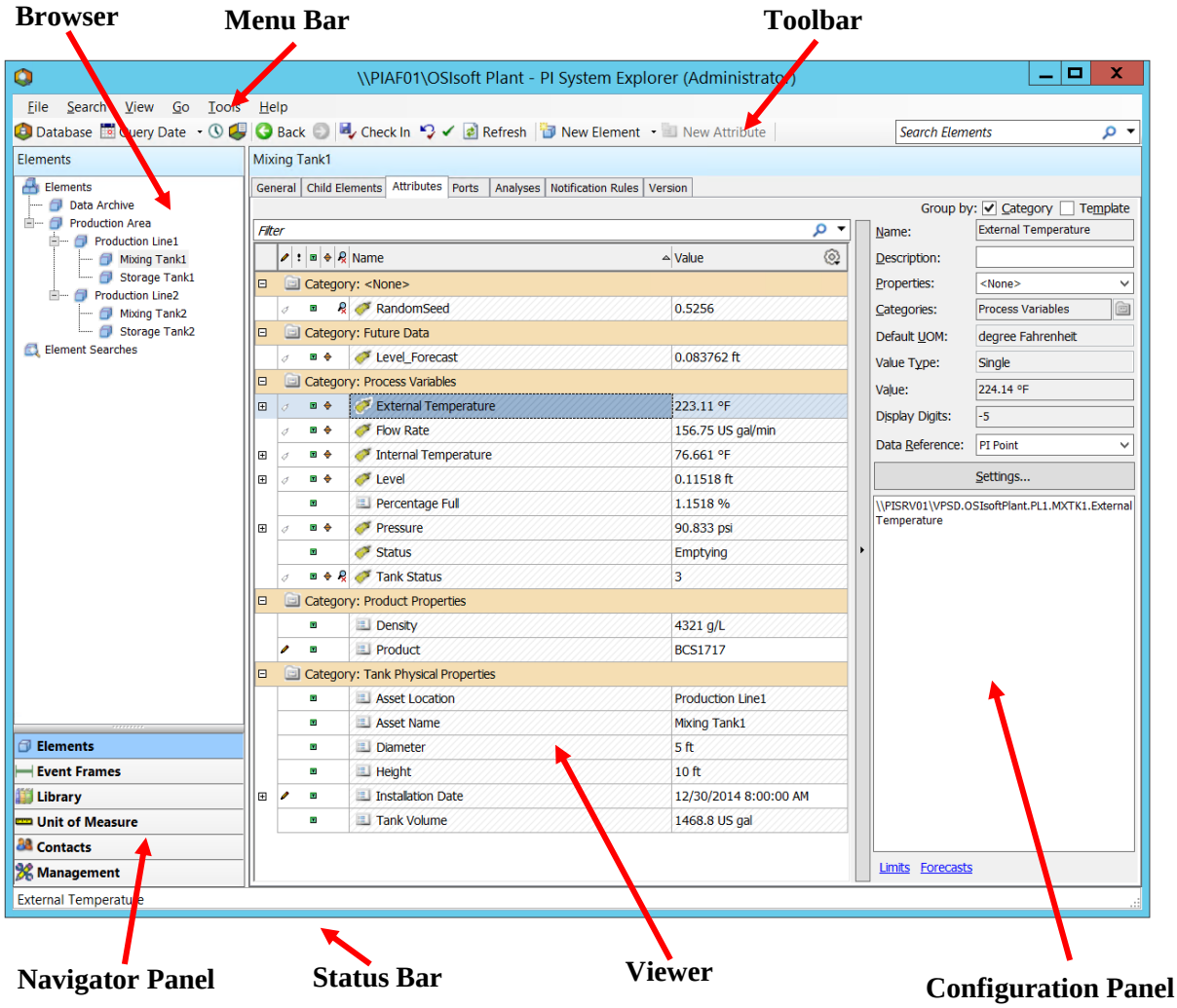
Note: Before the release of the Asset Framework, a Data Archive component called the “Module Database” (MDB) was used to create asset hierarchies under which PI Points could be organized. The MDB still exists, and continues to store configuration information for several PI System applications, such as the PI Interface Configuration Utility.

Some applications also continue to use the asset hierarchy built in the MDB, such as PI ACE and PI Batch. In order to continue using these applications, while also taking advantage of the Asset Framework, the Data Archive provides a mechanism for synchronizing the Module Database with the Asset Framework. For more information, consult the *PI MDB to AF 2015 Transition Guide*.

4.3 PI System Explorer

PI System Explorer, or sometimes referred to as PSE or AF Client, is the AF user interface that allows users to view and configure the asset hierarchy. It also has a rich range of features, making it the configuration and management tool for AF, PI Notifications, PI Event Frames and Asset Analytics.

The major components of the PSE are shown in the following:



Menu Bar / Toolbar

Use these bars for tasks such as opening/creating a database, searching for elements or contacts, applying and checking in changes, setting view options, and more. Menus and the Toolbar are context sensitive and will present different options depending on what section is selected from the Navigator Panel.

Navigator Panel

PI System objects are grouped into sections displayed in the Navigator Panel. Groups appearing by default include Elements, Event Frames, Library, and Unit of Measure, and Analyses. When the PI Notifications feature is installed, MyPI, Notifications, and Contacts also appear in the Navigator Panel.

Browser

Use the Browser to select the objects you want to work on and display in the Viewer panel. The Browser displays the PI System objects that have been added to the AF database, such as elements, templates, notifications, etc. Depending on the section selected from the Navigator Panel, the following will be available from the Browser:

- **Elements**: Elements (or assets) can be organized in several hierarchies. Users can drill down the element hierarchy to find the element of interest.
- **Event Frames**: An event frame is any event, defined by a start time, an end time and a context. Event Frames can represent downtime events, process and environmental excursions, batch processing steps or any other events important to your organization.
- **Library**: This is a collection of objects that can be re-used throughout the AF hierarchy. Types of objects that appear in the Library include Categories, Element Templates, Enumeration Sets, Reference Types, and Tables.
- **Units of Measure (UOM)**: The UOM database provides automatic handling of simple conversions between units of measure for attributes of the same UOM class.
- **Analyses**: This section provides a summary of all analyses (e.g. calculations) configured on the current AF database. It allows you to perform administrative tasks like starting, stopping and backfilling analyses.

Status Bar

Check the status bar after clicking an item in the Browser to see its status. For example, last modification time, if the object is checked out or if a notification is currently loading.

Configuration Panel

This panel is used to configure properties associated with attributes such as attribute references, UOM and values for static attributes.

Viewer

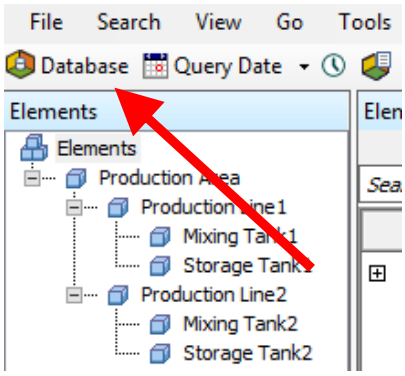
This is the primary work area. Use it to create and edit elements, attributes, templates, tables, contacts, notifications, analyses, and so forth. When configuring attributes through the Viewer, the Configuration Panel comes into view allowing you to make configuration changes.



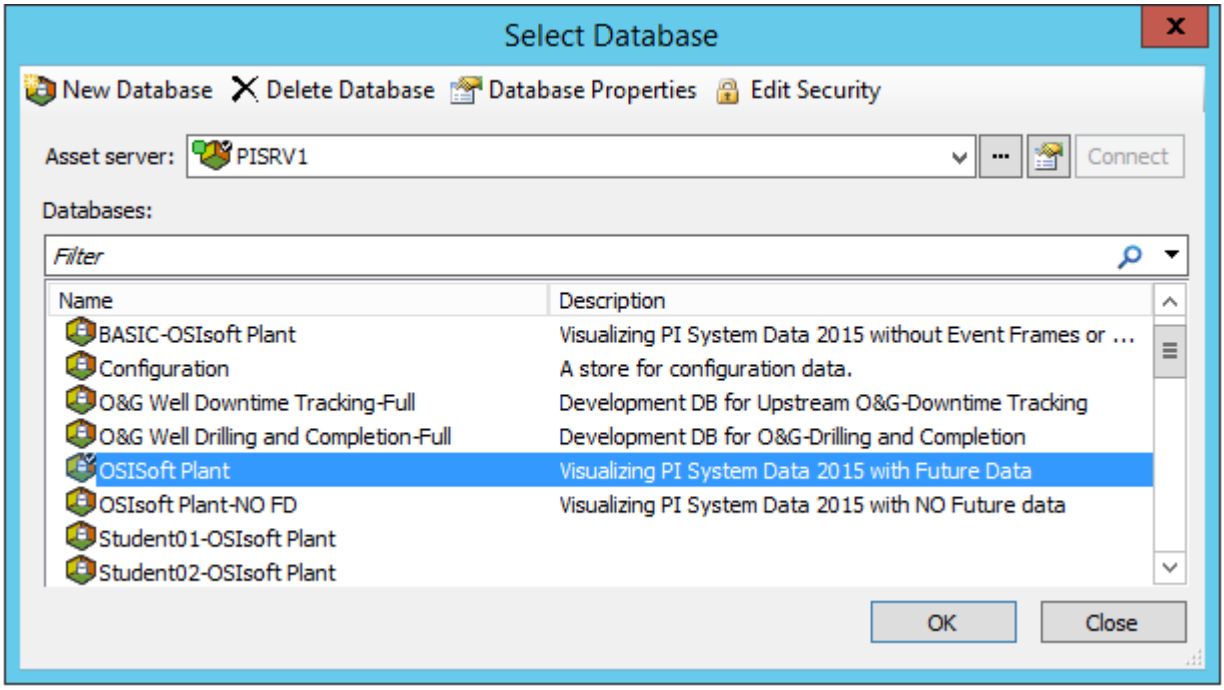
For more information see Using PI System Explorer (*PI System Explorer User Guide*).

4.3.1 Connecting to a AF Server to view the Element Hierarchy

AF stores the asset framework objects (elements, templates, and so on) in *AF databases*. You can have multiple AF databases in AF, although you can connect to only one at a time. In PSE, you can see which AF server you are connected to and its list of databases by selecting the Database button in the upper left corner.



The Select Database dialog box will appear and show you which AF server you are connected to (the drop down along the top).



Once connected to the desired AF server, you can select the database from associated databases list.

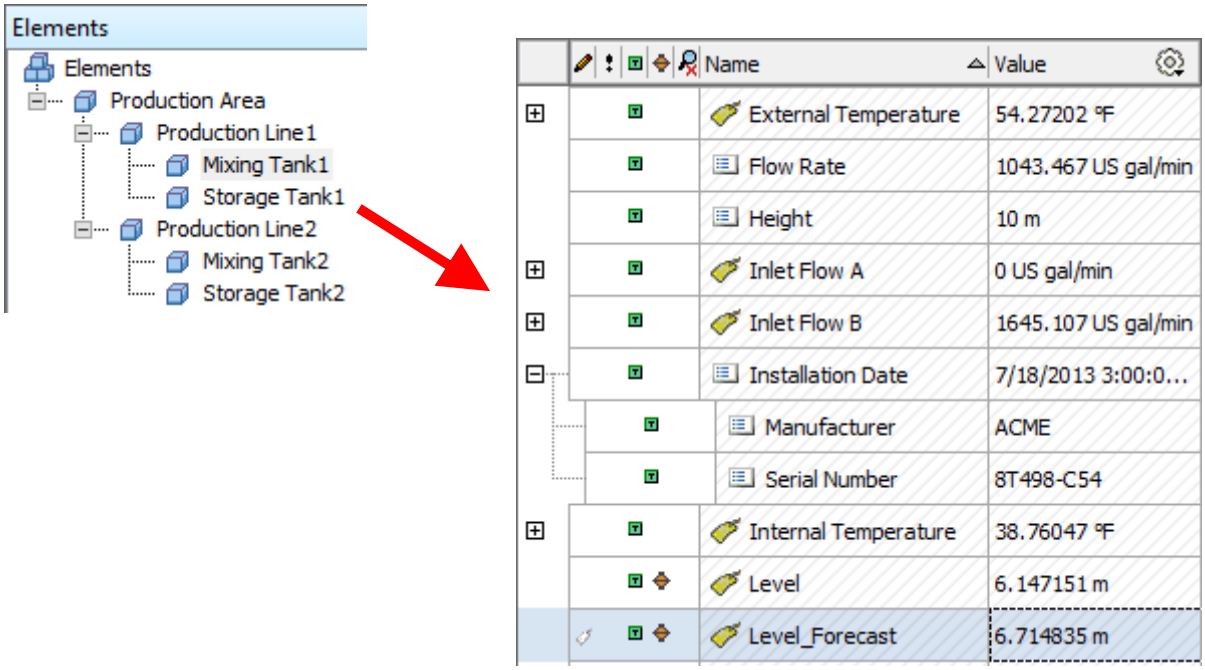
4.3.2 Directed Activity – Assets Defined



You are invited to watch what the instructor is doing or perform the same steps at the same time to explore the different concepts presented in this chapter or section.

Problem Description

A *data reference* is a mechanism that allows you to get an AF attribute value from external data. Using PSE, let's identify the types of data references available for AF attributes.



Step 1 : On PISRV01 or PIAF01, run the application "PI System Explorer"
Step 2 : Connect to the "OSIsoft Plant" database
Step 3 : In the "Browser" window on the left side, browse to Production Area > Production Line1 > Mixing Tank1
Step 4 : In the "Viewer" window in the middle, select the "Attributes" tab.
Step 5 : Find an attribute for each of the following data reference types:

a. <None> (Static) : _____
b. Formula : _____
c. PI Point : _____
d. String Builder : _____
e. Table Lookup : _____

4.4 Directed Activity – Organizing your PI Points into AF Assets



In this part of the class, you will perform a learning activity to explore the different concepts presented in this chapter or section. You may be invited to watch what the instructor is doing or perform the same steps at the same time. You may play a game or hold a quiz. Your instructor will have directions.

Activity Objectives

- Create a database in AF
- Create an element in AF
- Create a child element in AF
- Create PI Point data reference attributes in AF
- Convert an existing element to an element template

Problem Description

You have a set of pumps connected to your PI OPC Server and generating data. (***You have already configured the PI points for these pumps***).

You will create an AF hierarchy for these pumps, and link your PI points to the corresponding AF attributes.

Approach

Step 1. On PISRV01 or PIAF01, run PI System Explorer

Step 2. Go to the Select Database dialog.

Step 3. Select the database 'Pump Assets' and click 'OK'

Step 4. In the bottom left corner, make sure you have selected 'Elements'.

Step 5. On the 'Elements' symbol, right-click and choose 'New" > 'New Element'.

Step 6. Click 'Ok' on the dialog that appears.

Step 7. In the 'General' tab of the element you just selected, change the name to 'Pumps'.

Step 8. In the left pane, right-click on 'Pumps' and choose 'New Child Element'.

Step 9. In the 'General' tab of the new element, change the name to 'Pump1'.

Step 10. While clicked on 'Pump1' in the left pane, choose the 'Attributes' tab.

Step 11. Right-click on 'Pump1' in the left pane, and choose 'New' > 'New Attribute'.

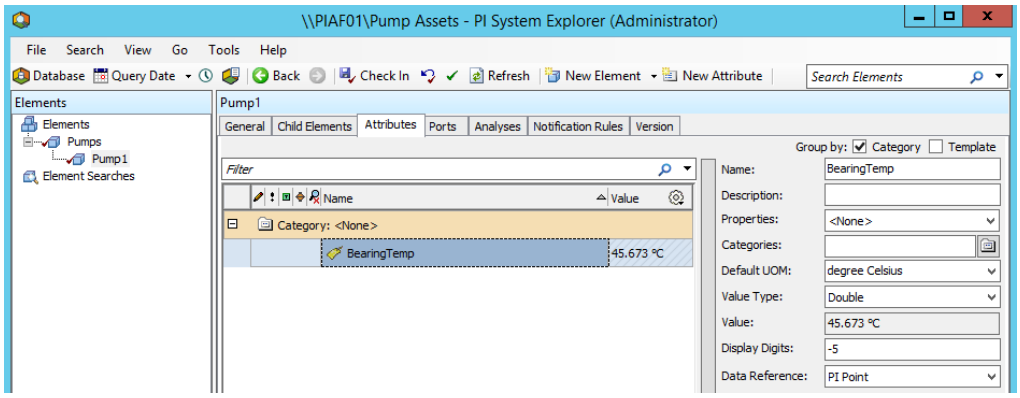
Step 12. Name the attribute: BearingTemp and click 'OK'

Step 13. In the right-pane, change the Default UOM to 'degree Celsius', and change the Data Reference to PI Point.

Step 14. Click the 'Settings' button below the data reference.

Step 15. Choose the magnifying glass next to Tag Name and do a search for your Pump1 Bearing Temperature point. Select this point and click 'Ok'.

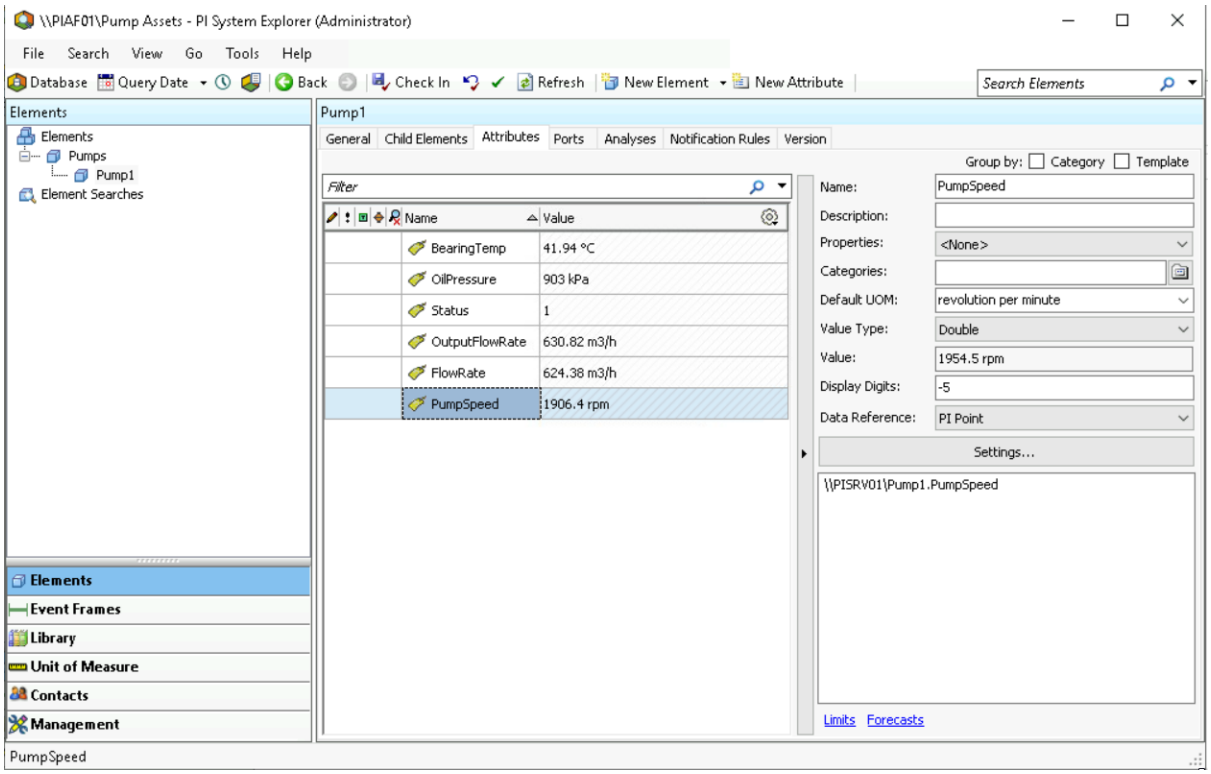
Step 16. Click the green check mark at the top to save your work, your current PI System Explorer screen should look similar to the figure below.



Step 17. Add five more attributes and name them: OilPressure (UOM=kPa), Status (UOM=none, pay attention to the pop-up when you select the tag Pump1.Status), OutputFlowRate (UOM=m^3/h), FlowRate (UOM=m^3/h) and PumpSpeed (UOM=rpm). Link these five attributes to their corresponding Pump1 points.

Step 18. When you are finished creating these elements, click on the 'Check-In' button above 

. On the dialog that appears choose 'Check-In'. Your structure will be similar to the figure below.



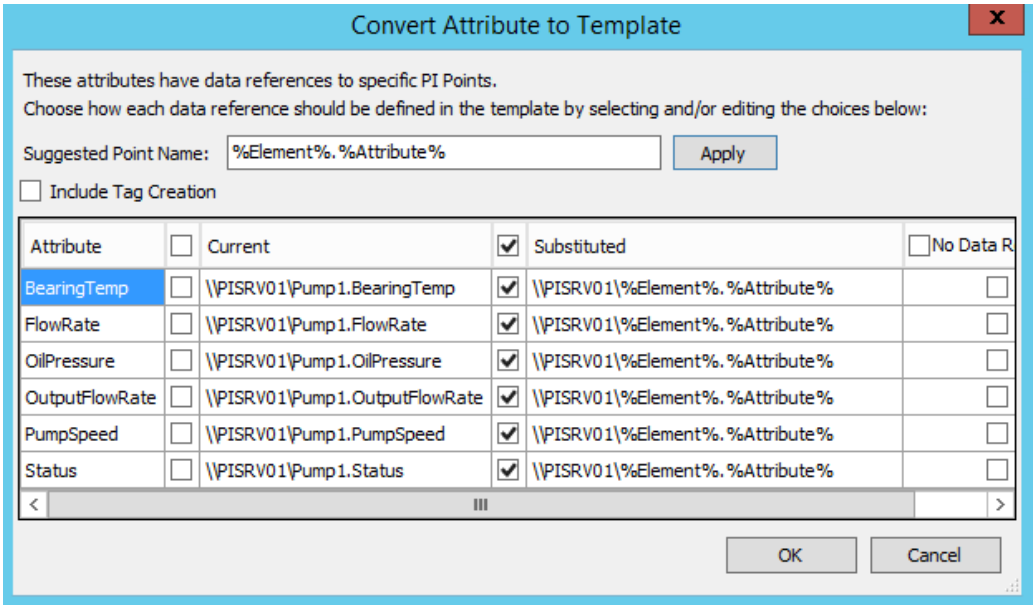
Step 19. Now that you have the Pump1 element complete, right-click on ‘Pump1’ and choose Convert > Convert to Template.

To make the template more applicable to other pumps, we are going to use substitution parameters within the point name reference.

Step 20. In the Substituted field, replace the substitution parameters according to the naming convention of your pump PI points. As an example, for the point Pump1.BearingTemp, I can use the fact that the name of my element is Pump1 and the name of my attribute is BearingTemp thus %Element%.%Attribute%.

When applied to an element, it will replace the ‘%Element% field with the value in the element name and the %Attribute% by the attribute name.

We will see this happen when we do some bulk creation of attributes.



4.5 Solo Exercise – Building Assets from Templates using PI Builder



This is a solo exercise that is designed to maximize learning in a specific topic area. Your instructor will have instructions and will coach you if you need assistance during the exercise.

Exercise Objectives

- Create an element using a pre-configured template in PI System Explorer
- Create an element using a pre-configured template in PI Builder

Problem Description

In a previous exercise, you created Pump1, and created a template for your pump assets. Now, let's see how this template can be used when creating the remaining pumps.

Approach

Method 1 – Using PI System Explorer

Step 1. Run PI System Explorer and go to your 'Pump Assets' database.

Step 2. In the Elements section, right click on 'Pumps' and choose 'New' >'New Child Element'.

Step 3. Choose the 'Pump Template' and click 'OK'.

Step 4. An element with the name "Pump2" should have been created. If not, change the name of the element to "Pump2"

Step 5. Verify the attributes of Pump2. They should have automatically been populate and linked to the 'Pump2' PI Points.

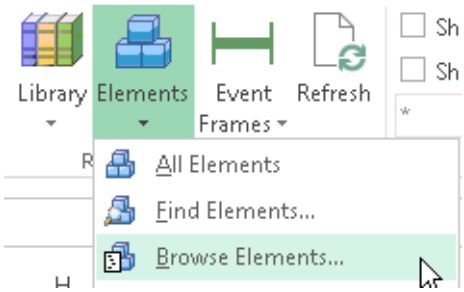
Method 2 – Using PI Builder

Step 6. Open Microsoft Excel on PISRV01, you should see a tab 'PI Builder'.

Step 7. In the top left, make sure your Database is pointing to your 'Pump Assets' database. If not, change it.

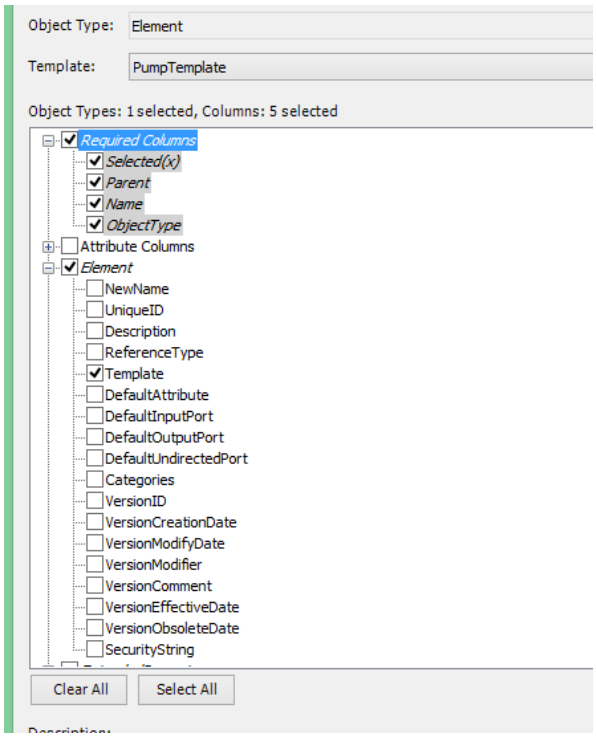


Step 8. Click on the arrow below 'Elements' in the 'Retrieve' section and choose 'Browse Elements'.



Step 9. Choose ‘Pump2’.

Step 10. Clear all the checks and then select Template, under Element. Then select ‘OK’.



Step 11. Copy the imported rows 2 times so you have 3 rows populated.

Step 12. In the first row, change ‘Pump2’ to ‘Pump3’, in the second row change ‘Pump2’ to ‘Pump4’, and in the third row change ‘Pump2’ to ‘Pump5’. Your resulting spreadsheet should look like this:

	A	B	C	D	E	F
1	**Selected(x)**	**Parent**	**Name**	**ObjectType**	**Template**	
2	x	Pumps	Pump3	Element	Pump Template	
3	x	Pumps	Pump4	Element	Pump Template	
4	x	Pumps	Pump5	Element	Pump Template	
5						

Step 13. Make sure the ‘Selected(x)’ column has an ‘x’ for all three rows.

Step 14. Choose ‘Publish’ in the ‘Build’ section.

Step 15. Change the ‘Publish Options, Edit Mode:’ to ‘Create Only’ and click ‘OK’.

Step 16. Go back to PI System Explorer. Click the refresh button. Verify that all three pumps were created with the correct attributes.

4.6 Directed Activity – Take advantage of your asset model in PI Vision



In this part of the class, you will perform a learning activity to explore the different concepts presented in this chapter or section. You may be invited to watch what the instructor is doing or perform the same steps at the same time. You may play a game or hold a quiz. Your instructor will have directions.

Activity Objectives

- Create a pump display using PI Vision, which takes advantage of the pump database you built in the previous exercise

Approach

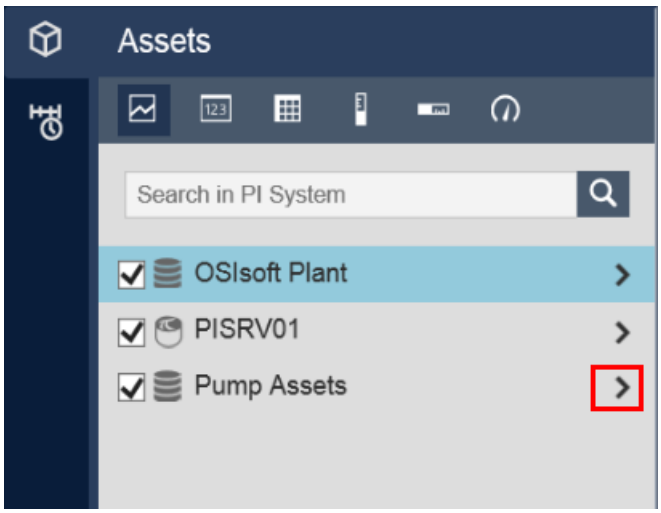
Now that your pump data has been organized in the Asset Framework, you can create a display which takes advantage of the contextual data you have provided.

Step 1 : From PISRV01, open the web browser “Edge”.

Step 2 : Click on the “PI Vision” bookmark.

Step 3 : On the PI Vision homepage, select the “New Display” button in the upper-right hand corner.

Step 4 : In the asset pane in the top-left hand corner, Drill down into your “Pump assets” database using the arrow



Step 5 : Drill down into Pump 1.

Step 6 : Make sure the trend symbol is selected . Drag and drop both the Bearing Temperature and Pump Speed attributes.

Step 7 : Select the radial gauge symbol . Drag and drop the Oil Pressure.

Step 8 : Select the value symbol . Drag and drop the Status.

Step 9 : Save your display with the name “Pump Overview”

Step 10 : At the top of the display, you will find a drop-down menu with “Pump 1”. Select a different pump from the menu. Notice how the data on your display changes to reflect the pump selected in the drop-down menu.

4.7 Components of PI Asset Framework

The Asset Framework is made up of the following software components:

- The AF Application Service (a Windows Service)
- The PIFD Database (a Microsoft SQL Server database)

These components do not need to be installed on the same computer. The PIFD database must be installed on a computer which hosts a Microsoft SQL Server. SQL Server Express editions are supported.



For more information on supported SQL Server versions, see "SQL Server Requirements for AF Server" in *AF Release Notes*

Applications communicate with the Asset Framework by making requests to the AF Application Service using AFSDK. The AF Application Service then retrieves information stored in the PIFD Database, and sends it back to the AF Client.

Given that PIFD is a SQL Database, the majority of the maintenance tasks are typical SQL Server Administrator tasks. Most large enterprises will have database administrators, and it will be their job to manage the PIFD database. However, for smaller PI System implementations, it is possible that no PI System administrators have prior SQL knowledge. There are SQL Server administration classes offered by Microsoft and a very large number of certified organizations.

A good place to begin would be by visiting the ***Microsoft SQL Server Learning Center*** at:

https://www.microsoft.com/en-ca/server-cloud/support/learning-center/learning-center.aspx

In the rest of this chapter, we will cover important Asset Framework administration tasks.



Tip

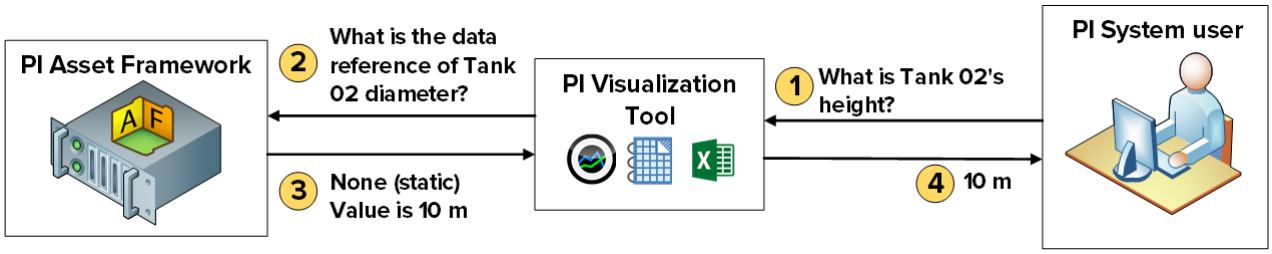
Do not make changes to the PIFD database manually using any sort of SQL management or scripting tool.

4.8 Data Flow when using the Asset Framework

When a PI System User is viewing data through the use of AF attributes, the flow of data through the system will depend on the type of data reference for the attribute. Here are the different scenarios:

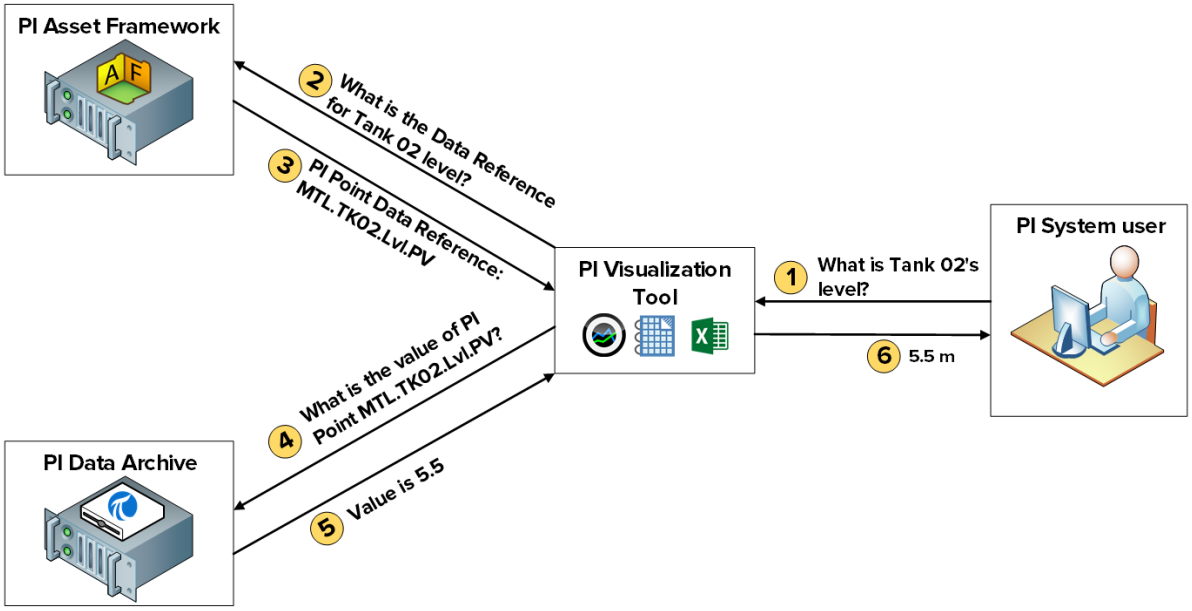
A. Static data reference

For data references of type “None”, the data is stored directly in the Asset Framework’s database. Therefore, a single connection is made to the Asset Framework.



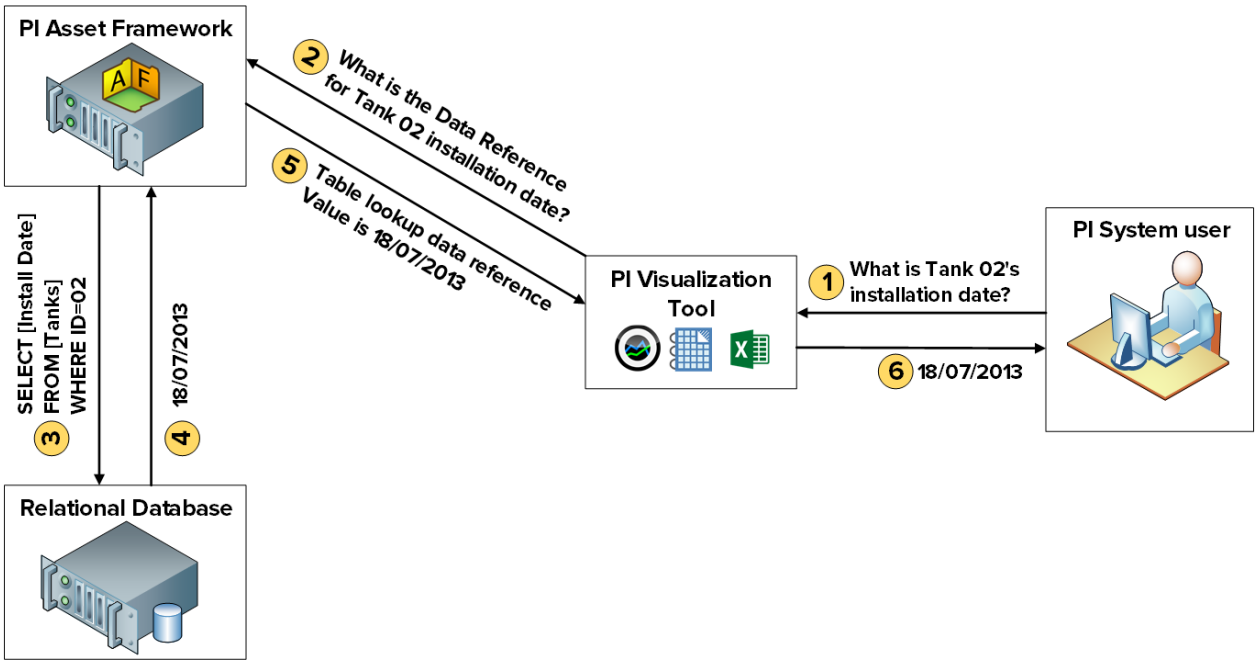
B. PI Point Data Reference

For data references of type “PI Point”, the Asset Framework stores the Data Archive Server and PI Point name for the attribute. After retrieving the PI Point Data Reference, the PI Visualization tool will make a request directly to the Data Archive for the specific PI Point. Hence, the AF Application Service does not need read access to the data on the Data Archive. We will continue our discussion on security in the next chapter.



C. **Table lookup data reference** (when the table is stored in a separate relational database)

For data references of type “Table Lookup”, the data is stored in a table. If this table is imported into the Asset Framework, a single connection is made (see scenario A). However, the table might be linked to a separate database. If this is the case, the AF Application Service will make the request to this external database directly, and return the result to the PI Visualization Tool. Thus, the AF Application Service needs read access to the data on the external database. If you also want to restrict access to the data based on the end-users credentials, Kerberos Delegation must also be configured. We will continue our discussion on security in the next chapter.



4.9 AF architecture

The Data Archive, the AF Application Service and the SQL Server (hosting the PIFD database) can all be installed on the same server, or on different servers. The choice of architecture will depend primarily on the size of the implementation. Below are a few examples of common architectures:

A. Small PI System

For systems with few assets (10,000 or less) and low-to-moderate number of PI Points (25,000 or fewer), the recommended architecture is a single server hosting the Data Archive, the AF Application service and a SQL Server Express.

B. Larger, higher performance PI Systems

For systems with more than 10,000 assets, and a moderate-to-high number of PI Points, it is recommended that:

- The SQL Server should be installed on a separate computer from Data Archive, and a Standard or Enterprise edition should be used.
- The AF application service should be installed on either the Data Archive or SQL Server computer.
- High Availability options for the Data Archive and AF should be considered, for scalability

C. Distributed, highly available PI System

For distributed systems with large workloads and point counts, and with multiple Data Archive servers or Data Archive collectives that link to a central AF database, AVEVA recommends that you install Data Archive collectives, chose a high availability option for AF, and use Microsoft SQL Server on separate, redundant computers to achieve the best level of performance and scalability.

4.10 Manage Asset Framework Backups

In the previous chapter we discussed how to manage Data Archive backups. The strategy for managing Asset Framework backups is the same. AVEVA recommends that a daily backup of the Asset Framework be performed, and that this backup be copied to an external storage device.

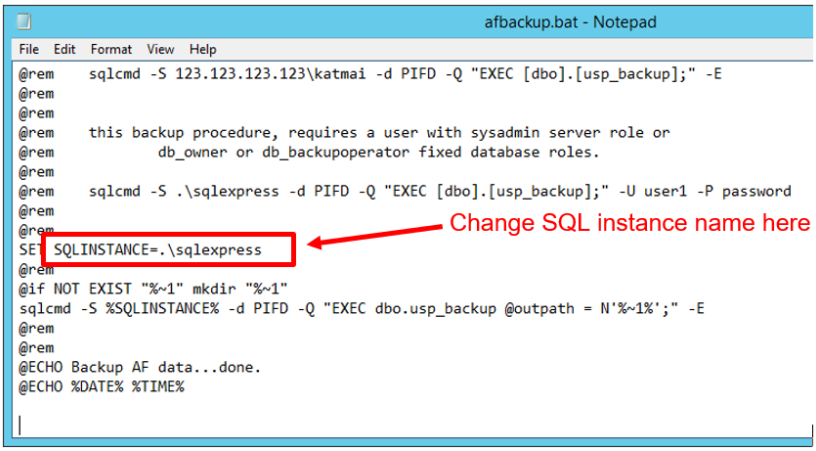
4.10.1 How the AF Backup works

All the changes made to the Asset Framework since the original installation are contained within the PIFD database. Therefore, the PIFD database is the only file which needs to be backed up. The specific backup mechanism will depend on the AF architecture, as well as the SQL Server edition.

A. SQL Server Express edition

For SQL Server Express editions, the PIFD database is backed up by a script called afbackup.bat (located under the %pihome64%\AF\sql directory).

The default name of a SQL Server Express instance is .\sqlexpress. If your SQL Server instance is named differently, you will have to manually edit the afbackup.bat script. Under the "SET SQLINSTANCE" line, enter the name of the instance (e.g. SET SQLINSTANCE=.\mysqlserver)

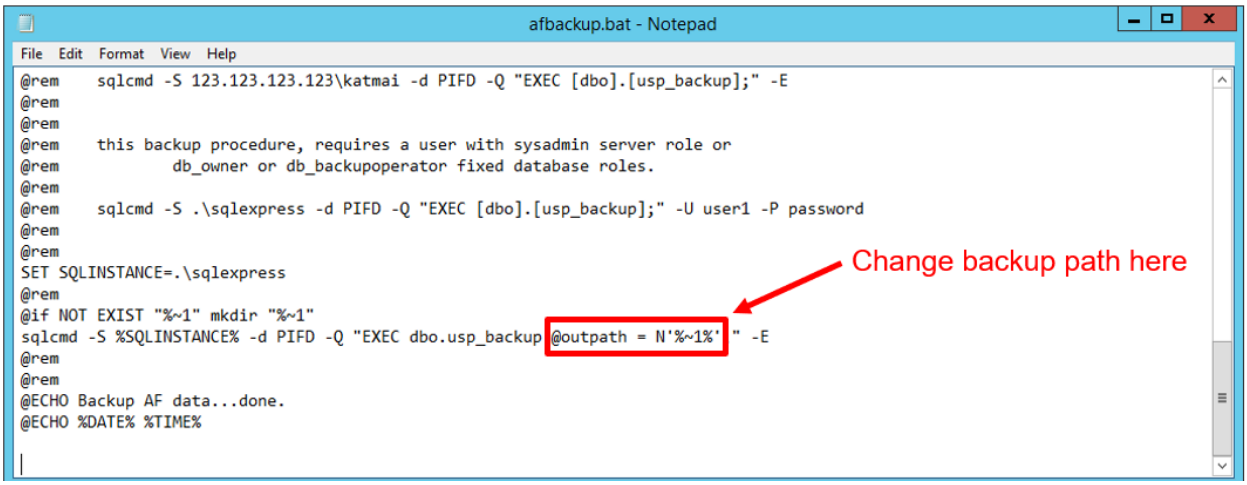


i. SQL Server Express is installed on the Data Archive

If the Data Archive is installed on the same computer as the SQL Server, the Data Archive backup script, pibackup.bat, will call the afbackup.bat script. The PIFD backup will be created in the same directory as the Data Archive backup.

ii. SQL Server Express is installed on a separate computer

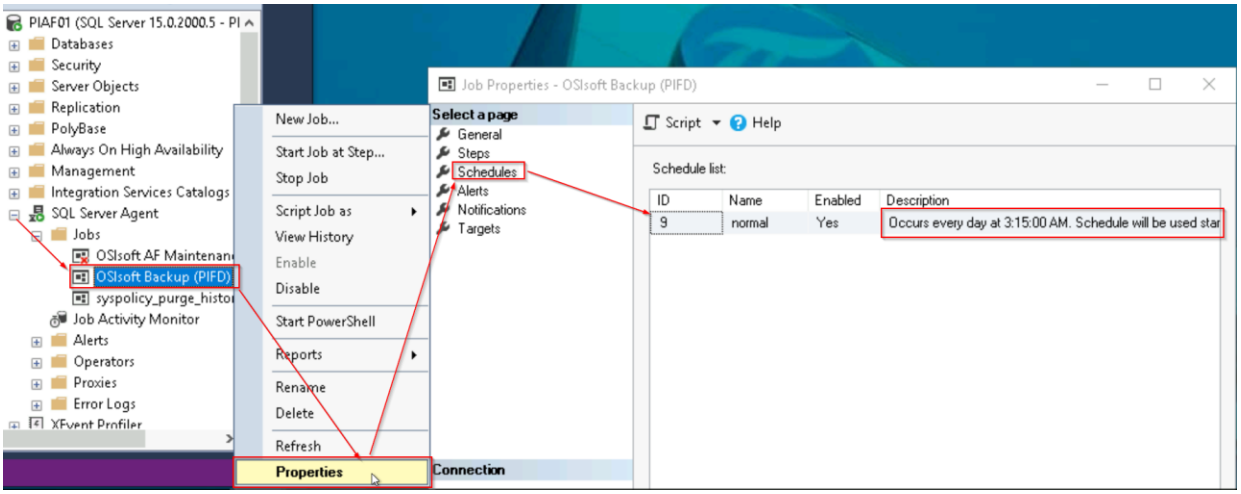
If the SQL Server is not installed on the same computer as the Data Archive, a scheduled task must be created manually on the AF Server to run the afbackup.bat script. The afbackup.bat script must also be modified in order to backup the PIFD database to a separate physical volume (e.g. sqlcmd -S %SQLINSTANCE% -d PIFD -Q "EXEC dbo.usp_backup @outpath = N'D:\PIBackup\AF\';" –E)



B. Non-Express editions of the SQL Server

Non-Express editions of the SQL Server come with the SQL Server Agent, a Windows Service that executes scheduled administration tasks, or “jobs”. In this case, AF will automatically install and schedule a nightly backup job at 3:15 AM.

The nightly backup job should be modified to change the backup directory to a separate physical volume. The default path is the Backup folder where SQL Server is installed



4.10.2 (Optional) Solo Exercise – Manage the AF Backup



This is a solo exercise that is designed to maximize learning in a specific topic area. Your instructor will have instructions and will coach you if you need assistance during the exercise.

Activity Objectives

- Review the AF Backup configuration
- Change the default directory of the AF Backup
- Run an AF Backup

Approach

Step 1. On PIAF01, create a new folder in C:\\PIBackup\AF and give

'NT Service\SQLSERVERAgent' write access to it.

Step 2. Run the program "SQL Server Management Studio"

Step 3. Connect to SQL Server instance PIAF01

Step 4. This is a SQL Server Enterprise edition. Therefore, the backup is being performed by the SQL Server Agent as a job. In the object explorer, expand "SQL Server Agent" then "Jobs"

Step 5. Right click on the job "OSIsoft Backup (PIFD)" and select "Properties".

Step 6. On the General tab, verify the last time the job was executed: _____

Step 7. On the Steps tab, select the "Backup" step and click "Edit"

Step 8. Change the output path of the PIFD backup. We want to backup this database to C:\PIBackup\AF\

Hint: You will need to change the end of the command to

@outpath = N'C:\PIBackup\AF\';

Step 9. Close the Job Properties window.

Step 10. Right click on the job "OSIsoft Backup (PIFD)" and select "Start Job at Step…"

Step 11. Once the backup job completes successfully, verify that the backup file: $$PIFD.bak exists in the directory.

4.11 Asset Analytics Performance

One of the most important components of a mature PI System is Asset Analytics. Most of the value should be coming from Asset Analytics since we prefer that users check their assets only when required instead of continuous monitoring. This requires a reliable Asset Analytics engine that will be informing the users about any abnormality.

Due to its value, Asset Analytics is used for wide range of use cases. Sometimes, this leads to performance related issues on Asset Analytics that impacts everyone who relies on it.

A PI System Administrator should

- Ensure that users create their analytics according to best practices.
- Ensure that only well-trained people can create analytics.
- Monitor the Asset Analytics performance to resolve issues proactively.
- Evaluate the hardware of the Asset Analytics and increase it when required.
- Troubleshoot performance issues in a timely and efficient manner.

In this chapter, we will try to understand the building blocks of reliable Asset Analytics and share some resources that will help in achieving the goals above.

4.11.1 Best practices for Asset Analytics

The following factors should be considered to optimize performance:

- Analysis organization
  - Number of analyses
- Input attributes
  - Types of input data
  - Data density
  - Data patterns
- Output attributes
- Analysis scheduling
- Expensive functions & dependencies
- Configuration parameters

Analysis organization

The PI AF Server is optimized to use element templates and analysis templates. Analyses should always be based on templates for ease of use, scalability, and optimal performance.

It is better to have many simple analyses rather than a few complex analyses.

Do not include multiple, independent calculations within the same analysis. Instead, we recommend creating elements for each asset and having separate sets of analyses on each asset. This ensures that only the relevant inputs trigger the analysis and that expressions aren't needlessly evaluated.

Input attributes

Input attributes should ideally be PI Point Data References, especially if the calculation uses historical data. Using attributes that are Table Lookups, Formulas, String Builder, and Analysis Data Reference types should be avoided if possible because they must be resolved by the Analysis Service and will slow down the calculations.

Inputs to Table Lookups, Formulas, String Builder, or Analysis data reference types should ideally not be other Table Lookups, Formulas, String Builder, or Analysis data reference types. Long input dependency chains are extremely resource-intensive because the Analysis Service must resolve the entire dependency chain before performing the calculations to determine the value of the input.

Data Density

For attributes using the PI Point data reference, it's extremely important to properly configure exception and compression on PI Points according to our recommendations (What are recommended Exception and Compression settings). More details:

- Poorly configured exception settings can lead to an unnecessary triggering of Event-Triggered analyses, which places an unnecessary load on the service. This can be exacerbated by dependent analyses also being unnecessarily triggered.
- Poorly configured compression settings lead to unnecessary data being stored in the PI Data Archive. The denser the data, the slower the data retrieval will be for clients such as the PI Analysis Service.

Output attributes

In most cases, it is best for performance to save the output history of the analysis to a PI tag. However, there are some scenarios where history is not needed. If this is the case, map the output to an attribute that uses an Analysis Data Reference.

Do not use an attribute with a None Data Reference (i.e., a static attribute) as the output for an analysis; this can cause performance issues on the SQL server.

Scheduling

Schedule analyses to evaluate only as often as necessary. The goal should be to reduce the number of calculations required while still getting the most up-to-date data.

For example, if your input data only updates once a day, don't schedule the analysis to run periodically once a minute. It would be better to schedule it once a day or have it be event-triggered. Conversely, if your data updates every second, but you only need a result once an hour, use an hourly periodic schedule rather than an event-triggered schedule.

If your system has many periodic analyses with the same schedule, add an offset to a subset of the analyses to help distribute the load across the processors.
See a detailed example in this article.

Expensive Functions and Dependencies

We recommended against having summary functions with large time ranges within analyses that trigger very often (e.g. high-frequency event-triggered analyses), as these functions will take a relatively long time to evaluate. A general rule of thumb to aim for is to have analyses evaluate within 100ms, though there are certainly reasonable exceptions.

For example, if there is an event-triggered EventCount function that queries data over the last day (*-1d to *), we recommend moving the EventCount function to a separate analysis. This separate analysis should be periodic (e.g. every 10 minutes) and output to a PI Point attribute. The original event-triggered analysis can then be configured to use that output attribute instead of the EventCount function itself. This effectively decouples the EventCount function from the event-triggered analysis while still maintaining high fidelity of data for calculations.

Configuration parameters

The global PI Analysis Service configuration parameters are automatically set at installation based on the machine's hardware resources, but under certain circumstances, they can be tweaked to improve service performance.

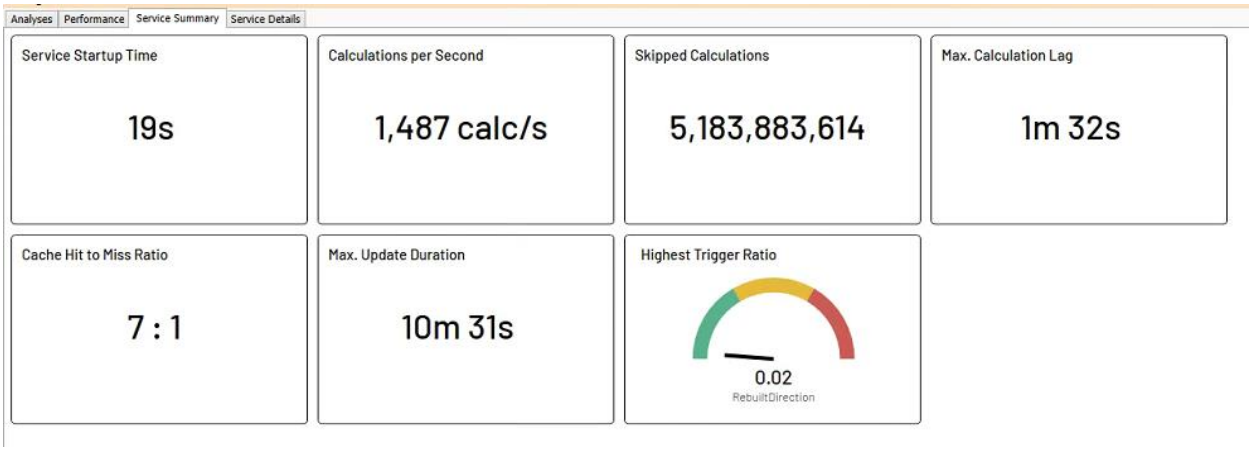
To access the global configuration parameters, open PI System Explorer > Management tab > and right-click the blank space under "Pending Operations" > Edit Analysis Service Configuration. More information about what each parameter controls, and guidance on what value to set these parameters, can be found here: [Analysis service configuration].

For more details on Asset Analytics Best Practices, please go to [this article].

4.11.2 Monitoring PI Asset Analytics

The best way of monitoring a component of the PI System is to use PI PerfMon interface. We will discuss this later.

In addition to this, PI System Explorer provides an overview of the PI Asset Analytics performance under Management Tab section.



These are general KPIs that are available on this page. More can be found in analysis statistics.

Service Startup Time	The amount of time between the initial startup of the AF Analysis Service and when the calculation of analytics began.
Calculations per Second	The average number of analytics calculations evaluated per second.
Skipped Calculations	The total number of calculations skipped by the AF Analysis Service since its startup. This number can be high simply because the last startup was long time ago. So, we usually monitor the rate of growth rather than the number itself.
Max. Calculation Lag	The latency of the worst-performing analysis since the startup of the AF Analysis Service. This metric includes a built-in wait time of 5 seconds.
Cache Hit to Miss Ratio	The ratio of the number of AF analysis inputs retrieved from the cache to the number of analysis inputs not in the cache.
Max. Update Duration	The time taken to update the cache for any AF analysis input.
Highest Trigger Ratio	The AF analysis template with the highest average trigger ratio. This ratio should be less than 1. This ratio is defined as the template's average elapsed time divided by its average trigger time.

4.11.3 Troubleshooting PI Asset Analytics Performance

But when you face a performance issue, finding the root cause can be difficult as there are many potential causes. In these cases, we recommend that you investigate with Technical Support. We usually take the Analysis Statistics, CPU/RAM usage on PI Data Archive and PI Analytics servers, logs from both applications to do a detailed investigation.

In short, there are three main solution paths to Performance related issues:

- Improve analyses for better performance.

We recommend this when we find that the issue is caused by analyses that do not follow the best practices given earlier. This might require reconfiguring hundreds of analyses and involve several stakeholders. Sometimes this is not possible in short term, so we give other suggestions.

- Change Analysis Service Configurations to use the existing hardware more efficiently.

There are some limitations on Analysis Service that prevent it from using too much resources. Sometimes this means that we are not using the resources available on the server. For example, we have an issue where Analysis Service is not able to keep up with number of calculations, but CPU usage is low on the server. In this case, we change the configuration to allow the service to use more CPU.

- Increase hardware on the servers.

This is the last resort when we have confirmed that the analyses are well configured, and the machine resources are fully utilized.

4.12 Troubleshooting Exercise



The following group exercise is intended to reinforce the concepts we have learned so far. The instructor will act as an end user with an issue, and you will resolve their issue as a group.

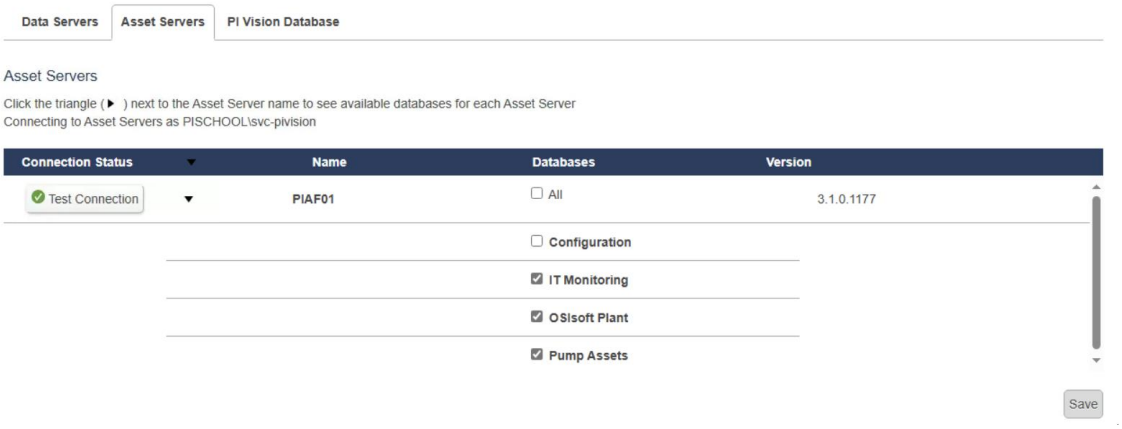
5. PI Vision Management

5.1 Administration

PI Vision administration page can be accessed by adding /admin to the normal PI Vision URL. In our environment, the link would be https://pisrv01.pischool.int/PIVision/Admin/

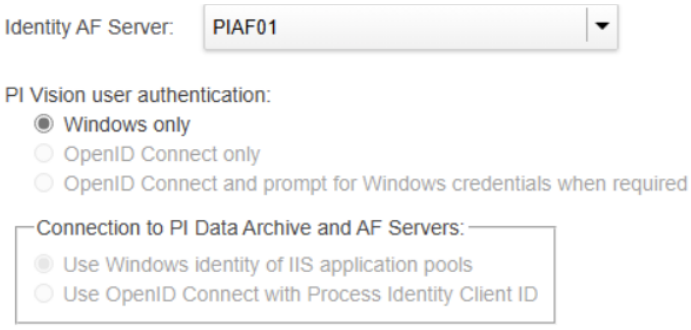
5.1.1 Configuration

This page is used to configure PI Data Archive and PI AF Servers that PI Vision has access to. The list is taken from PI System Explorer, but administrators can restrict access to some of these Servers/AF Databases.



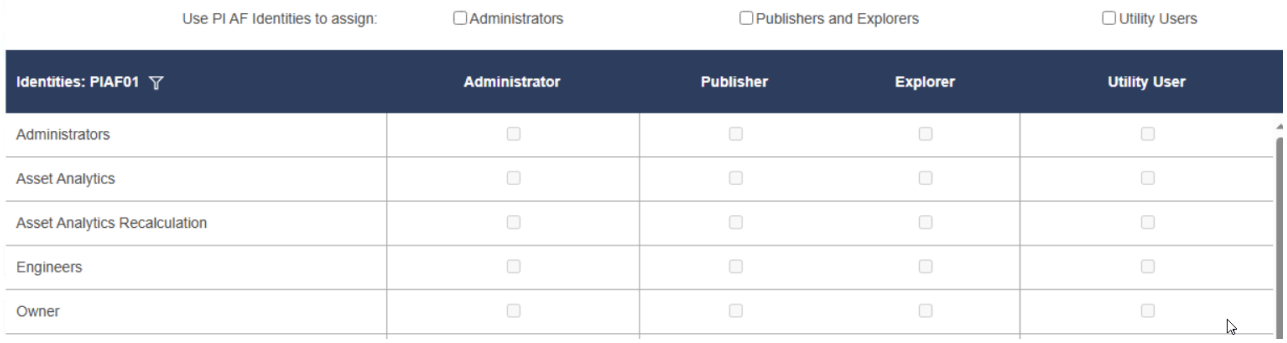
5.1.2 Security

Administrators can control the authentication method used in PI Vision from Identity page.



Under “User Access Levels”, administrators can set access to PI Vision displays. If nothing is selected on this page, PI Vision uses a local group “PI Vision Users’ to give access to PI Vision as Publisher. By default, any authenticated user will be in that group. Publishers can create and share displays, while Explorers can create displays only for themselves and see displays of others.

It is also possible to give access using AF Identities (from AF Server selected in Identity page).



5.1.3 Display Defaults

This page is used to configure some of the default settings when creating displays.

5.1.4 Reports

Administrators can generate reports on PI Vision usage, content within displays and list of users from this page. To see the current active PI Vision sessions, they can use Display Sessions page.

5.2 Backups

PI Vision uses an SQL Server to store its data. So, administrators need to take regular backups of this database. There is no built-in backup task but SQL Server Agent or a Task Scheduler task can be used to automate this. A simple batch file to take the backup would look like this:

```
sqlcmd -s .\sqlexpress -i "D:\SQLBackup\SQL\PIVision\PIVisionBackup.sql"
```

Where sql file would hold a backup query like

```
declare @DestFile varchar(128) = N'E:\Backups\PIVisionBackup.bak';
backup database PIVision
to disk = @DestFile;
go
```

Note that this will add new backup on top of existing backup so the size of the backup file might increase after some time.

6. PI System Security Management

Objectives:

- Describe which ports are used for communication in the PI System
- Describe which firewall rules must be in place
- Enable Windows firewall and create firewall rules
- Explain the difference between Authentication and Authorization
- Explain how security works on the Data Archive
- Describe the three security protocols: PI Mappings, PI Trusts, Explicit login
- Describe the possible security protocols for PI API and PI SDK connections
- Create PI Identity for PI Interfaces and PI Buffers
- Configure PI database security
- Configure PI point security
- Tighten security an existing interface connection
- Create a PI Mapping for a Windows group to give minimum required security
- Explain how security works on the AF Server
- Create an AF mapping for a Windows group to give minimum required security

6.1 Securing a PI System

In the context of the PI System, “Security” has multiple objectives:

- Add to the overall reliability and resiliency of the system
- Protecting PI System data and services from malicious attacks
- Limiting user access based on individual user needs

PI System Security is best implemented in a corporate network-secured computing environment. This usually includes:

- Domain security for users, directories, and applications
- Router security including router-based firewalls
- Antivirus programs and regular operating system patches
- Controlled access by remote parties (VPN)

First and foremost, AVEVA recommends hardening the platform using the Windows operating system and network environment. Administrators can do so effectively by leveraging industry standard profiles and built-in capabilities (e.g. AppLocker, Windows Advanced Firewall, etc).

Windows Integrated Security (WIS) brings improvements in authentication and encryption of data throughout the entire PI System. To take advantage of the security features built into the PI System platform, applications must authenticate with WIS. WIS is the strongest authentication mechanism available for the Data Archive. Additionally, transport security is automatically enabled to protect the confidentiality and integrity of data with the latest versions. The ideal Data Archive deployment has

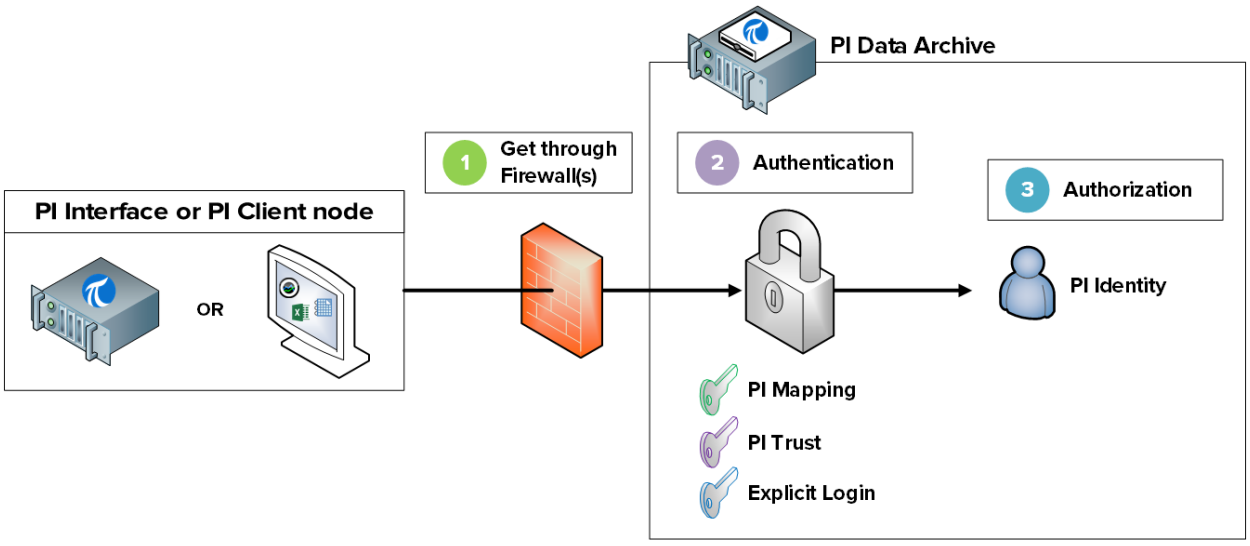
all client applications and services authenticating with WIS, so that all other authentication protocols can be disabled.

Antivirus software should be used on the PI System components. However, the archives and data files should be removed from the list of files scanned. Additionally, AVEVA recommends leveraging application whitelisting as a more effective measure. These strategies will be discussed later in this chapter

6.1.1 Accessing a secured PI System

To access a secure Data Archive, a connection must:

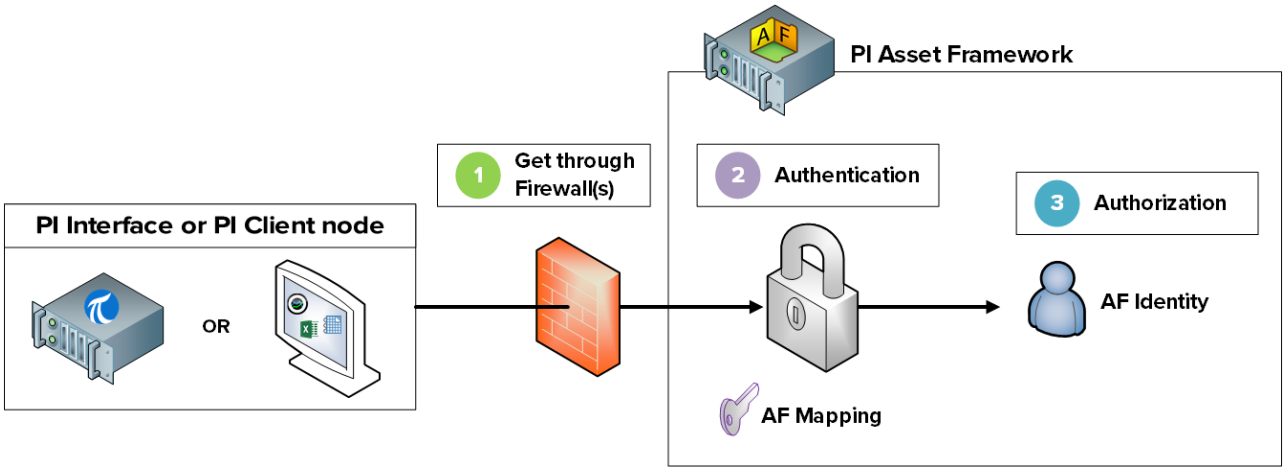
1. Contact the server over a network. The most common barrier to network communication are the firewalls, which guard the server.
2. Authenticate itself through either a PI Mapping, a PI Trust or Explicit Login
3. Receive the proper authorization through its PI Identity



In order to access a secure Asset Framework, a connection must:

1. Contact the server over a network. The most common barrier to network communication are the firewalls, which guard the server.
2. Authenticate itself through AF Mapping
3. Receive the proper authorization through its AF Identity

Note: AF Mappings were introduced in AF version 2015. Previously, authorization was assigned directly on active directory users and groups.



We will explore these steps in more detail in the next sections

6.2 Describe the ports used for PI System communication

As discussed in chapter 1, on a computer network, a port is an endpoint of communication. It is used by the Operating System to direct incoming packets of data to the correct computer program or service.

When PI System clients communicate over a network, the data is sent over specific ports. It is important to learn which ports are used in PI System communication, since firewalls work by blocking communication over network ports unless otherwise specified.

6.2.1 List of ports used in PI System communication

The following table lists the ports used for communication with PI System applications

Port	From	To
5450	All PI Client (e.g., PI Vision, PI Processbook, PI DataLink)	Data Archive
5457	AFSDK Client (e.g., PI Vision, PI Processbook, PI DataLink)	Asset Framework
5459	PI SQL for AF Clients (e.g. PI WebParts, PI OLEDB Enterprise)	Asset Framework
5468	PI Notifications Clients (e.g. PI System Explorer, PI DataLink)	PI Notifications
5463	AFSDK Client (e.g. PI System Explorer)	PI Analysis Service

Note: The ports listed above are used for PI System process data and configuration data flows. Infrastructure such as AD authentication and backend database connections may require additional ports. For an exhaustive list of ports, refer to *KB01162 – Firewall Port Requirements*.

6.2.2 Directed Activity – Enable the Windows Firewall on PISRV01



In this part of the class, you will perform a learning activity to explore the different concepts presented in this chapter or section. You may be invited to watch what the instructor is doing or perform the same steps at the same time. You may play a game or hold a quiz. Your instructor will have directions.

Activity Objectives

- Enable Windows firewall
- Create firewall rules necessary for PI System communication

Approach

Step 1. On PISRV01, in PI Vision, open your “Pump Overview” display

Step 2. On PIINT01, open your PI Message Logs.

Hint: Run the PI SDK Utility and click on “Get Messages” when needed.

Step 3. On PISRV01, run the program “Windows Defender Firewall”

Step 4. You should see that all firewalls (Domain, Private and Public) are off. We will enable all three Firewalls.

a. On the left menu, click on “Turn Windows Defender Firewall on or off”

b. Change the Firewall state to “On”. Inbound connections should be blocked by default.



Step 5. Watch what happens on your “Pump Overview” display. What has happened to your data?

Step 6. Validate your hypothesis. On PIINT01:

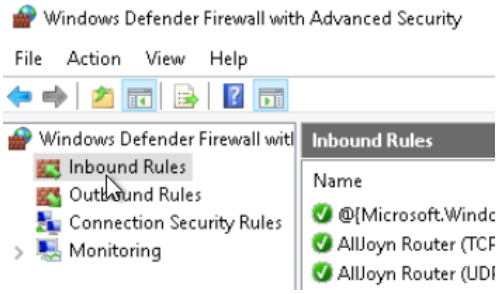
a. Check the PI Message Logs. Is your PI Interface throwing an error?

b. Run the Windows Powershell application

c. Run the following commands:

TNC PISRV01 -port 5450

Step 7. On PISRV01, in the application Windows Defender Firewall, click on Advanced settings in the left menu, select “Inbound Rules”



Step 8. Select “New Rule…”

Step 9. Create a rule to allow connections through TCP port 5450

a. In the “Rule Type” window, select **Port**

b. In the “Protocol and Ports” window, select “TCP” and select “Specific local ports”, and write port number 5450.

c. In the “Action” window, select “Allow the connection”

d. In the “Profile” window, select Domain + Private + Public

e. In the “Name” window, give your new rule a name (e.g. Data Archive – port 5450)

Step 10. Validate that the rule is working. On PIINT01:

a. Run the Windows Powershell application

b. Run the following commands:

TNC PISRV01 -port 5450

c. Check the PI Message Logs. Has the PI Interface reconnected?

Step 11. Make sure data collection has resumed by checking your “Pump Overview” display

Step 12. Enable firewall on your PI AF Server

a. Repeat steps 7, 8 and 9 using port 5457 to allow applications to connect to your PI AF Server

b. Repeat steps 7, 8 and 9 using port 1433 to allow PI Vision and other applications to connect to your SQL Server

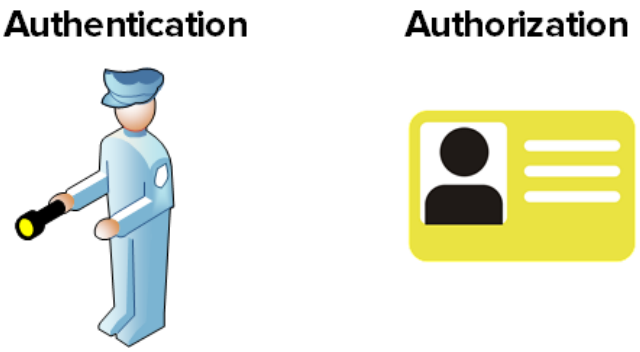
Step 13. Bonus: Lock down the Firewall further by only allowing connections to the Data Archive’s IP address using the rule you created.

6.3 Authentication vs. Authorization

We began our discussion of authentication and authorization in chapter 2, when configuring security for our PI Interface instance. Let's review what we know so far. In the context of the PI System:

- Authentication is the process that verifies the identity of a user or process, before allowing it to connect to the Data Archive
- Authorization is the process that determines what an application can do once connected to the Data Archive or the Asset Framework (e.g. create a PI Point, create an asset, run a backup, etc.)

The analogy we made previously was the Data Archive (or the Asset Framework) as a facility. The process of authentication is like the security guard at the entrance of the facility. He decides whether someone should be let in. If he does let them in, he gives them an access card. This access card is their authorization. It will give them access to specific rooms within the facility.



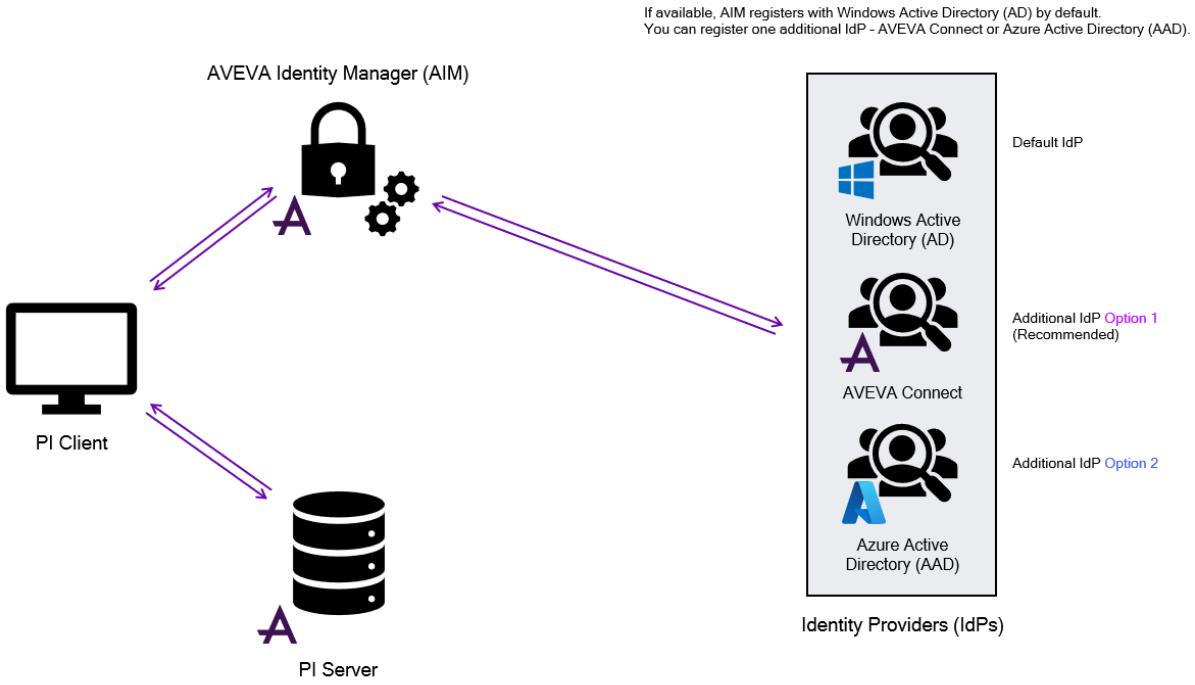
6.4 Asset Framework Security

6.4.1 Authentication and Authorization

AF version 2.7 and later uses a security model like the one used in PI Data Archive (will be discussed further in this chapter). AF Mappings use Windows Integrated Security to authenticate users on the AF. With this method, users and services connect directly to the AF using their Windows account. After authentication, the user is given an AF identity which is used for authorization.

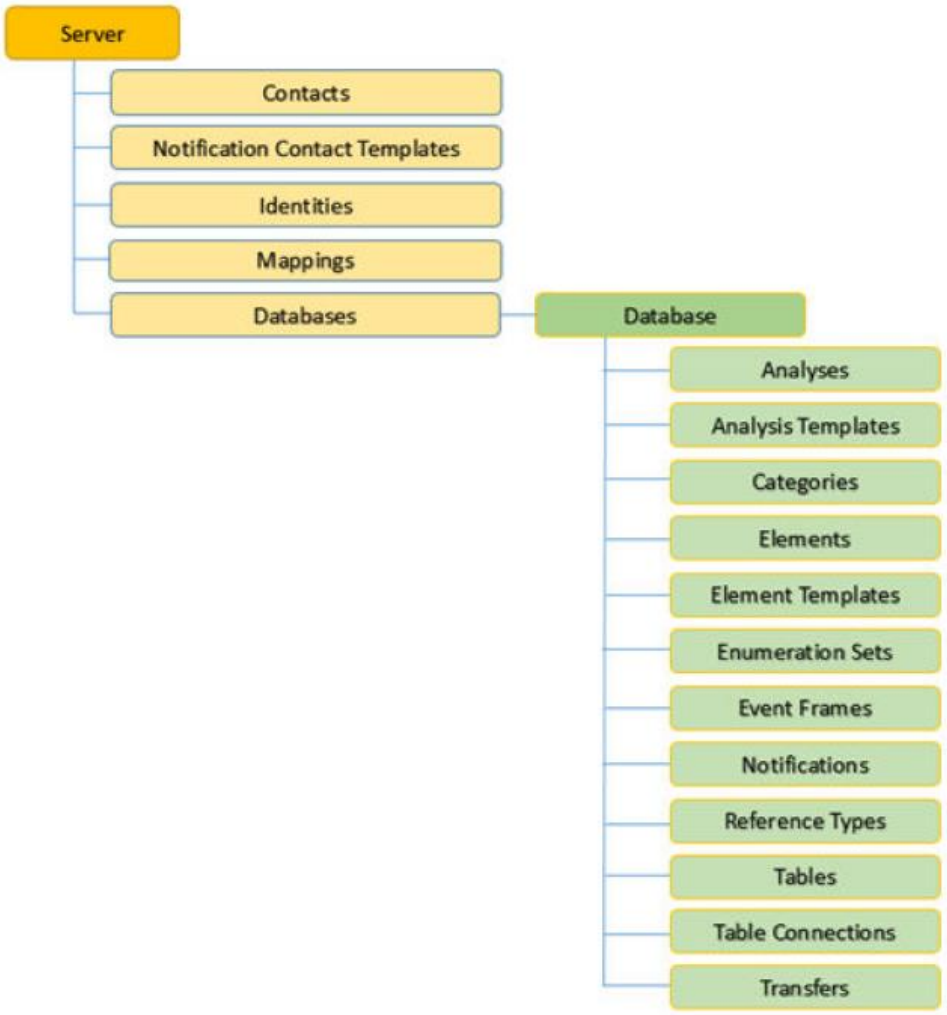
PI Server 2023+ supports for Claims-authentication. In this method, AVEVA Identity Manager (AIM) is the Identity Service that handles OIDC between PI Client applications, the PI Server, Windows Active Directory (AD), and either AVEVA Connect or Azure Active Directory (AAD) as an additional IdP. Users can use their AVEVA Connect/Azure account as well as their Windows account to authenticate to PI Server. The authorization part is the same.

Please note that PI Server 2018 SP3 will be a feature stable version and will keep receiving bug fixes and patches.



6.4.2 Security hierarchy

AF identities control read, write, delete and various other permission on AF components. Each AF object (shown in the illustration below) has an associated security descriptor as well as a type (Elements, Notifications, etc.). Each object of the same type belongs to a collection. Finally, each collection has an associated security descriptor that contains access permission information.



Security descriptors for some collections are configured for the entire server (Contacts, identities, Mappings, etc.) while others can be configured for specific databases (Elements, Event Frames, Notifications, etc.).

Note: A user **MUST HAVE** read permission on an AF database to be able to read any object within it. Same principle goes for write permission and modifying an object. Note, if you grant access at the database level, that access does not inherit down to the contained objects.

There is one exception: a user with “admin” rights on the “Server” objects will have unlimited access to everything in the server, regardless of the ACLs of the server’s objects.

6.4.3 Permission inheritance

When an AF object or collection is created, a default set of access permission is assigned, based on the access permissions that are set on the parent. However when you change permission on the parent, the following **Child Permission** settings can be used:

Option	Description
Do not modify child permissions	Prevents access permissions that have been set for the current object or collection from being replicated to child collections and objects in the AF hierarchy. This option is the default when the connected AF server is running 2.5 and earlier versions.
Update child permissions for modified identities	For each selected item on the **Items to Configure** list in the Security Configuration window, replicates the access permissions for all child collections and objects for each identity on the **Identities** list whose access permissions have been modified. This option is the default when the connected AF server is running 2.6 and later versions. This option is not available when the connected AF server is running 2.5 and earlier versions.
Replace child permissions for all identities	For each selected item on the **Items to Configure** list in the Security Configuration window, replaces all child permissions for every identity on the **Identities** list with the parent access permissions.



For more information on AF security see the "Security configuration in AF" section in *PI System Explorer User Guide,* version 2024.

6.4.4 Directed Activity – AF Security



In this part of the class, you will perform a learning activity to explore the different concepts presented in this chapter or section. You may be invited to watch what the instructor is doing or perform the same steps at the same time. You may play a game or hold a quiz. Your instructor will have directions.

Activity Objectives

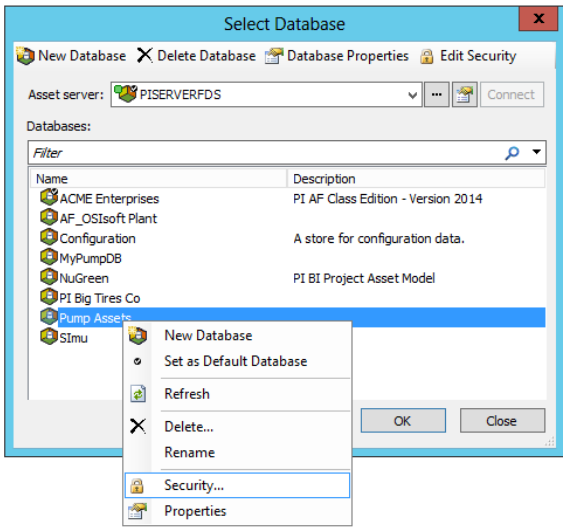
- Familiarize yourself with AF security
- Modify AF security from an existing database

Problem Description

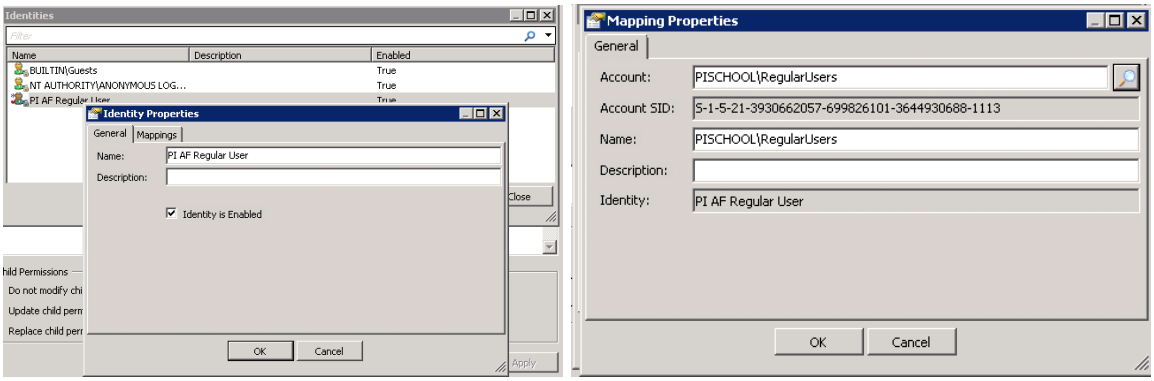
In order to maximize the AF experience, you want to create a AF Identity that will be used by all of your regular users. This identity should have read on the Pump Assets database and should also have the permission to create and modify only elements.

Approach

1. Open PI System explorer and click on the Database icon on top left of your screen.
2. In the select Database section right click on your database name and select security.



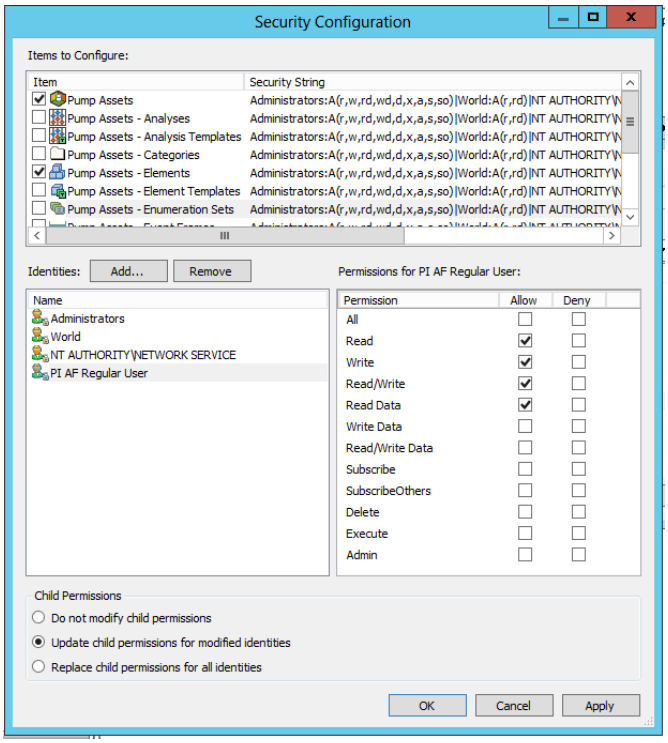
Step 1. In the Identities section, Add a new identity named “PI AF Regular Users” and map it to the RegularUsers Windows group



Step 2. We can now add the Read permission to the “Regular Users” Identity. With all the items selected under “Items to Configure”, select only **Read** and **Read Data** access for “Regular Users”. Select the child permission **Replace child permissions for all identities** and click on Apply.

Step 3. The last step is to add the Write permission to the Element section. In order to have write permission to an object in a database, the AF Identity “Regular Users” needs to have write access to the database.

Step 4. Under “Items to Configure”, uncheck all the items except “Database” and “Elements”. Then, select the identity “Regular Users”, and add Write permission. Finally, select **Replace child permissions for all identities** and click on Apply.



Step 5. Now go ahead and test if your AF security modifications worked by right-clicking the PI System Explorer icon on the task bar and selecting “run as a different user”. Run as the user “pischool\Joe” (password: ITPROSpwd01!) who is a member of the group “RegularUsers” .

6.4.5 Exercise – Your Database Security



This is a solo or group activity that is designed to maximize learning in a specific topic area. Your instructor will have instructions and will coach you if you need assistance during the activity.

Exercise Objectives

- Edit the Security for your Pump Asset database

Problem Description

You would like to configure the security of your Pump Asset database. The first thing that should be done is to restrict the access to the database so that only the Windows users you have added can read the database.

The engineers (Engineers Windows group) should have access to create and modify all the elements and analysis of the database but not the templates.

The supervisors (Supervisors Windows group) would like to be informed of any problem with the pumps. In order to do so they would need to be able to create PI Notifications on the Database.

As for the operators (Operators Windows group), they only need to be able to view the elements and attributes already build in the database.

Approach

Step 1. Start by creating the AF Identities needed and mapped those to the corresponding Windows account

Step 2. Next step would be to modify the AF security of your database in order to respect the security definition defined in the Problem Description

Step 3. Test the AF security by right clicking the PI System Explorer icon and selecting ***run as a different user***. For the following user are you able to:

- PISCHOOL\Bertha (password: ITPROSpwd01!)
  - Create a new element in your database: ☐ YES ☐ NO
  - Create a new analysis in any elements: ☐ YES ☐ NO
  - Modify the pump template ☐ YES ☐ NO
- PISCHOOL\Homer (password: ITPROSpwd01!)
  - View elements attribute and values: ☐ YES ☐ NO
  - Modify an element or template: ☐ YES ☐ NO
- PISCHOOL\Charles (password: ITPROSpwd01!)
  - Access and create notifications ☐ YES ☐ NO

6.5 Data Archive Security

In this section, we will cover main topics regarding security of PI Data Archive and implement some of the best practices into our environment.

- Secure authentication:
    - Upgrading to PI API For Windows Integrated Security on an interface machine.
    - Disabling PI Trusts on PI Data Archive.
- Secure authorization:
    - Disabling PIWorld identity.
    - Providing minimum access to users and applications

6.5.1 Authentication

There are three different methods of authentication on the Data Archive:

1. PI Mappings

PI Mappings use *Windows Integrated Security* to authenticate users on the Data Archive. With this method, users and services connect directly to the Data Archive using their Windows account. A PI Mapping grants a Windows user or group specific rights on the Data Archive by assigning a PI Identity.

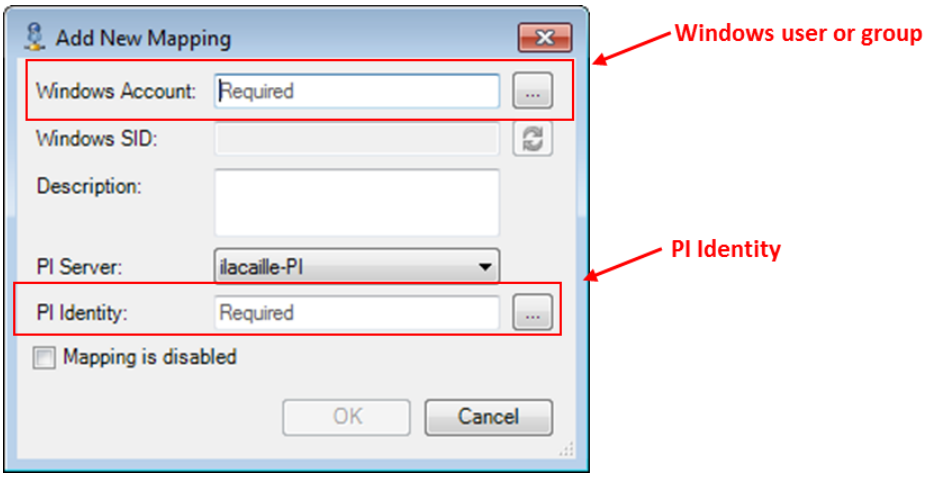
This method of authentication has several advantages:

- It is the most secure
- It enables transport security (encryption in transit) of communications with the Data Archive[1]
- It represents the least amount of maintenance for PI System administrators
- It allows users to connect directly with their Windows accounts

The recommended strategy for using PI Mappings is to create a Windows Group for each level of authentication needed on the Data Archive (e.g. one group for Read-Only users, one group for PI System Administrators, etc.), then assign a unique PI Identity to each one of these groups.

PI Mappings are created from System Management Tools, from Security > Mappings & Trusts > Mappings Tab, by pressing the New button . This will open the Add New Mapping Window

[1] Requires the following versions or later: PI Data Archive 2015, PI Buffer Subsystem 4.4, PI AF SDK 2015, PI SDK 2016 and PI API 2016 for Windows Integrated Security



The following conditions must be true to use PI Mappings:

- The application must connect with **PI AFSDK (any version), PI SDK version 1.3.6 or later or the PI API for Windows Integrated Security (version 2.0.1.35 and later, released in 2016)**
- The connecting application is running on a Windows operating system

If these conditions cannot be met, a PI Trust should be used for authentication.

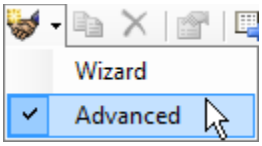
2. PI Trusts

PI Trusts should NOT be used unless it is not possible to authenticate using Windows Integrated Security. The most common scenario is PI Interfaces and other applications running on non-Windows Operating Systems

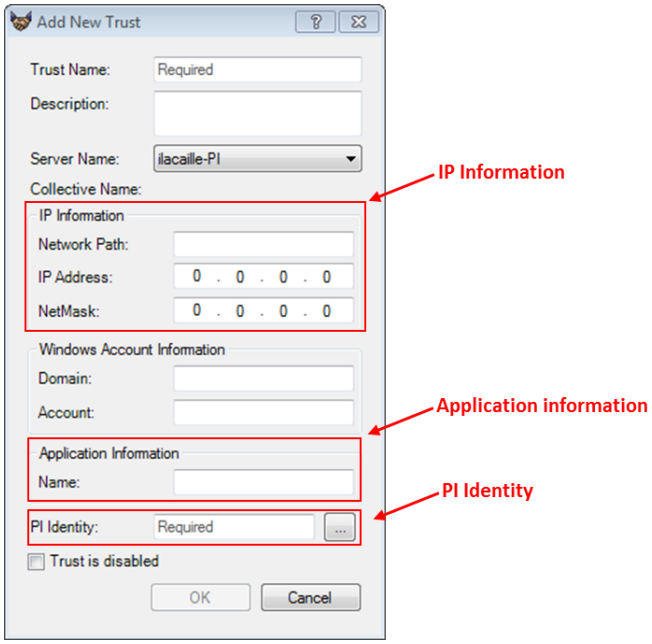
Note: Prior to 2016 release of the PI API for Windows Integrated Security, any applications using the PI API, such as PI Interfaces, could not use PI Mappings. Now, almost all PI Interface nodes can be upgraded to the new security model, regardless domain or workgroup configuration. For more information, see KB00354 - Supported Windows Security Configurations in Domains and Workgroups for the PI Data Archive

The PI Trust authentication method works by comparing the connection credentials of the connecting application to the credentials saved in PI Trusts. If the credentials match, the connection is allowed. No login is required by the application.

PI Trusts are created from System Management Tools, from Security > Mappings & Trusts > Trusts Tab, by pressing the arrow next to the New… button and selecting the advanced option:



This will open the Add New Trust Window.



It is not necessary to fill in all of the information in this Window. AVEVA recommends that you fill out PI Trusts using the 2+ Trust convention. This means you need to enter the following:

- **The IP Information:**

  The Network Path (Host name or fully qualified domain name of the computer)

  OR

  The IP Address and a NetMask of 255.255.255.255.

- **The Application Information**

  The application name. Applications that connect through the PI API send an identifier called an application process name, or procname. This is a four-character string with an E appended. For example, the procname for the PI Perfmon interface is PIPeE.

3. Claims authentication

This method is available from PI Server 2023.

This feature comes in addition to the existing options: the Windows Integrated Security (NTLM or Kerberos) authentication provider and the native PI Trust authentication provider. OIDC is possible through Identity Services which federate to an Identity Provider (IdP) allow for Single Sign-On (SSO)

In this method, AVEVA Identity Manager (AIM) is the Identity Service that handles OIDC between PI Client applications, the PI Server, Windows Active Directory (AD), and either AVEVA Connect or Azure Active Directory (AAD) as an additional IdP. Users can use their AVEVA Connect/Azure account as well as their Windows account to authenticate to PI Server.

4. **Explicit Login**

The final authentication method, Explicit Login, is not recommended in any scenario. It only exists for backwards compatibility purposes. Using this method, users log in to the Data Archive directly using a PI User and a password.



Tip

AVEVA now recommends upgrading from PI trusts and Explicit Login to Windows authentication & PI mappings as the authentication model throughout your PI system. This can be done by upgrading to the PI API for Windows Integrated Security on all PI Interface nodes, and all other custom PI API applications, which run on Windows Operating Systems.

PI trusts and explicit logins are disabled on PI API 2018 for Windows Integrated Security. Therefore, before upgrading to PI API 2018 for Windows Integrated Security, you must configure PI mappings to replace any existing PI trusts used by PI interfaces.

6.5.2 Group Discussion – Windows Integrated Security Myth Buster!



The following questions are intended to reinforce key information, or to discover a new insight. Your instructor may choose to have you try to answer the questions on your own or have the group answer them together aloud.

Activity Objectives

- Describe requirements for Windows Integrated Security

Approach

For each of the following statements, circle **MYTH** or **FACT**. You will review each statement together as a class.

1. Windows Integrated Security (WIS) cannot be used if the PI Interface is in a workgroup and the Data Archive is in a domain

   MYTH FACT

   Explanation:

2. WIS cannot be used if the Data Archive is in a workgroup



Explanation:

3. WIS cannot be used if the Data Archive and the PI Interface reside in separate, untrusted domains



Explanation:

4. WIS cannot be used if the PI Interface is on a non-Windows operating system



Explanation:

6.5.3 Authorization

There are three types of security objects that grant authorization on the Data Archive: PI Identities, PI Users and PI Groups. All three represent a set of access permissions on the Data Archive.

1. **PI Identities**

   It is recommended to use PI Identities when configuring PI Mappings and PI Trusts. They cannot be used with Explicit Login, since there is no password associated with a PI Identity.

2. **PI Users**

   PI Users can be used when configuring PI Mappings and PI Trusts. Each PI User is associated with a password and therefore can be used with Explicit Login authentication. PI Users are still supported for backwards compatibility, and the standard built-in accounts, piadmin and pidemo, are still provided.



Tip

Piadmin is the default “god user” and should not be used in any PI Mapping or PI Trust for security purposes. The only valid use of piadmin is disaster recovery.

3. **PI Groups**

   PI Groups can be used when configuring PI Mappings and PI Trusts. In the past, PI Groups were used for grouping together PI user accounts and providing them with the same access permissions. This can now be accomplished by mapping Windows Groups to PI Identities using PI Mappings. PI Groups are still supported for backwards compatibility, and the standard built-in groups, piadmins and piusers, are still supported.

Note: The PIWorld Identity is a special PI Identity created by default during the Data Archive installation. This identity is granted by default to any user connecting to the Data Archive. By default, the PIWorld Identity has read access to all PI Points.

To limit the read access to all PI Points granted by the PI World Identity, two solutions are available: (1) Disable the PI World Identity or (2) remove the PI World Identity from the database security access control lists.

6.5.4 Directed Activity – Disable PIWorld Identity



In this part of the class, you will perform a learning activity to explore the different concepts presented in this chapter or section. You may be invited to watch what the instructor is doing or perform the same steps at the same time. You may play a game or hold a quiz. Your instructor will have directions.

Activity Objectives

- Disable PIWorld Identity on the PI Data Archive.

Approach

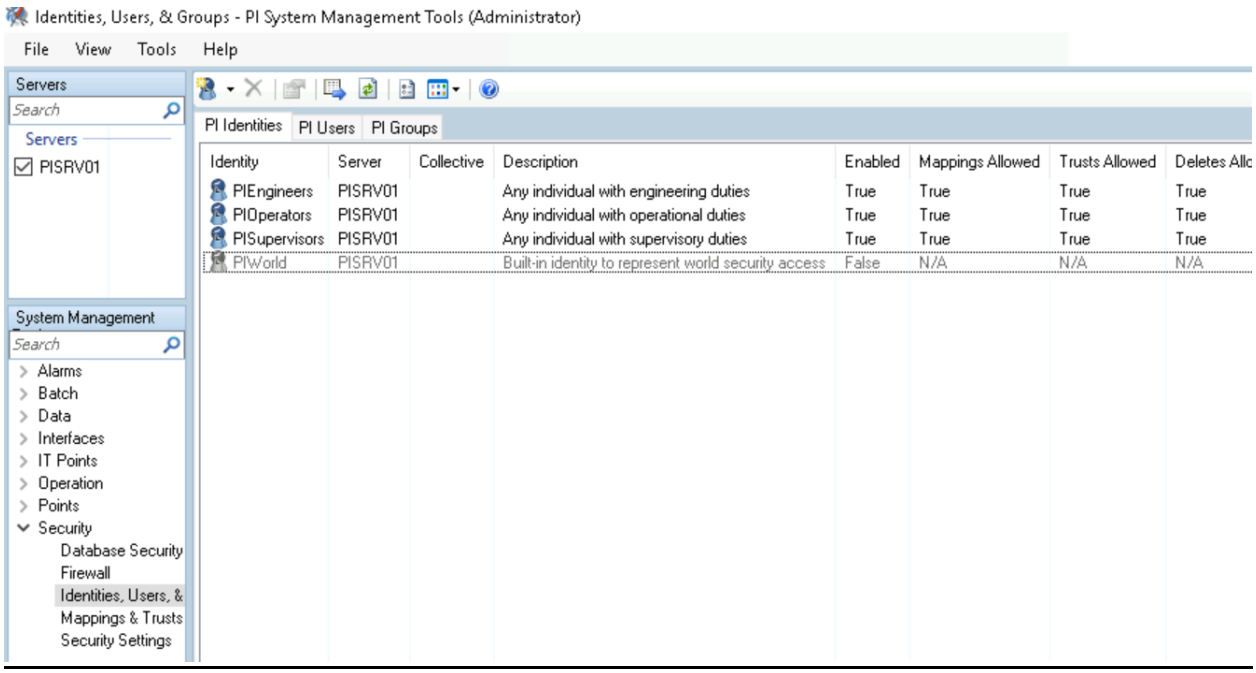
You are now sure that all users and applications are using PI Identities other than PIWorld. Now it is time to disable it. We recommend disabling it initially to leave a rollback option in case a user/application reports access issues related to this.

Step 1. On PISRV01, open PI SMT.

Step 2. Go to Security > Identities, Users and Groups.

Step 3. Find PI Identity called PIWorld. Right-click > Properties.

Step 4. Click on “Identity is disabled” and click OK. You will see that it is greyed out.



6.5.5 Access Permissions on the Data Archive

So far, we have seen how a connection can be authenticated (using PI Mappings, PI Trusts or Explicit Login), and what gives them authorization (PI Identities, PI Users or PI Groups). However, what permissions can you get once you obtain authorization?

The Data Archive has a variety of resources to which you can control access. These resources include PI Points, modules, archive configuration, backups, batches, audit trails, and so on. We refer to those PI resources as *secure objects*.

For each secure object, you can define which PI Identities (or PI Users or PI Groups) have read and/or write access. This security setting is stored in an access control list (ACL).

For example, let's say you have the following three PI Identities:



The "Read-only users" PI Identity should view the tuning parameters of the Data Archive, but should not be able to edit them. The "Administrators" and "Power Users" on the other hand, should have write access to the tuning parameters. The ACL for the tuning parameters should therefore be:

```
Administrators:A(r,w) | Power Users:A(r,w) | Read-only users:A(r)
```

There are three places where ACLs can be set:

1. On groups of secure objects in the Database Security table (SMT > Security > Database Security)
2. On individual PI Points (the Point Security and Data Security attributes)
3. On individual modules in the Module Database



Tip **AVEVA recommends providing minimum access to all users and applications.**

6.5.6 Group Discussion – Default Security



The following questions are intended to reinforce key information, or to discover a new insight. Your instructor may choose to have you try to answer the questions on your own or have the group answer them together aloud.

Activity Objectives

- Explore the default settings of the Database security table
- Explore the default security settings of a PI Point

Approach

Your instructor will give you a few minutes to complete the answers and then will facilitate a discussion.

Part 1 – Open SMT. Navigate to Security > Database Security.

1. What are the 2 default security settings for all the objects in the DB Security table?

2. Why would no read access be given to PIWorld for some of the tables?

3. What access would a PI Interface and the PI Buffer Subsystem need?

Part 2 – Examine the security settings for the point ***Sinusoid*** in Point Builder (SMT > Points > Point Builder).

4. What is the default security setting for the PI Point? What determines the default security for PI Points?

5. Describe what you think the difference is between Data Security and Point Security

6. What access would a PI Interface and the PI Buffer Subsystem need?

6.5.7 Exercise – Customize User Security



This solo or group activity is designed to maximize learning in a specific topic area. Your instructor will have instructions and will coach you if you need assistance during the activity.

Exercise Objectives

- Create PI Identities that will be mapped to Windows Users and Groups
- Configure point security for data access

Problem Description

You have many users requiring access to your PI System, but they all require different levels of access to different PI Points. Therefore, you want to grant access to the Data Archive and its resources based on user roles.

You have three domain groups:

1. Engineers
2. Operators
3. Supervisors

You need to create a security structure that enforces the following business rules:

- The point **OSIsoftPlant.Production** is a sensitive calculation and it should only be visible to the Supervisors group.

- The pressure sensor on Mixing Tank 2 is broken, so the data is being entered manually by the operators. Therefore, the Operators group needs write access to the data for PI Point **VPSD.OSIsoftPlant.PL2.MXTK2.Pressure**. It should be readable by anyone.

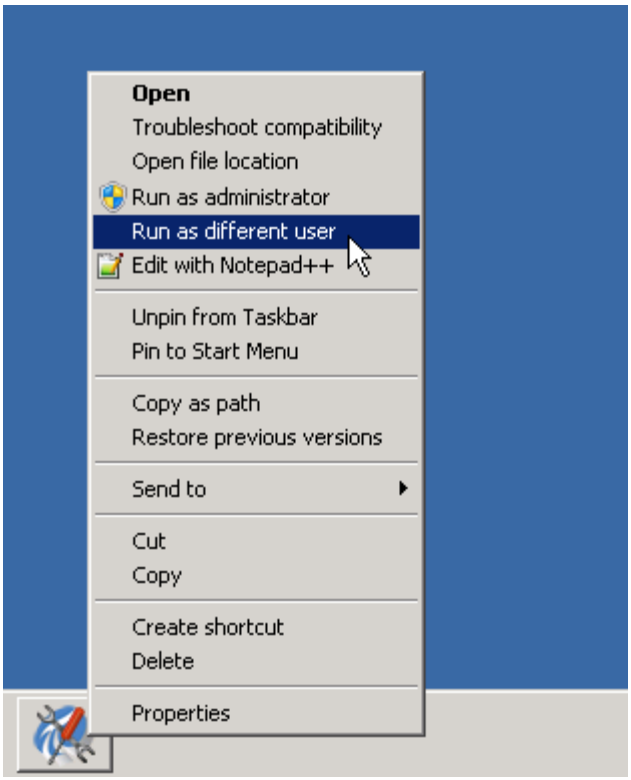
- The Engineers group needs to be able to edit the attributes of all of the OSIsoft Plant PI Points (with the exception of **OSIsoftPlant.Production**, which they should not be able to see).

Approach

Step 1. Map the three domain groups to the default PI Identities PIEngineers, PIOperators, PISupervisors

Step 2. Edit the Database Security and PI Point Security according to the rules above

Step 3. Test your security rules. To run SMT as a different user, hold the Shift key, then right-click SMT in the task bar and select “Run as different user”.



You may use the following accounts for your tests:

Domain Account Name	Member Of	Password
Charles	Supervisors	ITPROSpwd01!
Homer	Operators	ITPROSpwd01!
Bertha	Engineers	ITPROSpwd01!

1. Logon as Homer and try to search for the point **OSIsoftPlant.Production.** What result do you get? As Homer, write data to PI Point **OSIsoftPlant.PL2.MXTK2.Pressure** using SMT > Data > Archive Editor. Does it work?

2. Logon as Bertha. Try writing data to PI Point **OSIsoftPlant.PL2.MXTK2.Pressure**. Does it work? Now try turning compression off for **OSIsoftPlant.PL2.MXTK2.Pressure**.

3. Logon as Charles. Can you find and read PI Point **OSIsoftPlant.Production**?

6.5.8 Directed Activity – Tighten security for the PI Interface for OPC DA



In this part of the class, you will perform a learning activity to explore the different concepts presented in this chapter or section. You may be invited to watch what the instructor is doing or perform the same steps at the same time. You may play a game or hold a quiz. Your instructor will have directions.

Activity Objectives

- Create a PI Identity for PI Interfaces and for PI Buffers with least privileges.

Problem Description

In chapter 2, we installed and configured a PI Interface for OPC DA. We created a single identity called “PI Interfaces & PI Buffers”. Now, we want to tighten our security even further, by creating two PI Identities that gives them the minimum required privileges to the PI Interface and PI Buffer Subsystem on the Data Archive. Our knowledge base article Seven best practices for securing your PI Server, outlines the most secure configuration as follows:

Process	Read Permissions	Write Permissions
Interface	1. Database security > PIPOINT Table 2. Point Security on the PI Points	None
Buffer	None	Data security on the PI Points

You will implement this security configuration

Part 1 – Monitor the data coming from the PI Interface

Step 1. Whenever making changes to a PI Interface, it’s important to make sure we don’t affect data collection. Open the display “Pump Overview”, so you can keep track of data flow.

Part 2 – Create an identity for the PI Interface and PI Buffer Subsystem

Step 2. On PISRV01, run SMT. Navigate to Security > Identities, Users, & Groups.

Step 3. On the PI Identities tab, create a new PI Identity called “PIInterfaces” and another called “PIBuffers”

Part 3 – Edit the Database Security for the new PI Identity

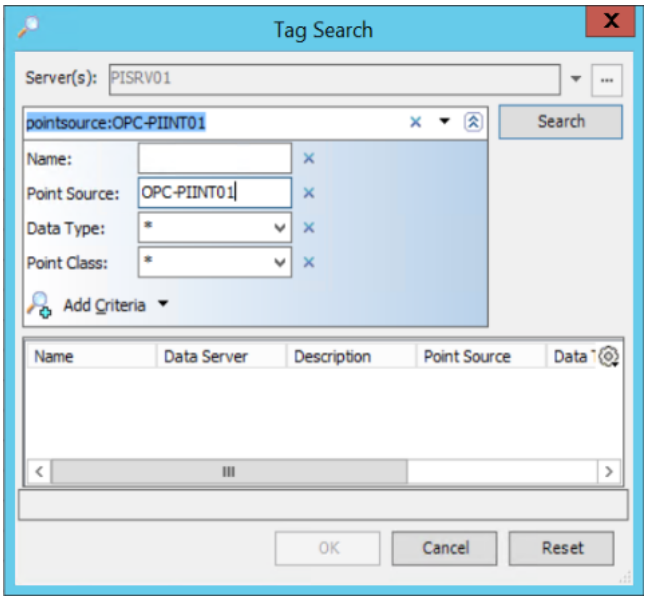
Step 4. Navigate to Security > Database Security

Step 5. Double click on the “PIPOINT” table.

Step 6. Add the “PIInterfaces” Identity, and give it read access

Part 4 – Edit the PI Point security for the Pump PI Point

Step 7. Load all the PI Points with point source “OPC-PIINT01” in Excel using PI Builder. Make sure to select the security attributes when importing the PI Points.



Step 8. Edit the ACL in the datasecurity and ptsecurity column.

a. “PIInterfaces” should have read access to ptsecurity

b. “PIBuffers” should have write access to datasecurity

Step 9. Publish your changes

Part 6 – Edit the PI Mapping for the PI Interface & Buffer

Step 10. Navigate to Security > Mappings & Trusts

Step 11. On the Mappings tab, open the PI Mapping you created for the Windows Account “svc-PIInterface”. Assign it to the PI Identity “PIInterfaces”

Step 12. Open the PI Mapping you created for the Windows Account “svc-PIBuffer”. Assign it to the PI Identity “PIBuffers”

Step 13. Navigate to Security > Identities, Users and Groups

Step 14. Delete the PI Identity “PI Interfaces & PI Buffers”

Part 7 – Validate your new security configuration

Step 15. On PIINT01, restart the PI Buffer subsystem (this should also restart the PI Interface)

Step 16. On PISRV01, in SMT, navigate to Operations > Network Manager Statistics. How are opcpE and pibufss.exe connecting?

Step 17. Return to your “Pump Overview” display and validate that you are still receiving pump data from PIINT01



Tip

Always check your data after making changes to security!

6.5.9 Exercise – Identify the missing best practices



This solo or group activity is designed to maximize learning in a specific topic area. Your instructor will have instructions and will coach you if you need assistance during the activity.

Exercise Objectives

You have an interface that is running on PIAF01. Please have a look on its operation and identify what improvements we can make in terms of reliability and security.

Approach

Identify what are the possible improvements.

Part 1 – Check if the interface is working reliably.

- Is buffering configured?

  _____

- Are health points created?

  _____

Part 2 – Check the current security configuration

- What authentication method is used by the interface? How did you check this?

  _____

- What access does the interface have on PI Data Archive? Is it enough/too much?

  _____

- Is the access restricted to the interface application only?

  _____

Based on these, what do we need to do on this server?

_____

_____

_____

_____

_____

6.5.10 Directed Activity – Make the interface more reliable



In this part of the class, you will perform a learning activity to explore the different concepts presented in this chapter or section. You may be invited to watch what the instructor is doing or perform the same steps at the same time. You may play a game or hold a quiz. Your instructor will have directions.

Activity Objectives

- Make the interface more reliable by enabling buffering and creating health tags.

Approach

First, we need to make sure that security is set up in PI Data Archive for PI Buffering. What happens if we enable buffering before setting up access?

Part 1 – Monitor the data coming from the PI Interface

Step 1. On PISRV01, create a display in PI Vision a trend of the tag **PIAF01_Processor(_Total)_% Processor Time**

Part 2 – Prepare the Data Archive

Step 1. In the last directed activity, you created two PI Identities with the following permissions:

Identity	Read Permissions	Write Permissions
PIInterfaces	1. Database security > PIPOINT Table 2. Point Security on the PI Points with point source OPC-PIINT01	None
PIBuffers	None	Data security on the PI Points with point source OPC-PIINT01

These PI Identities are mapped to the following domain service accounts:

PI Identity	Domain Account
PIInterfaces	PISCHOOL\svc-PIInterface
PIBuffers	PISCHOOL\svc-PIBuffer

PI Interface and PI Buffer will be running under these accounts. This means they should get the correct PI Identities when connecting using PI Mapping. We do not need to test this, as it is working on PIINT01. But we need to ensure that these PI Identities have access to all tags that are coming from this machine.

Step 2. Load all the PI Points with point source “PERF-PIAF01” in PI SMT > Point Builder. Do the PIInterfaces and PIBuffers identities do not have the correct access on these tags. So, fix it for one of the tags.

Step 3. Load all the PI Points with point source “PERF-PIAF01” in Excel using PI Builder. Make sure to select the security attributes when importing the PI Points.

Step 4. Edit the ACL in the datasecurity and ptsecurity column. If you edited a single tag on PI SMT in Step 2, just copy the security settings from that tag to other tags.

a. Give the “PIInterfaces” identity read access to ptsecurity

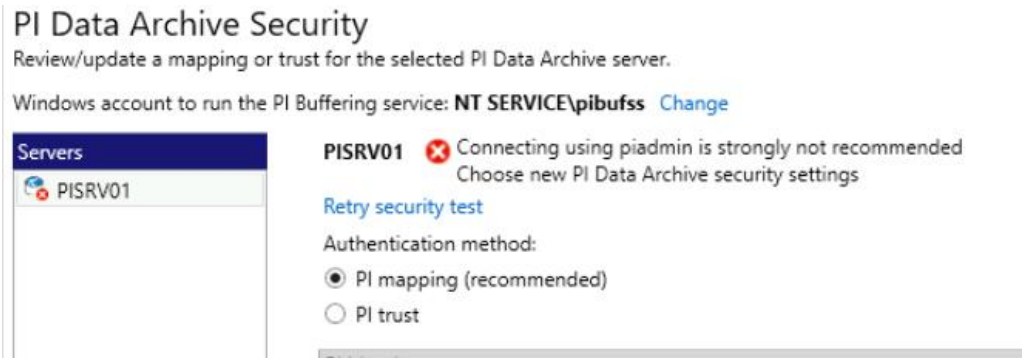
b. Give “PIBuffers” identity write access to datasecurity

Step 5. Publish your PI Point security changes.

Step 6. (Optional) You also have PI Analysis Service that is sending data to PI Data Archive from this machine. Once the PI Buffering is enabled, it will also start sending data through PI Buffering. What do you think will happen? How can we ensure that data flow is not affected?

Part 3 – Enable PI Buffering

Step 7. From PIAF01, configure PI Buffer Subsystem following the same steps in 2.17.3.
Note: You can see that currently it is connecting using piadmin. Why and how? Change the user running the PI Buffer Subsystem to PISCHOOL\svc-PIBuffer. Note that this time you do not have to configure any PI Mapping. Why?



Keep the buffer files in C drive. What could be the risk associated with this?

Step 8. Confirm that the data is flowing through PI Buffer Subsystem. How did you confirm?

6.5.11 Directed Activity – Upgrading existing PI Interface authentication from PI trusts and to Windows authentication



In this part of the class, you will perform a learning activity to explore the different concepts presented in this chapter or section. You may be invited to watch what the instructor is doing or perform the same steps at the same time. You may play a game or hold a quiz. Your instructor will have directions.

Activity Objectives

- Start using PI Mappings for this interface.

Approach

AVEVA recommends upgrading to the PI API for Windows Integrated Security on all PI Interface nodes, and all other custom PI API applications, which run on Windows Operating Systems.

Default PI API that is installed with any interface is using PI Trusts & Explicit Login. It does not support Windows Integrated Security. Therefore, we must install PI API for Windows Integrated Security separately. This will disable PI Trusts & Explicit Login on this machine. This will be shown in a warning before installation.



Part 1 – Upgrade the PI API to the PI API for Windows Integrated Security

Step 1. Log on to PIAF01

Step 2. In the folder C:\Class Files\Install Kits, run the program "PIAPI-2018-for-Windows-Integrated-Security_x.x.x.xx_". You may be prompted to reboot the computer.

Step 3. When the install finishes and the computer restarts, make sure the PI Buffer Subsystem and PI Interface services are running.

Part 2 – Validate the new authentication model

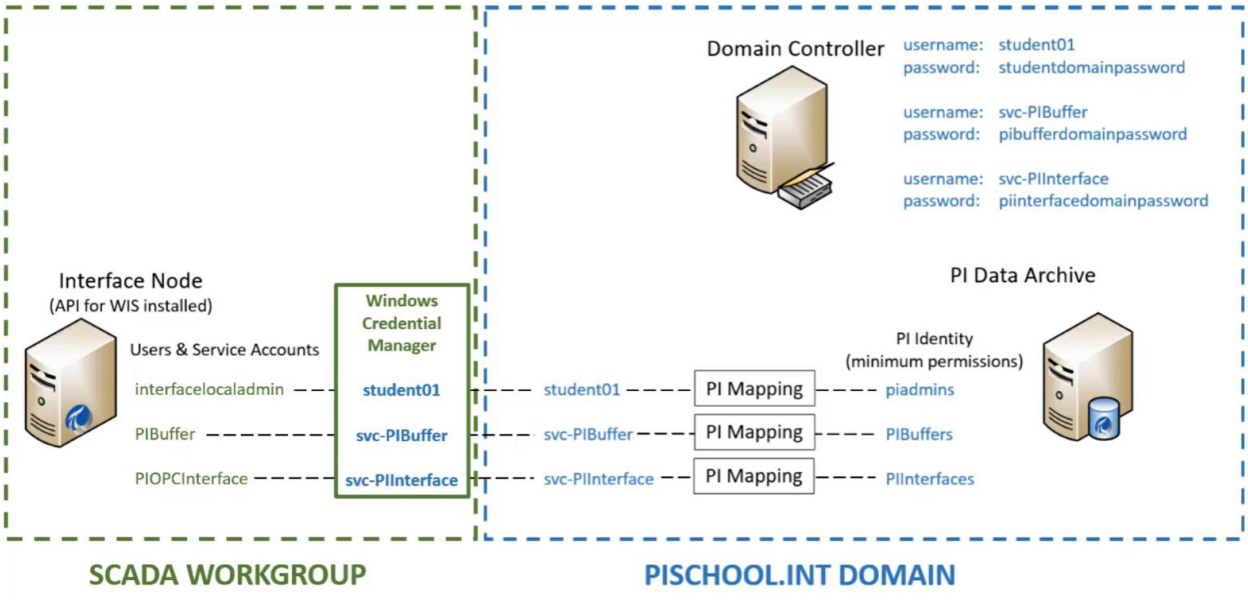
Step 4. Log on to PISRV01

Step 5. In SMT, navigate to Operations > Network Manager Statistics

Step 6. Confirm that the PI Interface for PerfMon has received the PIInterfaces identity using PI Mapping.

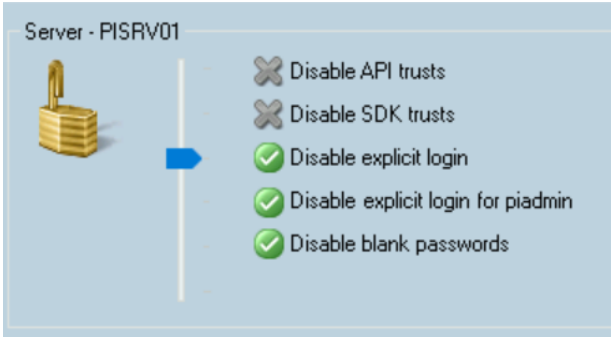
Step 7. In PI Vision, validate that you are still receiving data for the tag **PIAF01_Processor(_Total)_% Processor Time**

If you have a machine that is in another domain/workgroup, you will need to use Windows Credential Manager to be able to use Windows Integrated Security. [This video on our YouTube channel] shows the procedure step-by-step.



6.5.12 The "Security Slider"

You can disallow specific methods of authentication to your Data Archive. This is controlled by the **Security Settings** plug-in in SMT (Security > Security Settings).



You should set the slider to a minimum of explicit logins disabled. This should not impact you at all if you avoid using piusers and pigroups. The new installations will have this configured by default.

Tip

If you want to set the security slider all the way to the top, you need to ensure that none of the active connections to the Data Archive use PI Trusts. A good way of checking is by using Network Manager Statistics in SMT. The connections using PI Trusts will be indicated in the "Trust" column.

6.5.13 Exercise – Disable PI Trusts on PI Data Archive



This solo or group activity is designed to maximize learning in a specific topic area. Your instructor will have instructions and will coach you if you need assistance during the activity.

Exercise Objectives

- Disable PI Trusts on PI Data Archive.

Problem Description

Now that we ensured that all the connections are using PI Mappings, we can disable PI Trusts on the PI Data Archive. This means that all the connections will have to use Windows Integrated Security

Step 1. Check if all the connections are using PI Mappings using Network Manager Statistics.
Step 2. Disable PI Trusts using Security Slider. Save the setting.
Step 3. Stop and start PI Data Archive.
Step 4. Check if all your PI Points are updating.

7. Monitoring a PI System

Objectives:

- Browse for Stale and Bad Points
- Build Health Points for interfaces
- Use PI Notifications to monitor PI System
- Create displays and building dashboards for PI System monitoring

7.1 Monitoring Tools

The PI System itself can be used to monitor its components. As we saw in chapter 2, UniInt interfaces include the functionality to create health tags. You can also collect data about the health of the PI System using PI Interfaces specifically designed to collect IT data.

1. **PI Interface for Performance Monitor:** Collects Windows Performance Counter data. Performance counters provide information about the performance of the operating system, applications, services, and drivers. Windows Performance Counters can be viewed outside of the PI System using the "Performance Monitor" application (perfmon.exe) included on the Windows operating system.
2. **PI Interface for Ping:** Measures the response times in milliseconds of ICMP echo messages (i.e. "pings") that it sends to a remote machine. It is therefore used to determine the latency of a TCP/IP network and help diagnose network connection issues between two machines.
3. **PI Interface for TCP Response:** Measures the availability and response times of various essential services that are part of a TCP/IP network, including web servers, mail servers, and of course, PI Servers.
4. **PI Interface for SNMP:** Gathers information from SNMP enabled devices residing in a TCP/IP network. Devices that typically support SNMP include routers, switches, servers, workstations, printers and modem racks.

7.2 Group Questions – What should I monitor?



The following questions are intended to reinforce key information, or to discover a new insight. Your instructor may choose to have you try to answer the questions on your own or have the group answer them together aloud.

Questions

1. As a PI System Admin, what would you like to monitor in your PI System?

On interface server:

_____

On PI Data Archive server:

_____

On PI AF Server:

_____

Other applications:

_____

2. Please go to documentation page Which performance counters to monitor and find at least 5 counters that you find useful and share them with others.

7.2.1 Solo Exercise – Explore IT Monitoring Database



This is a solo exercise that is designed to maximize learning in a specific topic area. Your instructor will have instructions and will coach you if you need assistance during the exercise.

Activity Objectives

In our PI System, we have an existing PI Interface for Performance Monitoring on PIAF01 that monitors some basic Performance Counters. We also monitor an interface using Health Tags.

Please go to the IT Monitoring database and answer these questions.

Questions

1. What are we monitoring on PIAF01 server?
2. What are we monitoring on PIPerfMon1 interface?
3. Are these elements using a template? How do you think this will help if we decide to monitor other servers or interfaces?

7.3 Stale and Bad Points

- **Stale Point**: a PI Point which has not received a value in a predetermined amount of time.
- **Bad Point**: A PI Point whose value is taken from the **System** Digital State set. The "System" Digital State set is a collection of digital states that the system uses for indicating errors or other unique conditions. An assumption is made that if a point has one of these values, it has a problem.

7.3.1 Directed Activity – Stale and Bad Points



In this part of the class, you will perform a learning activity to explore the different concepts presented in this chapter or section. You may be invited to watch what the instructor is doing or perform the same steps at the same time. You may play a game or hold a quiz. Your instructor will have directions.

Activity Objectives

- Define a Stale point
- Define a Bad point
- Search for Stale and Bad points

Approach

Step 1 : On PISRV01, run SMT and navigate to Data > Stale and Bad Points

Step 2 : Click on the "Search" button  to find Stale points (from 4 hours to 365 days) and all tags with a System digital state

Step 3 : How might you approach addressing these points? What questions should you ask?

8. Troubleshooting a PI System

8.1 Message Logs

The first step in troubleshooting is always the same: check the message logs! All PI System software write messages to log files. It is therefore important to learn which log files to check, and how to read them.

1. **PI Message Logs**

   Also known as the “SDK Logs”, these are the logs of all applications based on the PI SDK. There is one PI Message log per computer where an SDK application is installed. The logs are managed by the PI Message Subsystem.

   Applications that write to this log:

   - Data Archive subsystems
   - PI Interfaces (UniInt version 4.5.0.x and later)
   - PI Client applications

   How to access these logs:

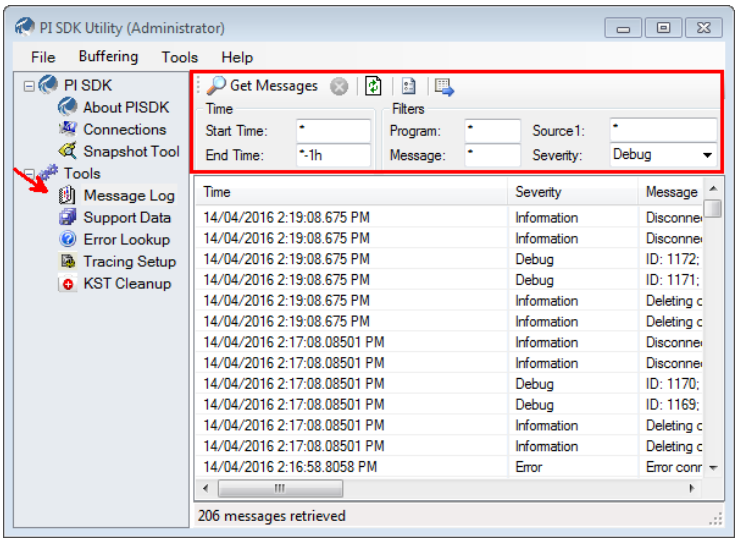
   - On the Data Archive: SMT > Operations > Message Logs
   - On all computers with PI-SDK 1.4.0 and greater:

     Step 1 : Run the program “PISDKUtility”

     Step 2 : In the left pane, select Tools > Message Logs

     Step 3 : Set the filters to obtain messages (start time, severity, etc.)

     Step 4 : Click on “Get Messages”



- On a PI Interface node: PI ICU > Press the "View Current PI Message Log continuously" button 
- On all computers, you can use the command line utility pigetmsg.exe

  Step 1 : Open a command prompt window

  Step 2 : Change the directory to pi\adm or pipc\adm

  Step 3 : Type **pigetmsg –f** to view the logs continuously

  Step 4 : For more filtering options, type **pigetmsg –?**

2. **Event Logs**

   Event Logs are the centralized logs on a Windows machine. There are two different types of logs:

   - **Windows Logs**: These logs includes all important events on the operating system, split up into the following categories: Application, Security, Setup, System and Forwarded Events.
   - **Applications and Services Logs**: These logs are specifically for applications, with each application writing to its own log.

   PI System applications write to the Windows Application Log, and sometimes to a dedicated log under Applications and Services.

   It's also a good idea to look at the other Windows logs (Security, System) if you suspect an issue might have occurred at the operating system level.

   Applications that write to these logs:

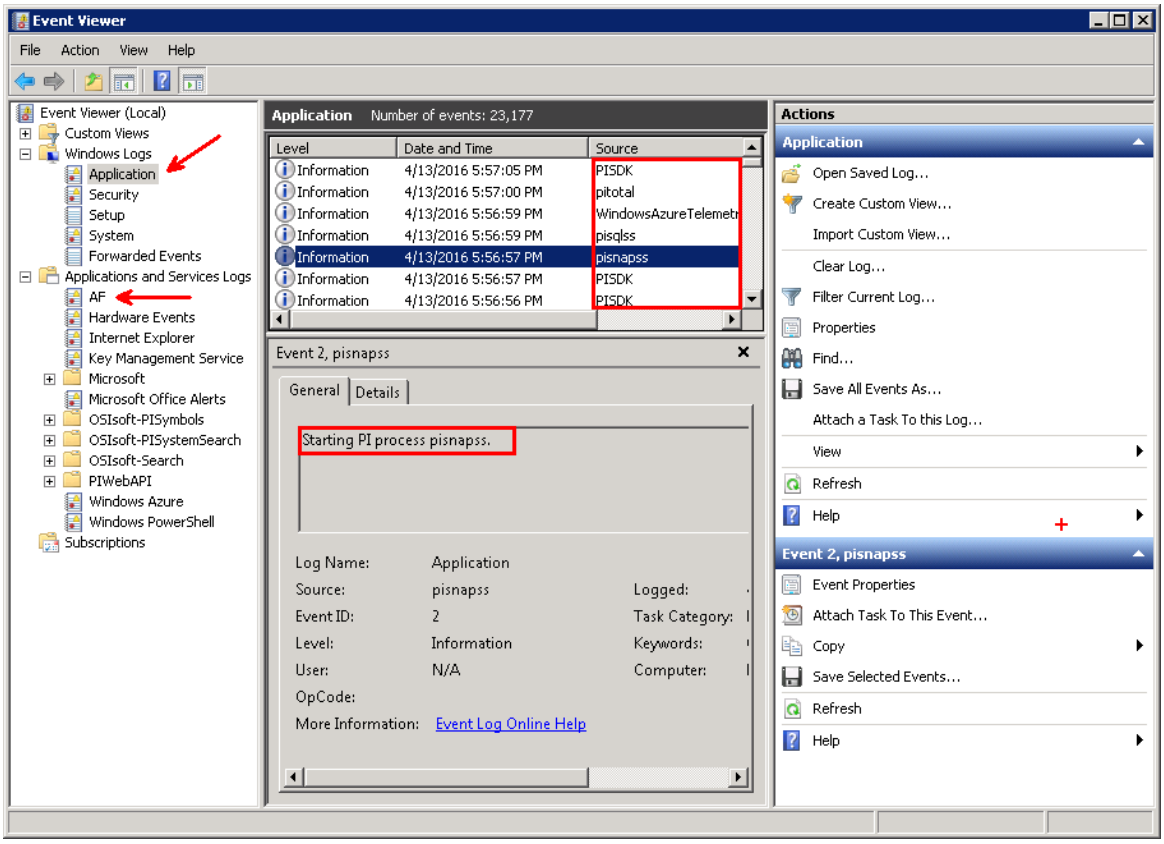
   - AF Application Service
   - Data Archive Subsystems (occasionally)

   How to access these logs:

   Step 1 : Run the application "Event Viewer"

   Step 2 : To access the Windows Application log, browse to Windows Log > Application. Look at the "Source" column to find messages written by PI System applications. You can also use the filter functionality of Event Viewer.

   Step 3 : To access a specific application's log, browse to Application and Services Logs, then find the name of your application (e.g. AF)



3. PIPC Logs

These logs are only used by older, PI API based applications. You should only need to access these logs if you are running older software.

Applications that write to these logs:

- PI Interfaces with a UniInt version earlier than 2.5.0.x
- PI API based applications

How to access these logs:

- On a PI Interface node: PI ICU > Press the “View current pipc.log continuously” button 
- Open the file PIPC\dat\pipc.log

8.2 Where to look for answers

So, you found an error message, now what? There are a few resources you can use to translate that message and find your solution:

1. **Search for a Solution on the customer portal** (https://my.osisoft.com)

   This solution search crawls all our online resources, including product documentation and Knowledge Base (KCS) articles.

2. **Search the documentation** (https://docs.aveva.com)

   This is an online repository of AVEVA documentation. It contains all the up-to-date administration and user guides for our products.

3. **Ask the community on PISquare** (https://pisquare.osisoft.com)

4. **Contact AVEVA PI Tech Support!** (https://my.osisoft.com)

   When contacting Technical Support, always make sure to have the following information on hand:

   - A clear description of the issue
   - Product and version information
   - A copy of the relevant message logs
   - Relevant screenshots, and if possible, steps to reproduce the issue
   - Urgency and Impact of this case
   - Your PI Server Serial Number (SMT > Operation > Licensing > InstallationID)

9. Final Exercise



This is a solo or group activity designed to maximize learning in a specific topic area. Your instructor will have instructions and will coach you if you need assistance during the activity.

Exercise Objectives

- Install another interface to collect data from a different data source.
- Consult other resources (documentation, YouTube channel) to complete the installation.

Problem Description

You have three options in this exercise:

1. Collect performance data from servers and applications using PI Interface for Performance Monitoring (PerfMon).
2. Collect data from text files using PI Interface for Universal File and Stream Loading (UFL).
3. Collect data from SQL Database using PI Interface for Relational Database (RDBMS).

9.1 Final Exercise – Import performance data from servers



This is a solo or group activity designed to maximize learning in a specific topic area. Your instructor will have instructions and will coach you if you need assistance during the activity.

Exercise Objectives

- Install and Configure a PI Interface for Performance Monitor on PIINT01 to collect performance data.
- Create AF Elements and Attributes for the newly monitored objects and organize these elements in a logical hierarchy.
- Create a PI Vision Display to monitor your PI System in real time.

Problem Description

You are the new PI System Administrator at Stark Industries. You want to monitor your PI System to make your life easier. There is already a database for this, and you need to add other servers and applications to this database.

In case you are stuck with a step, feel free to contact TechSupport (your instructor☺). Note that they will require some information from you to understand the issue. Check the previous page on what they might ask.

Approach

Step 1 : On PIINT01, install the PI Interface for Performance Monitor. Here is the documentation you should follow. This should be a simple installation.

a. Installation kit is in C:\Class Files\Installation Kits\PIPerfMon)_x.x.x.x.exe

Step 2 : Configure an instance to collect local Performance Counters. Make sure to follow documentation closely.

a. Set Point Source as PERF-PIINT01 and interface ID as 1

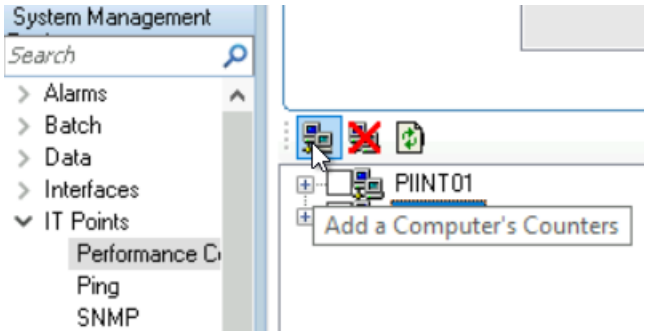
b. Use scan frequency of 5 seconds

c. Run the service as PISCHOOL\svc-piinterface (password: student). Hint: Do not skip the part of the documentation starting with "***On the target machine, open lusrmgr……***"

Step 3 : Create tags to collect Performance Counters.

a. To create tags, go to **PI SMT > IT Points > Performance counters**. On the **Tag Settings** tab, select the interface you created in Step 2.

b. On the **Build Tags** tab, make sure to create the tags for PIINT01, not PISRV01. For this, you should first add the computer to PI SMT:



c. Once the computer is added, select the correct tags. We need:

   i. CPU usage (**Processor(_Total)_% Processor Time**),

   ii. Available Memory (**Memory_Available MBytes**)

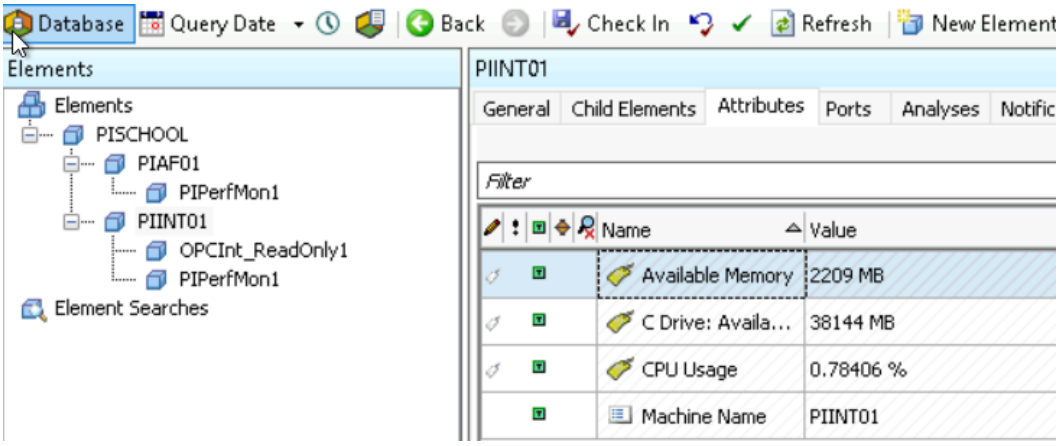
   iii. Available space in C drive (**LogicalDisk(C:)_Free Megabytes**).

d. Once the tags are created, edit their security settings to ensure the PI Interface and PI Buffer have correct permissions. You can copy the settings from previous OPC DA tags.

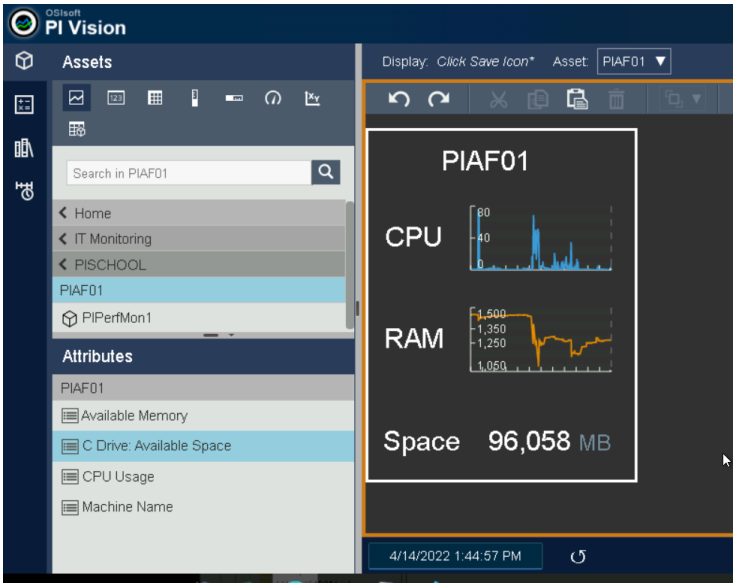
Step 4 : Create UniInt Health Tags for the PI Perfmon Interface. Device Status and Heartbeat are enough, but you can create other tags, too.

Step 5 : Confirm that you are receiving data to PI Data Archive. You should have at least 5 tags coming from the new interface.

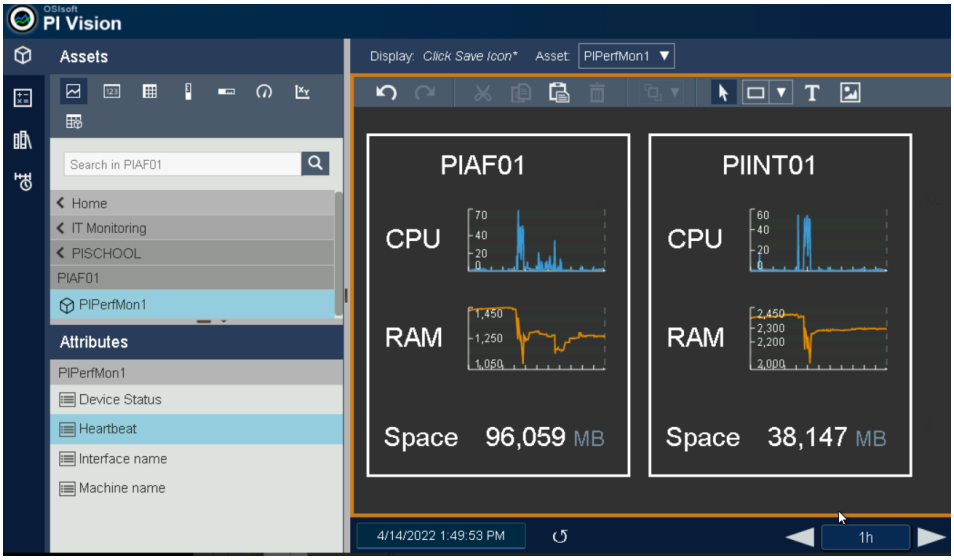
Step 6 : Create an element for PIINT01 and elements for PI Perfmon and PI OPC DA interfaces on IT Monitoring database. Hint: make use of existing templates.



Step 7 : Create a PI Vision display to monitor the servers and another display to monitor interfaces.



a. Make use of PI Vision Collections: after creating a display that shows everything for one element, select everything and click on “Convert to Collection”. This should replicate to all similar elements



Bonus tasks:

1. Create a new instance of the PI PerfMon interface on PIINT01 to collect Performance Counter data from PISRV01 remotely. Note that you do not need to install the PI Perfmon interface on PISRV01.
2. Add this machine and the new instance to the AF hierarchy.

9.2 Final Exercise – Import data from text files



This is a solo or group activity designed to maximize learning in a specific topic area. Your instructor will have instructions and will coach you if you need assistance during the activity.

Exercise Objectives

- Install and Configure a PI Interface for Universal File and Stream Loading (UFL)

Problem Description

You have some data about your pumps that is being exported to some folders on your interface machine. Your task is to collect data from text files located in these folders and then create attributes for the new data in your AF template..

Approach

Part 1 – Collect data from **C:\Class Files\Data\Exercise1** folder. These files contain the current that each pump has.

Step 1 : On PIINT01, install the PI Interface for Universal File and Stream Loading (UFL). This should be a simple installation. You can watch the second video in our YouTube playlist for installation.

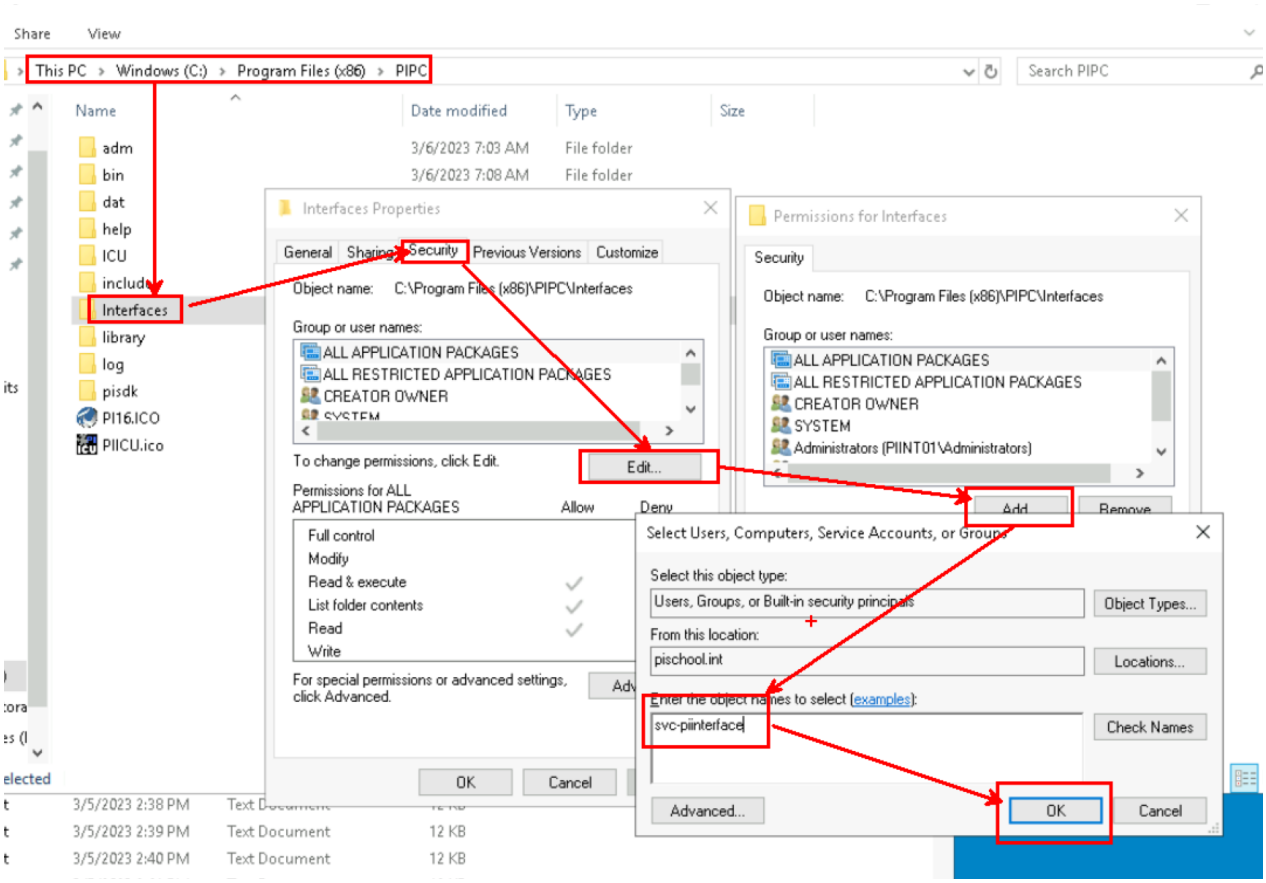
1. Installation kit is in C:\Class Files\Installation Kits\UFL_x.x.x.x.exe

Step 2 : Check the data files and configure new tags for PI UFL interface.

1. Tagname = PumpX.Current
2. Point Source = UFL-PIINT01.
3. PointType = Int16
4. Location1 = 1
5. Location4 = 1

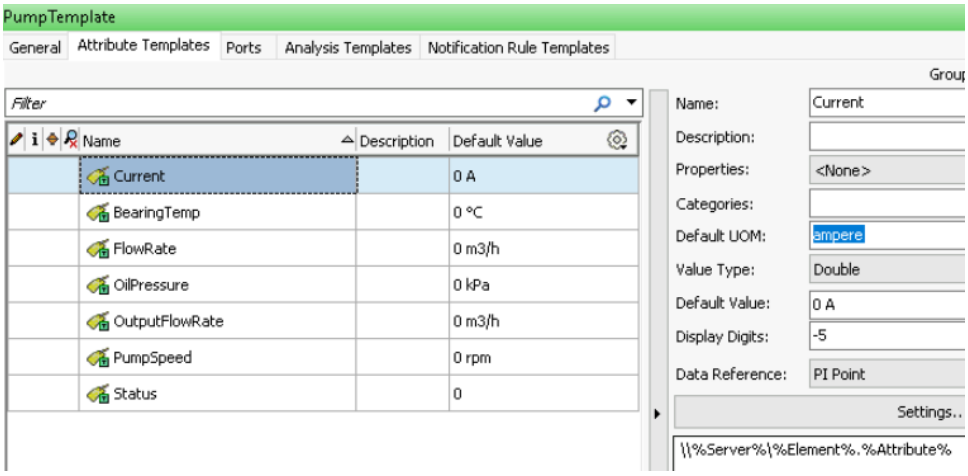
Step 3 : Configure and instance of the UFL interface to collect data from Exercise1 folder. You can follow the documentation or the YouTube playlist.

1. Set Point Source as UFL-PIINT01 and interface ID as 1
2. Use scan frequency of 30 seconds
3. Run the service as PISCHOOL\svc-piinterface (password: student).
4. Go to folder C:\Program Files (x86)\PIPC\Interfaces and provide full access for the user PISCHOOL\svc-piinterface



5. For configuring the INI file, follow the steps in the YouTube video. Note that the timestamp format is different in the video.
6. Once done, run the interface. Check PI Data Archive to see if data is updating.

Step 4 : Create a new attribute in your Pump element template for Current information. UoM is Ampere (A).



Part 2 – Collect data from **C:\Class Files\Data\Exercise2** folder. These files contain efficiency data coming from a third-party application.

Step 1 : Check the data files and configure new tags for PI UFL interface.

a. Tagname = PumpX.Efficiency

b. Point Source = UFL-PIINT01.

c. PointType = Int16

d. Location1 = 2

e. Location4 = 1

Step 2 : Configure and instance of the UFL interface to collect data from Exercise1 folder. You can follow the documentation or the YouTube playlist. This time the ini file will be a bit more complicated, so take your time to understand the logic behind it.

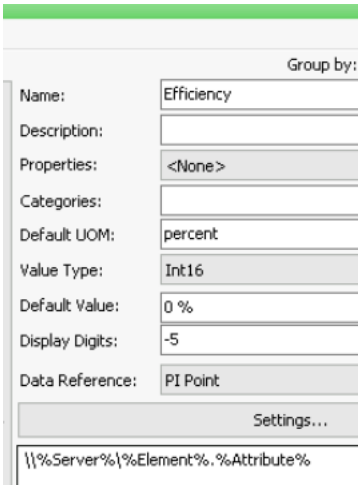
a. Set Point Source as UFL-PIINT01 and interface ID as 2

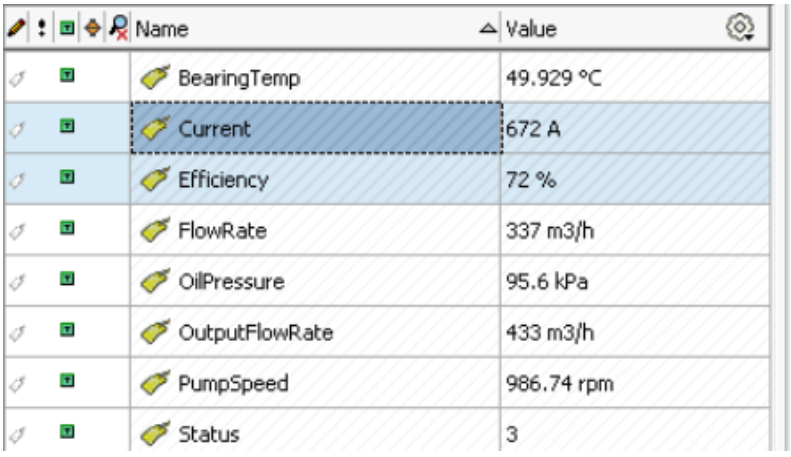
b. Use scan frequency of 30 seconds

c. Run the service as PISCHOOL\svc-piinterface (password: student).

d. For configuring the INI file, watch the YouTube video to understand the logic. Note that the timestamp format is different in the video.

Step 3 : Create a new attribute in your Pump element template for Efficiency information. UoM is percent (%). Confirm that you are getting data for both attributes.





9.3 Final Exercise – Import data from SQL Server



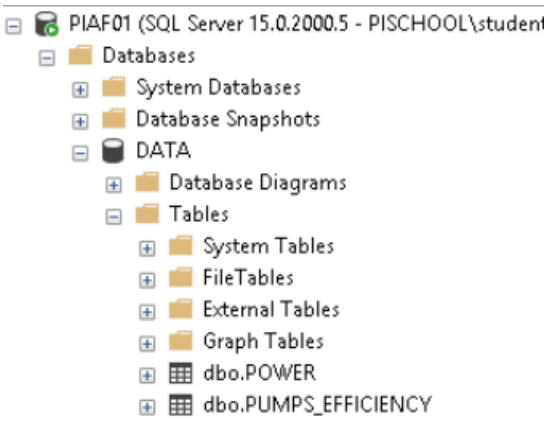
This is a solo or group activity designed to maximize learning in a specific topic area. Your instructor will have instructions and will coach you if you need assistance during the activity.

Exercise Objectives

- Install and Configure a PI Interface for Relational Database Management Systems (RDBMS)
- Note that this exercise requires basic level of understanding of SQL.

Problem Description

Collect data from tables in SQL database on PIAF01.



Approach

Part 1 – Collect data from **DATA.dbo.POWER** table. This table shows total power consumption for all the pumps.

Step 1 : On PIINT01, install the PI Interface for Relation Database Management Systems (RDBMS). This should be a simple installation.

a. Installation kit is in C:\Class Files\Installation Kits\RDBMSPI_x.x.x.x.exe.

b. Once asked, reboot the machine.

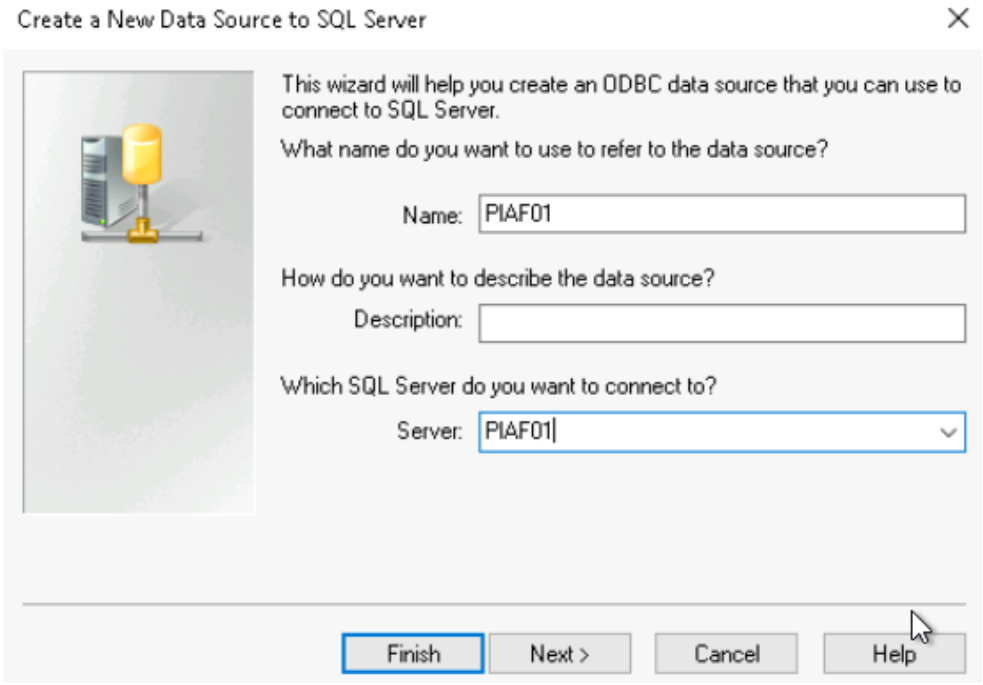
Step 2 : Configure and instance of the RDBMS interface to collect data from Power table. You can follow the documentation or YouTube video (note that you will need to skip the part about security between 2:45-6:15).

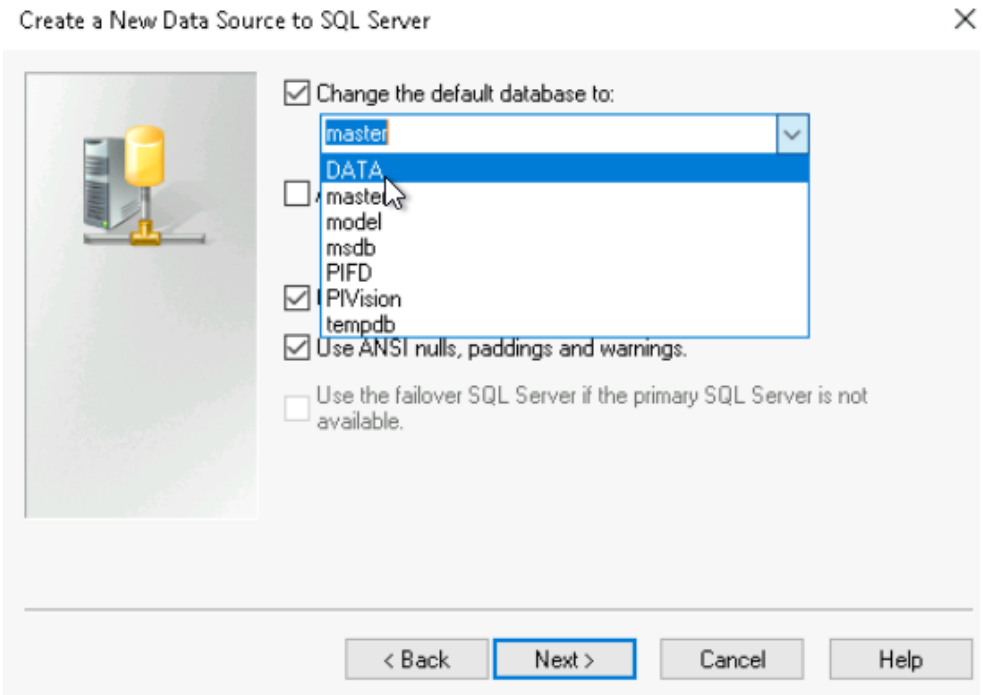
a. Set Point Source as RDBMS-PIINT01 and interface ID as 1

b. Use scan frequency of 30 seconds

c. Run the service as PISCHOOL\svc-piinterface (password: student).

d. Do not forget to create a DSN for PIAF01.

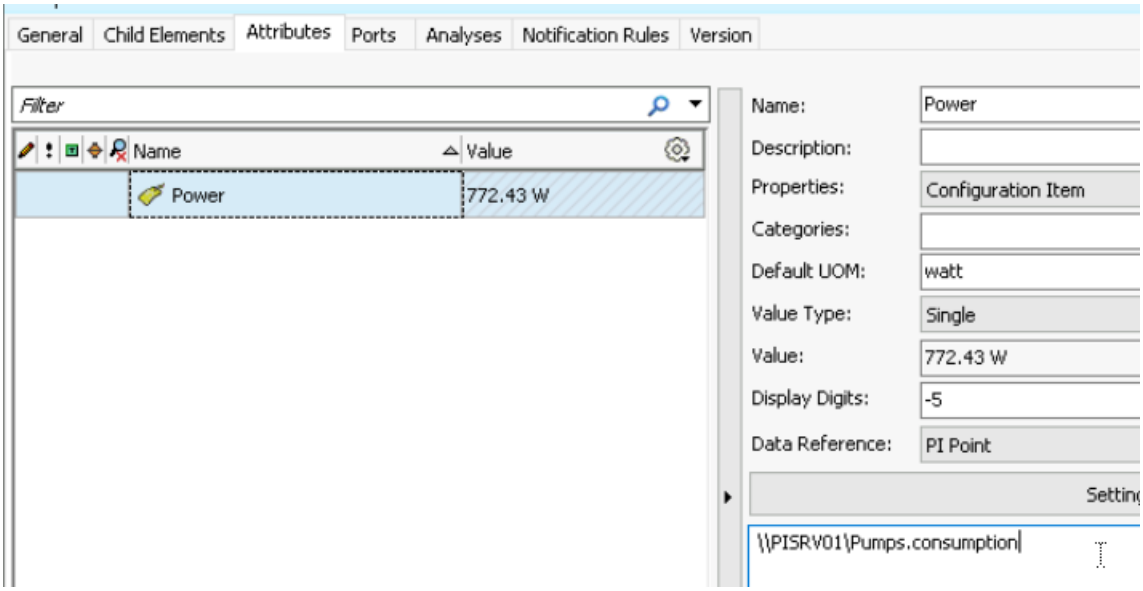




Step 3 : Configure a new tag for PI RDBMS interface using PI System Management Tools. Watch this video to configure the SQL query.

a. Tagname = Pumps.Consumtion
b. Point Source = RDBMS-PIINT01
c. PointType = Float32
d. Location1 = 1
e. Location4 = 1
f. ExDesc = Write a similar query after watching the video given above. Do you think you need to provide an ORDER BY clause in your query?
g. Check data on PI Data Archive. Do you see historical data? How can you import all the historical data?

Step 4 : Create a new attribute on Pumps element on AF to see the total Power. UoM is watt.



Part 2 – Collect data from **DATA.dbo.PUMPS_EFFICIENCY** table. This table shows efficiency for each pump.

Step 1 : Configure and instance of the RDBMS interface to collect data from Power table. You can follow the documentation or YouTube video (note that you will need to skip the part about security between 2:45-6:15).

a. Set Point Source as RDBMS-PIINT01 and interface ID as 2

b. Use scan frequency of 30 seconds

c. Run the service as PISCHOOL\svc-piinterface (password: student).

Step 2 : Configure new tags for PI RDBMS interface. Watch this video to configure the SQL file.

a. Tagname = PumpX.Efficiency

b. Point Source = RDBMS-PIINT01.

c. PointType = Float32

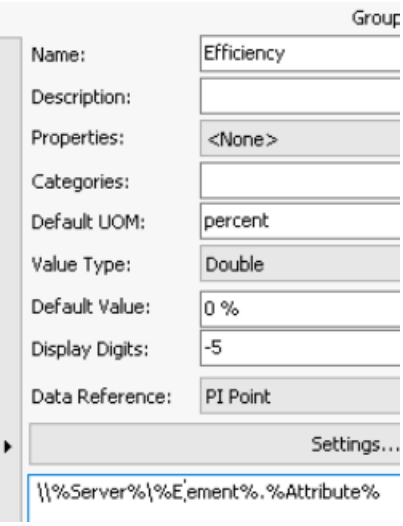
d. Location1 = 2

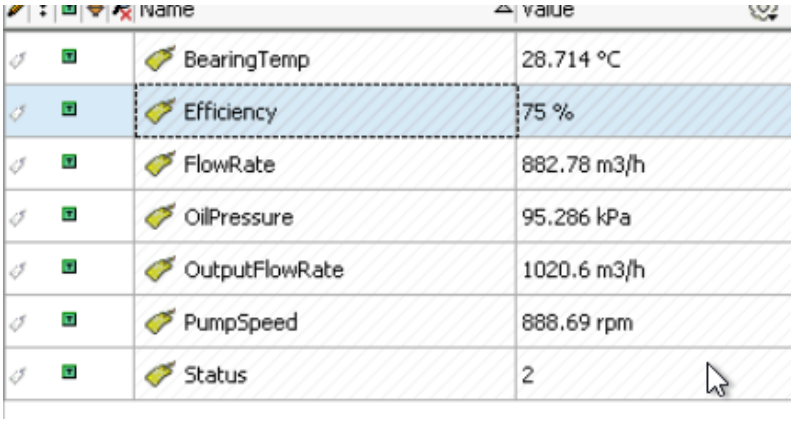
e. Location3 = depends on your query. Please refer to the video above.

f. Location4 = 1

g. Instrumenttag = sql file you created.

Step 3 : Create a new attribute in your Pump element template for Efficiency information. UoM is percent (%).





10. Appendix

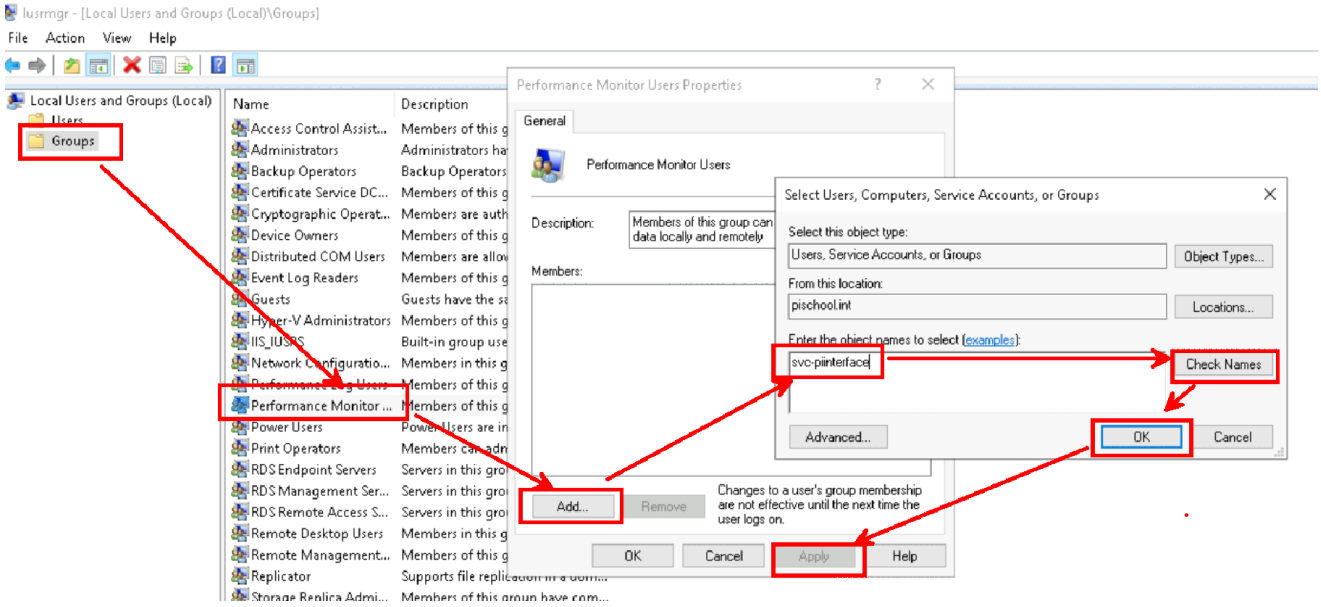
10.1 Final Exercise – Import performance data from servers

Step 1 : On PIINT01, install the PI Interface for Performance Monitor.

1. Run the installation kit (C:\Class Files\Installation Kits\PIPerfMon_x.x.x.x.exe) as administrator.
2. Click OK on extraction pop-up.
3. Click OK on Welcome page.
4. Click Next on Running service page.
5. Click Next once the installation is finished.
6. Click Close.

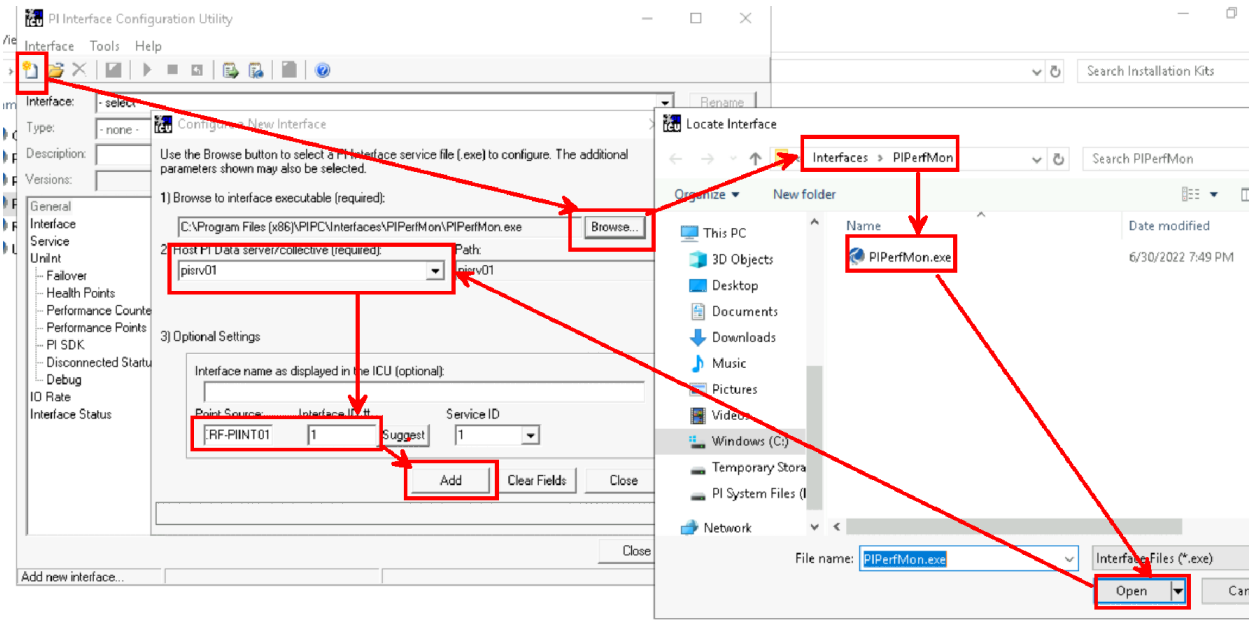
Step 2 : Prepare the permissions on the machine.

1. Search for lusrmgr.msc or “Edit Local Users and Groups” in Start menu.
2. Go to Groups > Performance Monitor Users and add svc-piinterface.

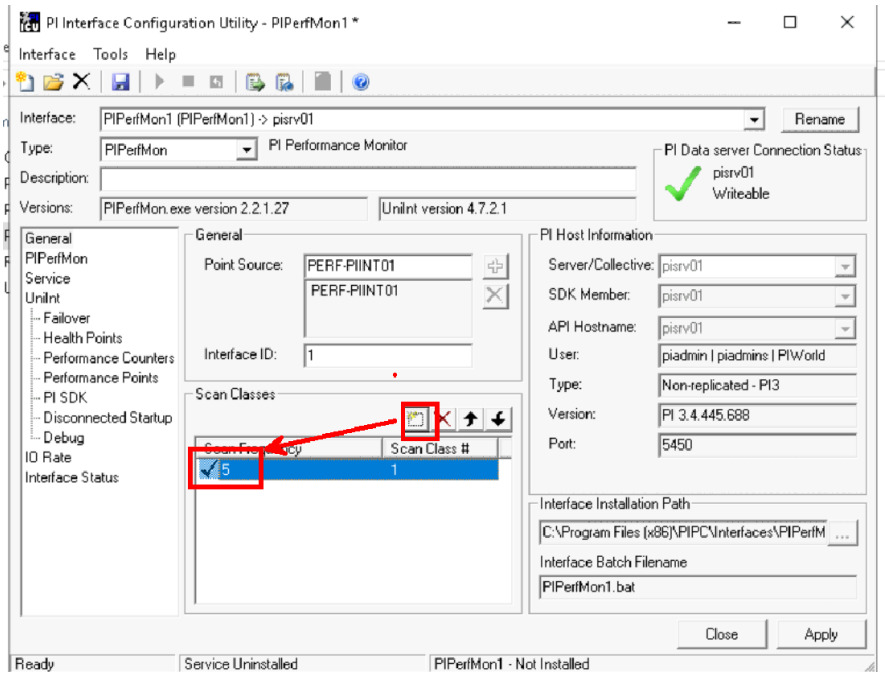


Step 3 : Configure an instance to collect local Performance Counters.

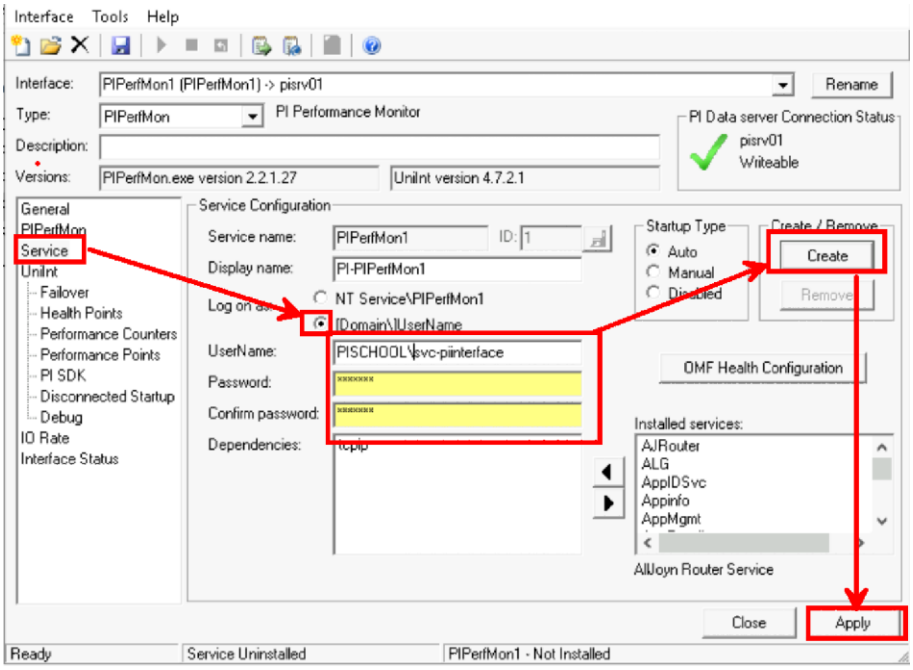
1. Open PI ICU. Click on “Create a new instance from EXE”.
2. Browse and find the PerfMon interface EXE file.
3. Select host as PISRV01. Set Point Source as **PERF-PIINT01** and interface ID as 1. Click Add.



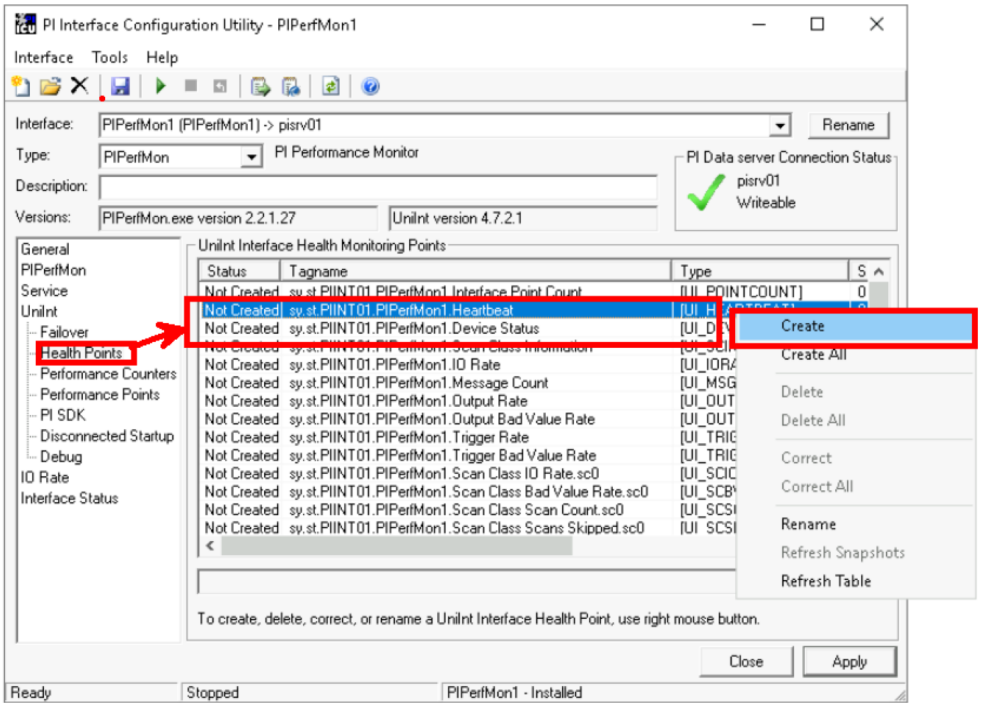
4. Create a scan frequency of 5 seconds.



5. Go to Service tab. Click on [Domain\]UserName. Enter PISCHOOL\svc-piinterface and password: student. Click on Create and Apply.

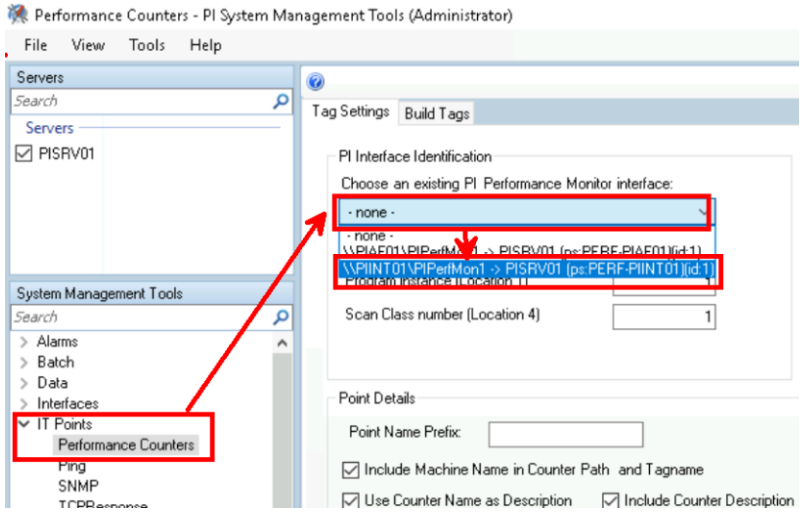


Step 4 : Create UniInt Health Tags for the PI Perfmon Interface. Device Status and Heartbeat are enough, but you can create other tags, too.

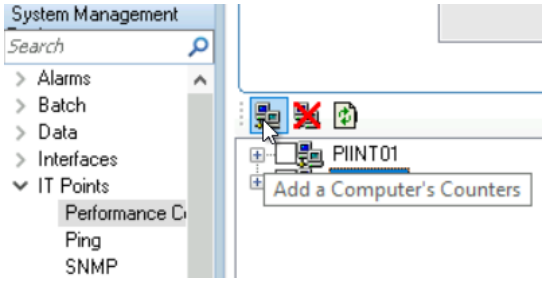


Step 5 : Create tags to collect Performance Counters.

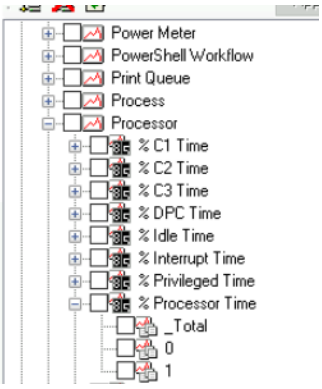
1. To create tags, go to **PISRV01 > PI SMT > IT Points > Performance counters**. On the **Tag Settings** tab, select the interface you created in Step 3.



2. Go to **Build Tags** tab on top.
3. Click on “Add a Computer’s Counters” and add PIINT01.



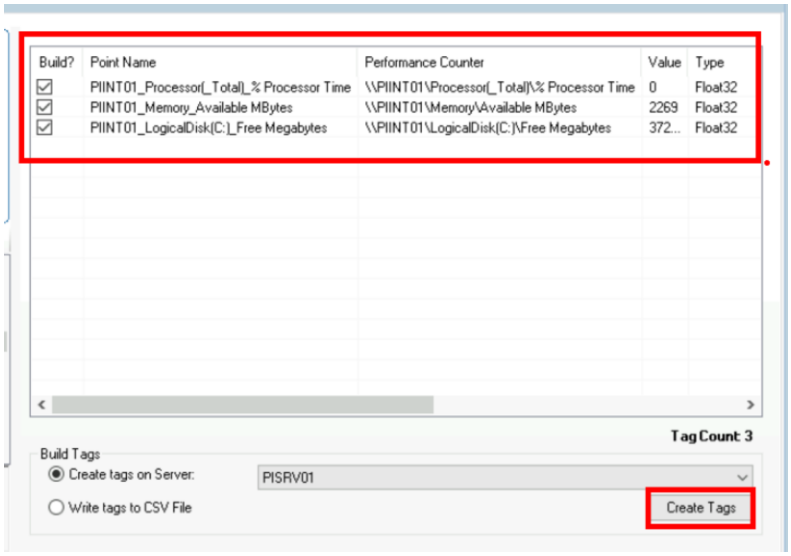
4. Once the computer is added, select the correct tags from PIINT01. We need:
   i. CPU usage (**Processor > % Processor Time > _Total**).



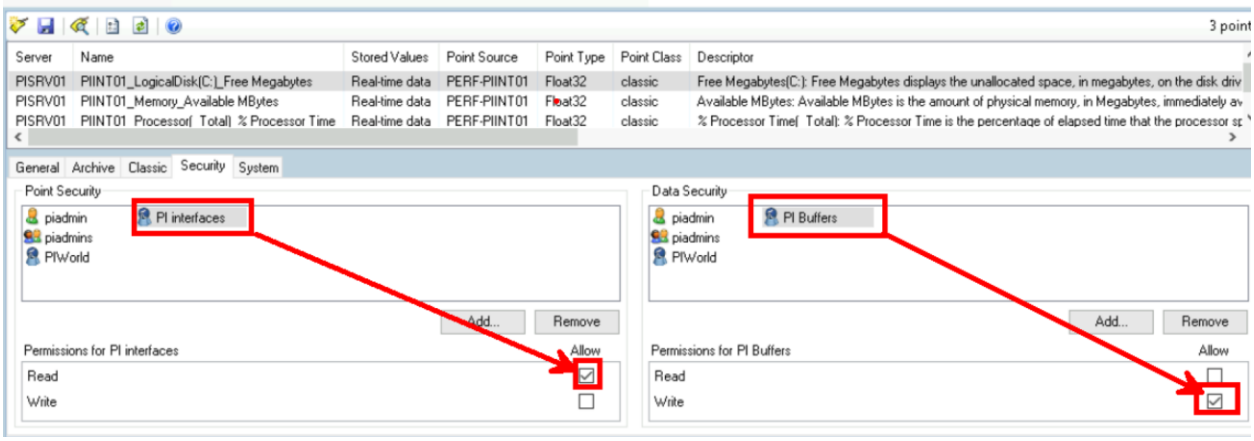
   ii. Available Memory (**Memory > Available MBytes**)

iii. Available space in C drive (**LogicalDisk > Free Megabytes > C:**).

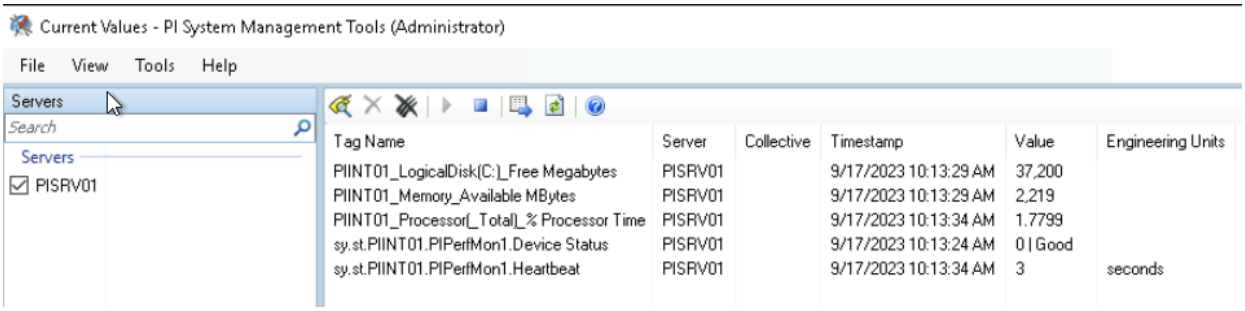
5. You should see these three listed on the right side of the screen. Click on Create tags.



6. Once the tags are created, edit their security settings to ensure the PI Interface and PI Buffer have correct permissions. You can copy the settings from previous OPC DA tags.

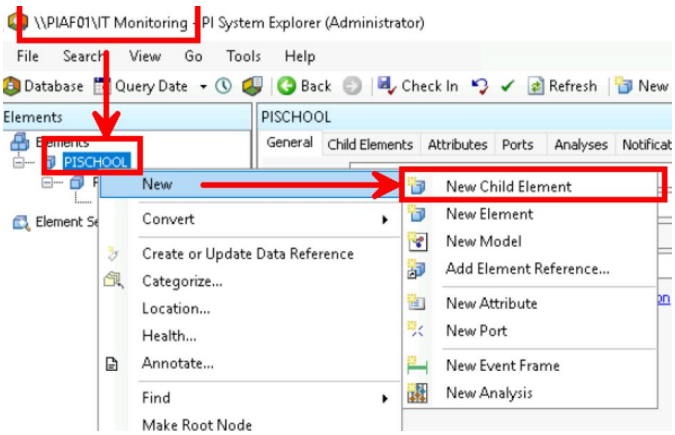


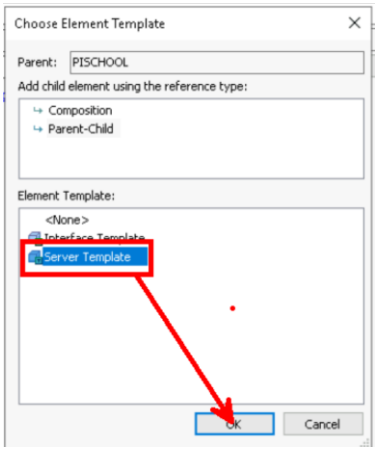
Step 6 : Start the interface and confirm that you are receiving data to PI Data Archive. You should have at least 5 tags coming from the new interface.

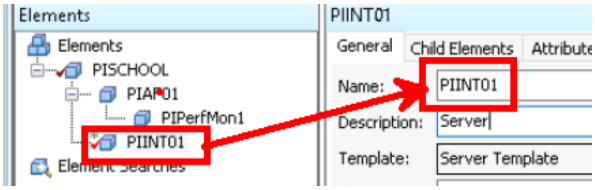


Step 7 : Create an element for PIINT01 and elements for PI Perfmon and PI OPC DA interfaces on IT Monitoring database.

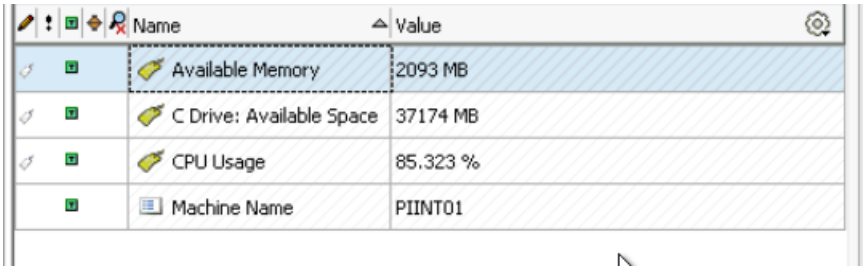
1. Create a new element under PISCHOOL from “Server template” and call it PIINT01.



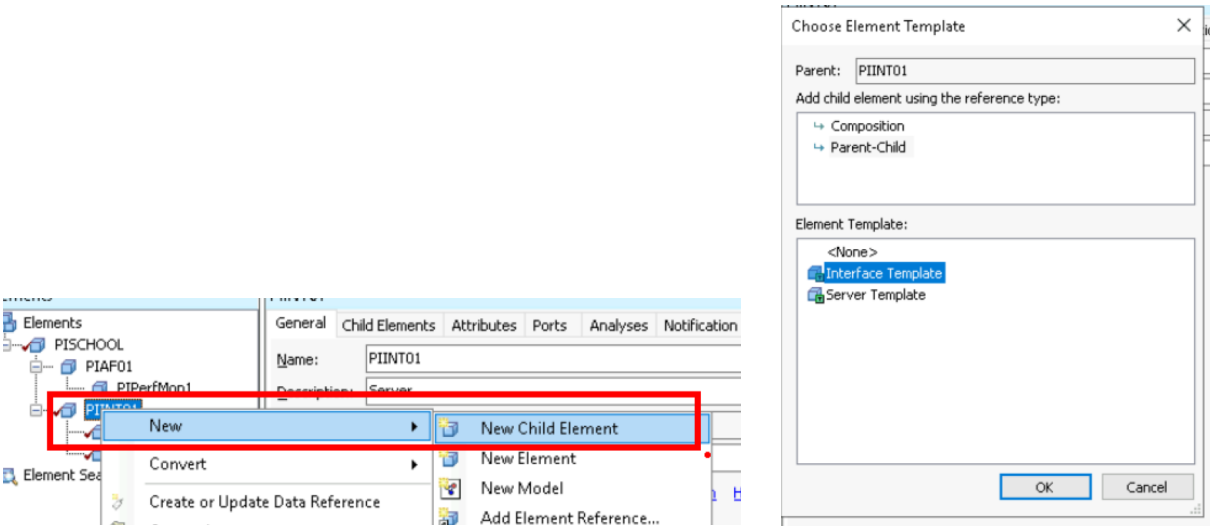


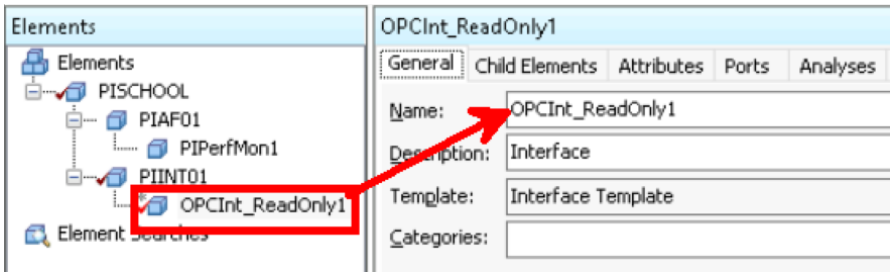


2. Check in your changes. Go to Attributes tab and confirm that all 4 attributes show data.

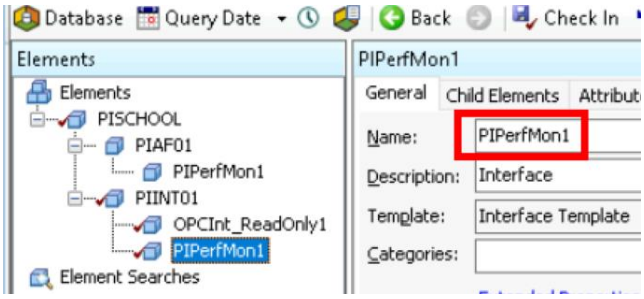


Step 8 : Create a new element under PIINT01 from “Interface template” and call it OPCInt_ReadOnly1. Check in your changes.



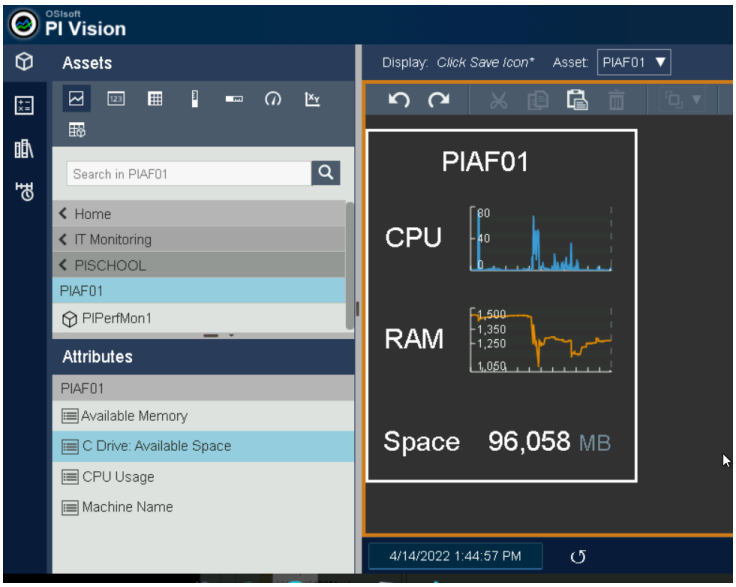


Step 9 : Create another element under PIINT01 from “Interface template” and call it PIPerfMon1. Check in your changes.



Step 10 : Confirm that all the attributes are showing data for these elements.

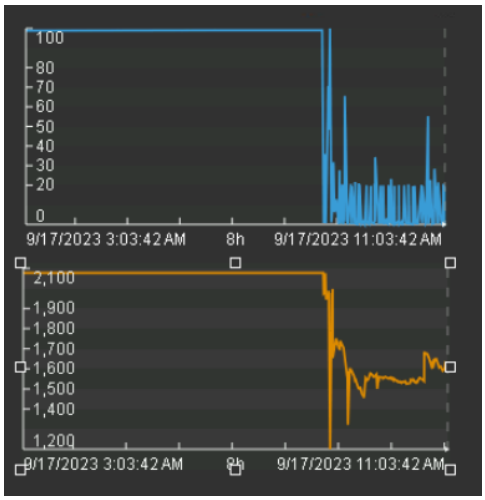
Step 11 : Create a PI Vision display to monitor the servers and another display to monitor interfaces.



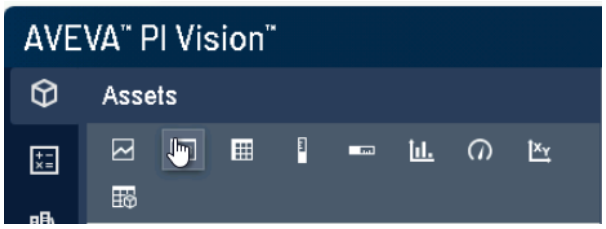
1. Open PI Vision. Click on “New Display” on top right menu.
2. On the left menu, drill to PIAF01.



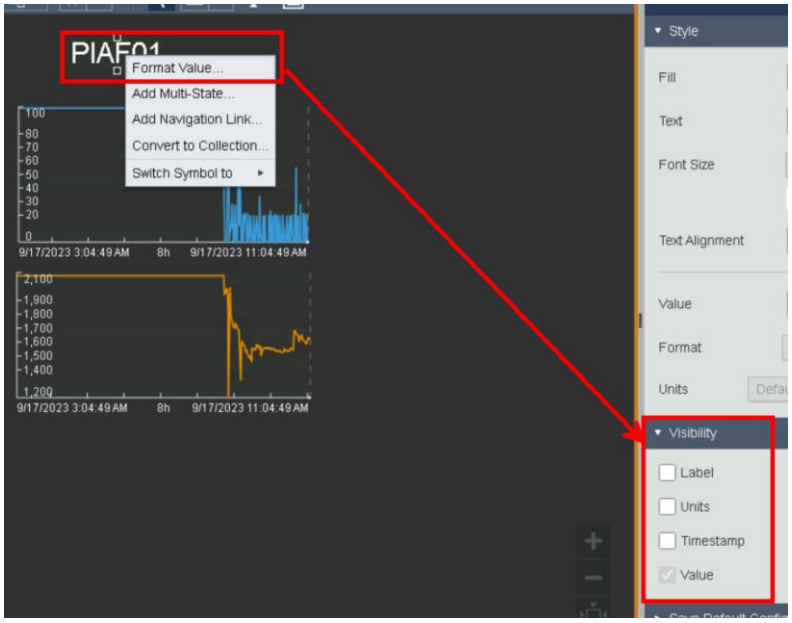
3. From the attributes menu, select CPU usage and drag to the screen.
4. Select Available memory and drag it below CPU trend.



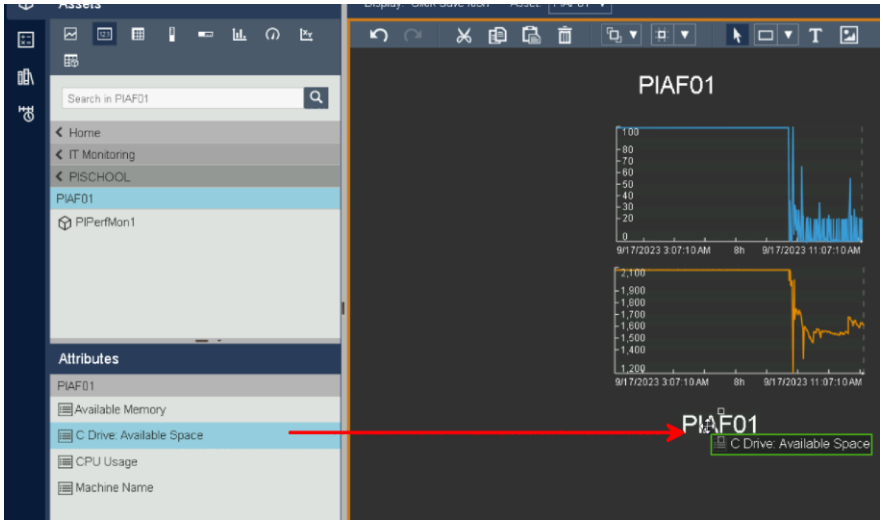
5. On the top menu, select Value symbol.



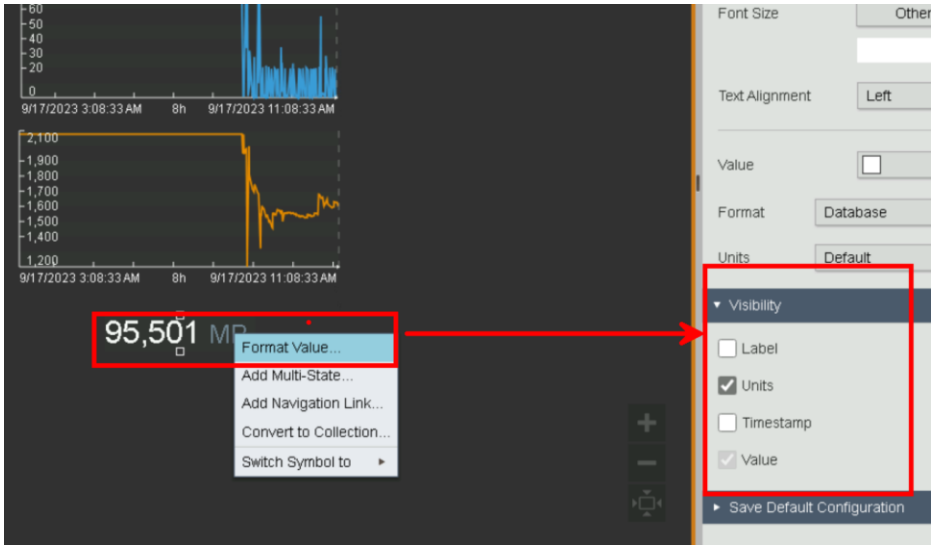
6. Select Machine name attribute and place it above the CPU trend. Right click and go to Format value to show only the Value. Increase the font size.



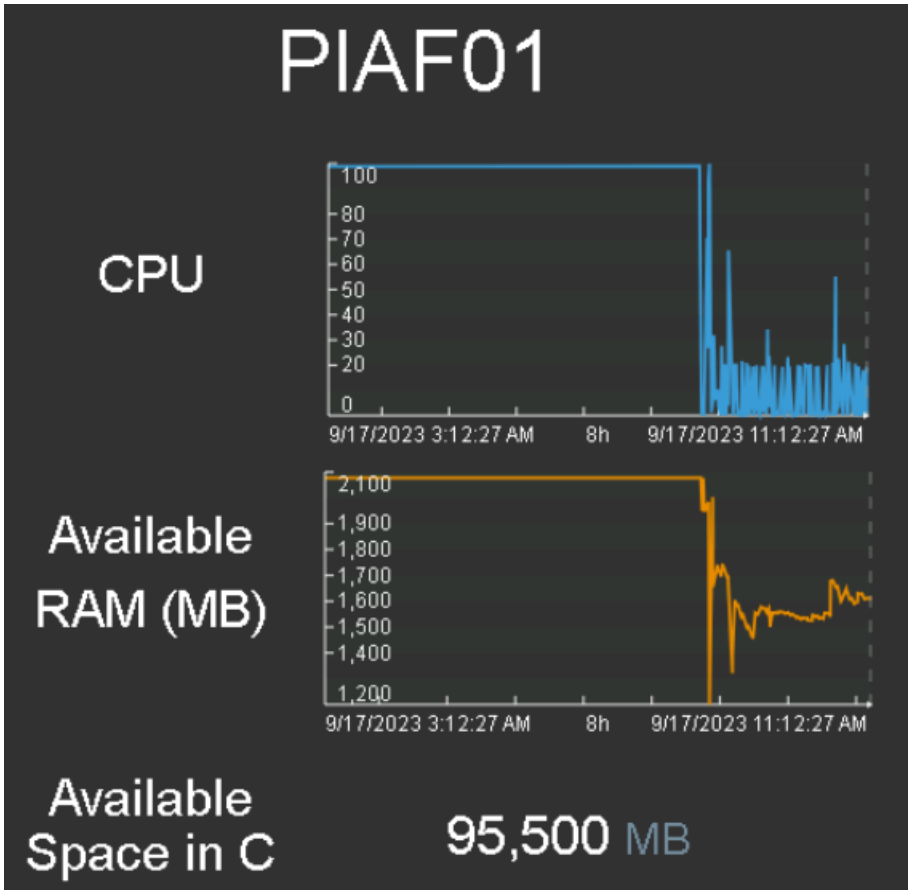
7. Copy the Value you just created and paste it below the RAM trend. From the attributes pane, drag “C Drive” on to the newly created symbol.



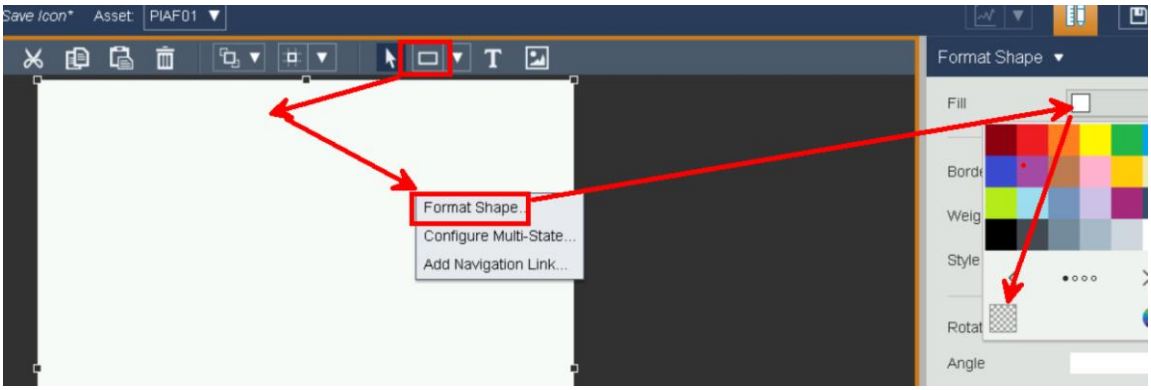
8. It should now show the value. Add the Units from Format value pane.

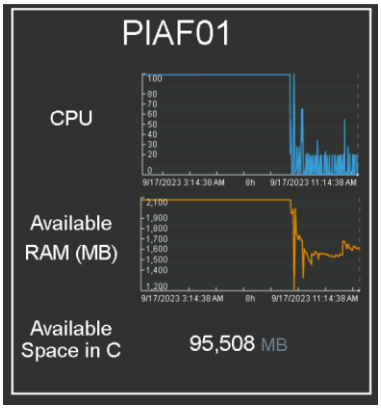


9. Add text boxes next to each of these symbols.

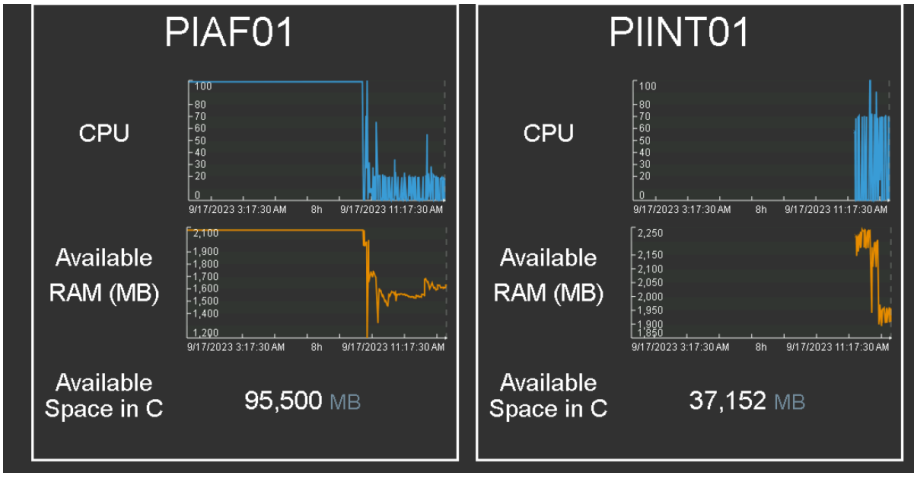


10. Draw a rectangle covering all the objects. Make the fill transparent.



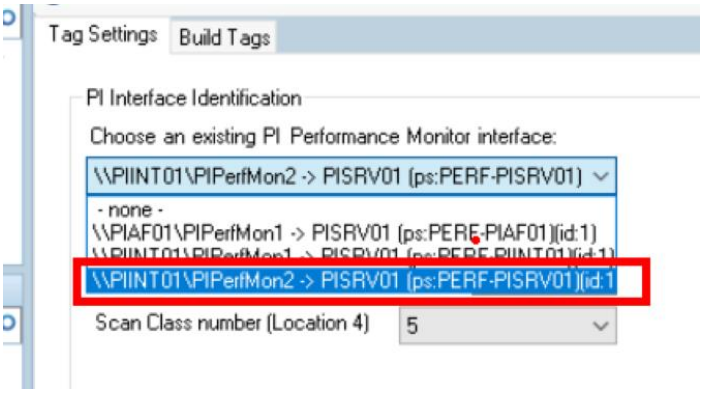


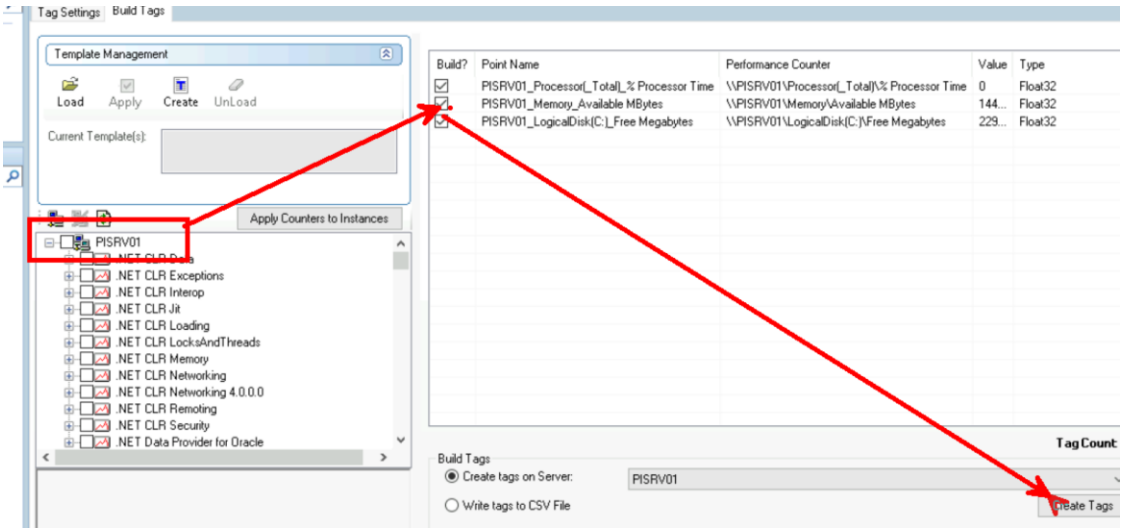
11. Select everything on the screen. Right click on the object and click on “Convert to Collection. Adjust the size of the Collection to fit the PIINT01 on the right.



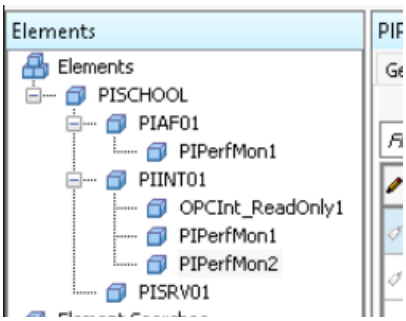
For Bonus part, do everything the same way as above except:

- Create the interface instance on PIINT01.
- Use Points source PERF-PISRV01
- When creating PI Points on PI SMT, expand the Counters from PISRV02 (not PIINT01).





In the AF hierarchy, add PISRV01 as a server and PIPerfMon2 as an interface under PIINT01.



10.2 Final Exercise – Import data from text files

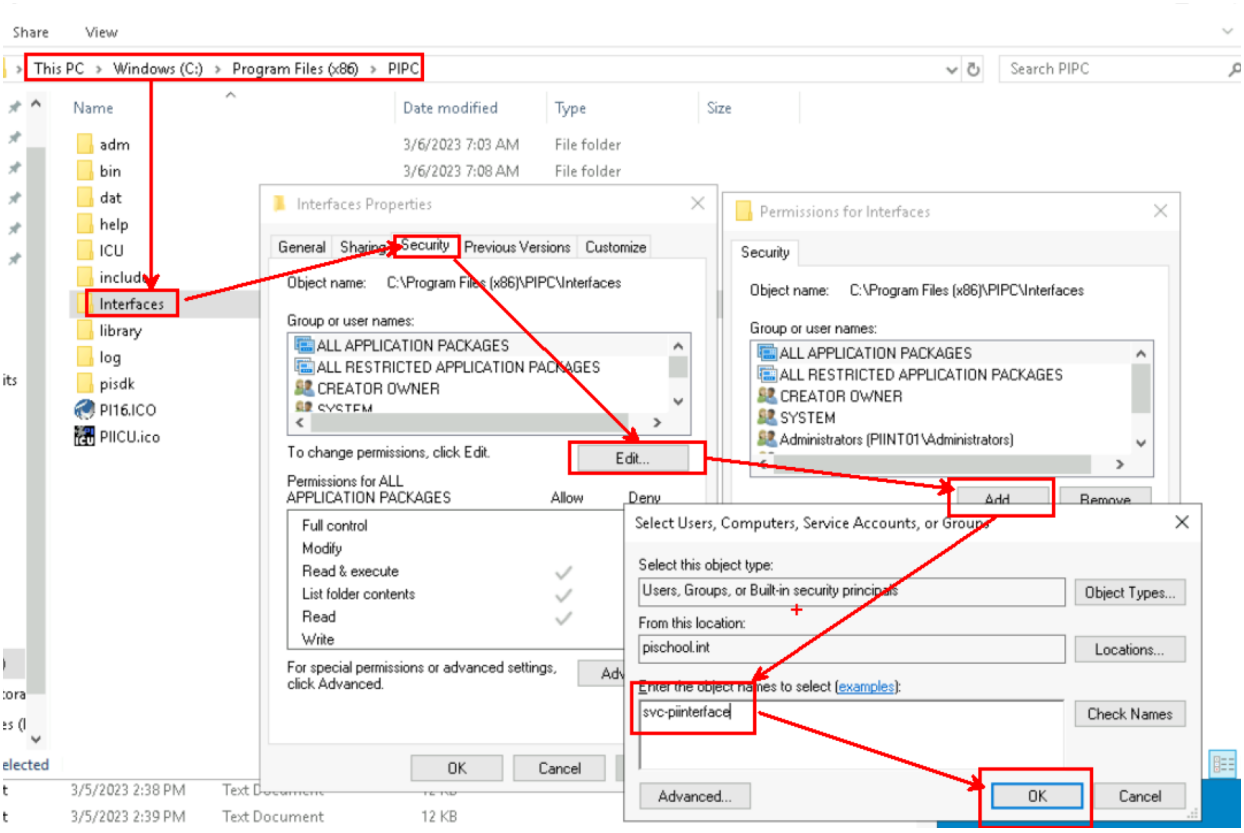
Part 1 – Collect data from **C:\Class Files\Data\Exercise1** folder.

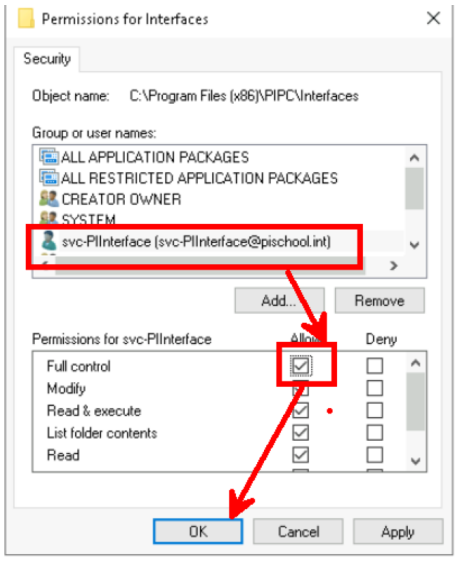
Step 1 : On PIINT01, install the PI Interface for Universal File and Stream Loading (UFL).

1. Run the installation kit (C:\Class Files\Installation Kits\UFL_x.x.x.x.exe) as administrator.
2. Click OK on extraction pop-up.
3. Click OK on Welcome page.
4. Click Next on Running services page.
5. Click Next once the installation is finished.
6. Click Close.

Step 2 : Prepare the permissions on the machine.

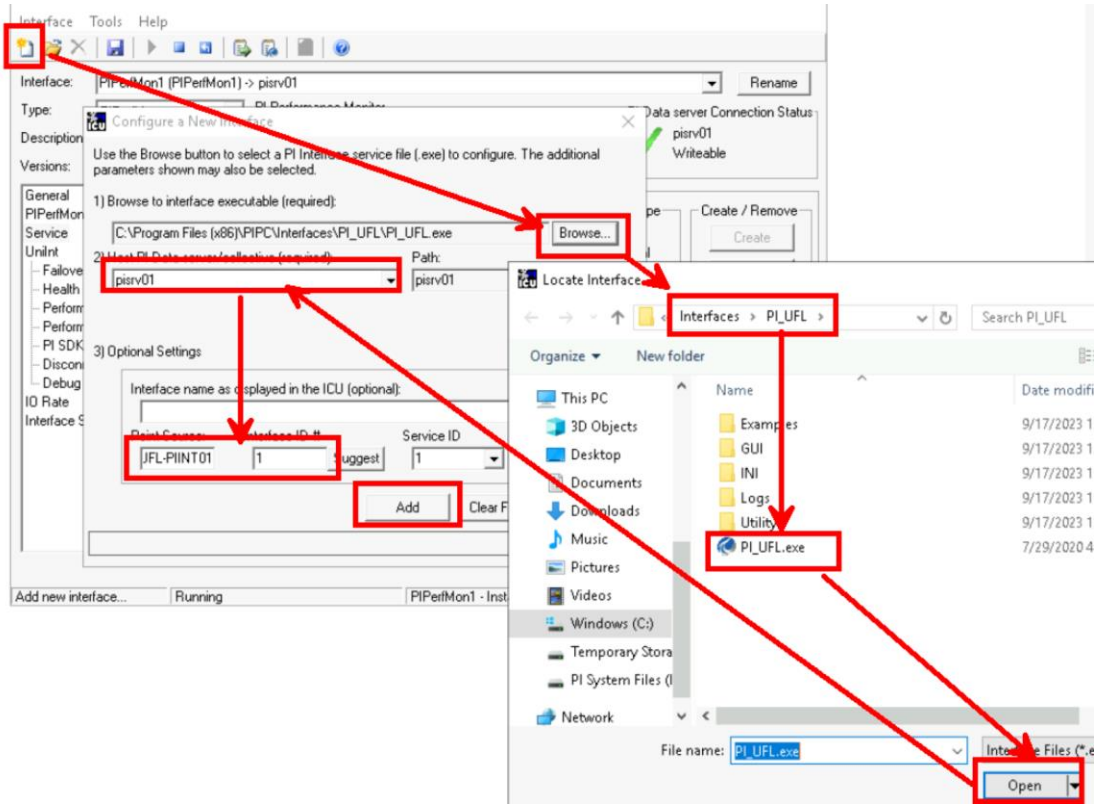
1. Go to folder C:\Program Files (x86)\PIPC\Interfaces and provide full access for the user PISCHOOL\svc-piinterface.



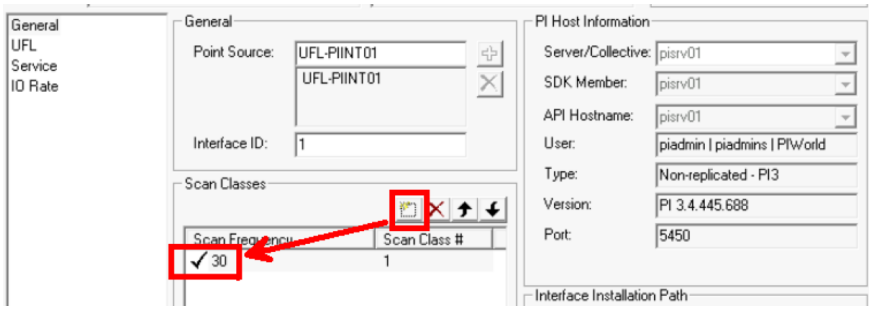


Step 3 : Configure and instance of the UFL interface to collect data from Exercise1 folder.

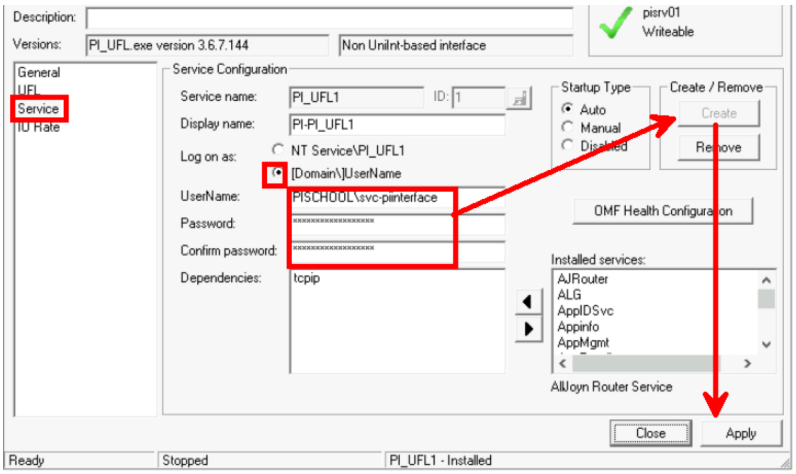
1. Open PI ICU. Click on “Create a new instance from EXE”.
2. Browse and find the UFL interface EXE file.
3. Select host as PISRV01. Set Point Source as **UFL-PIINT01** and interface ID as 1. Click Add.



4. Create a scan frequency of 30 seconds.

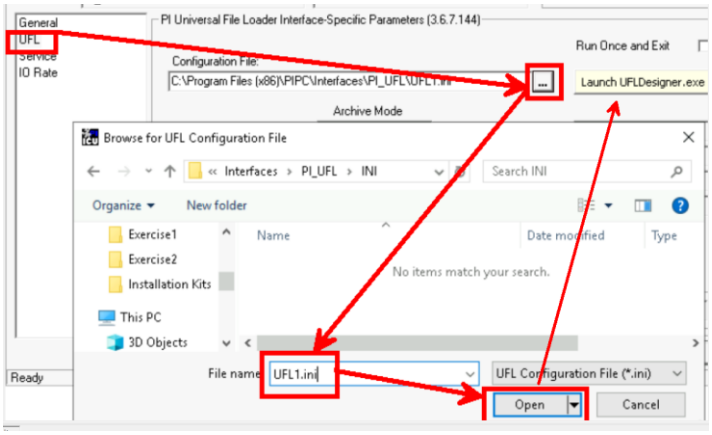


5. Go to Service tab. Click on [Domain\]UserName. Enter PISCHOOL\svc-piinterface and password: student. Click on Create and Apply.



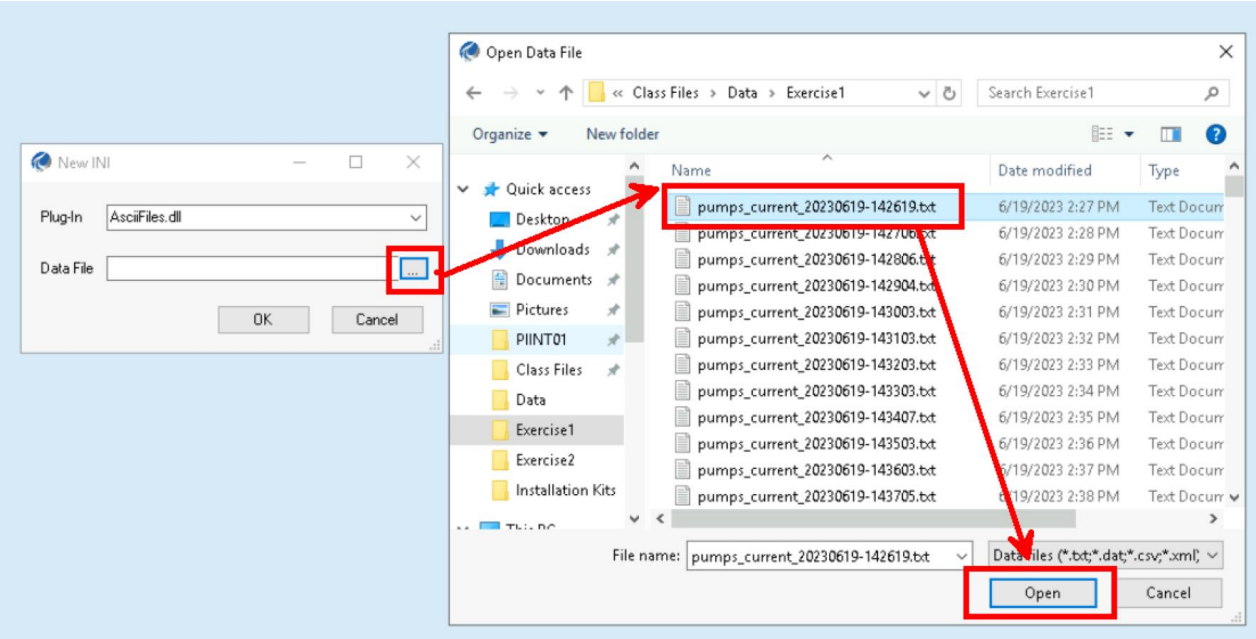
Step 4 : Configure the ini file.

1. Go to UFL Tab and create a new ini file as shown below.

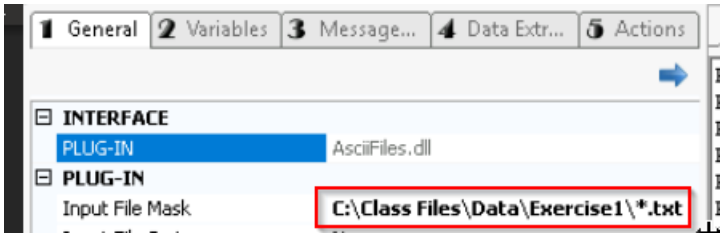


2. Click on Launch UFLDesigner.exe.
3. Click Yes on the warning pop-up.

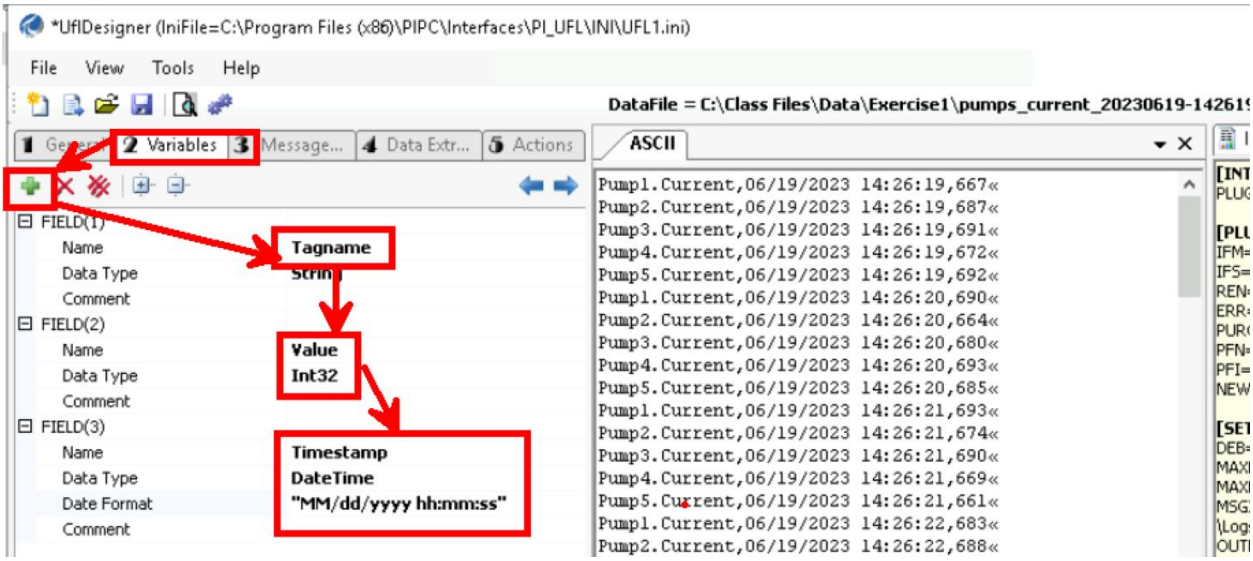
4. Select any file from Exercise1 folder for the next pop-up.



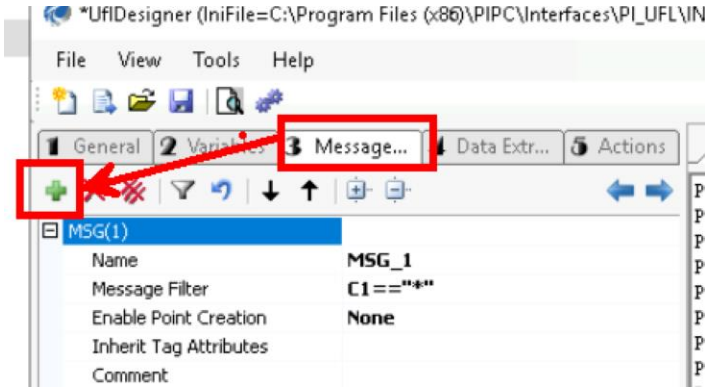
5. Change the Input File Mask to **C:\Class Files\Data\Exercise1*.txt**



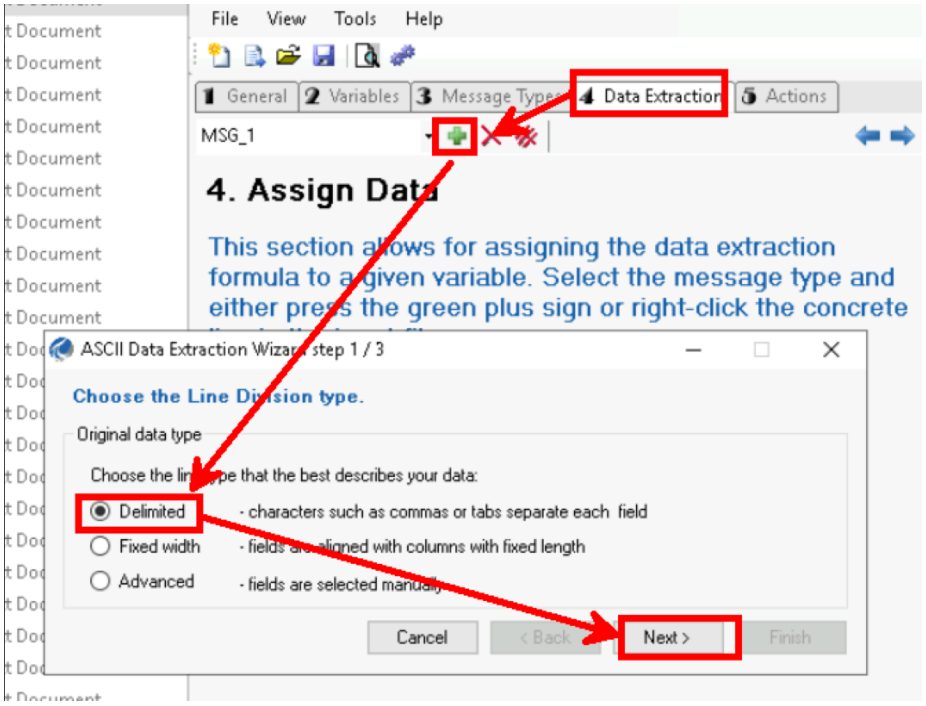
6. Click on Variables and add three variables as shown below.



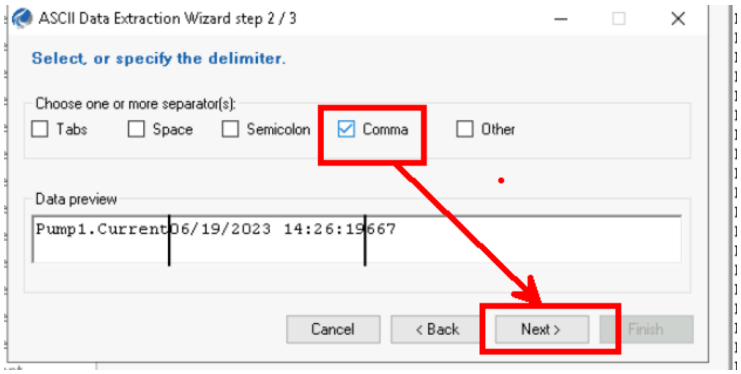
7. Click on Message Types and add a new Message Type.



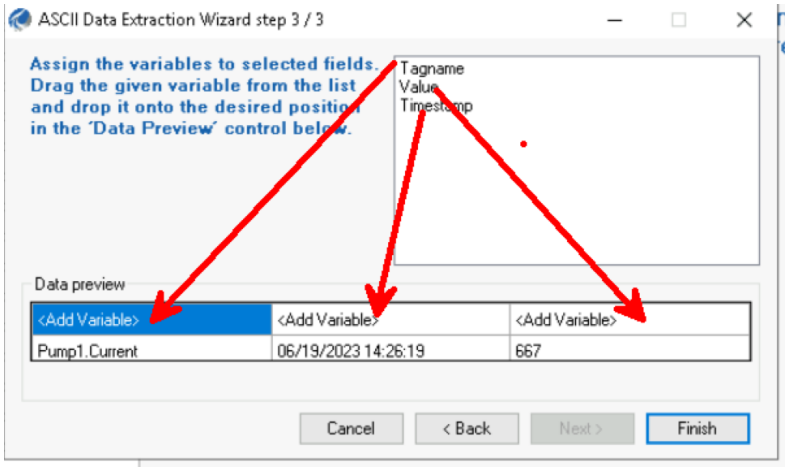
8. Click on Data Extraction and add a new rule. In the pop-up, select Delimited.



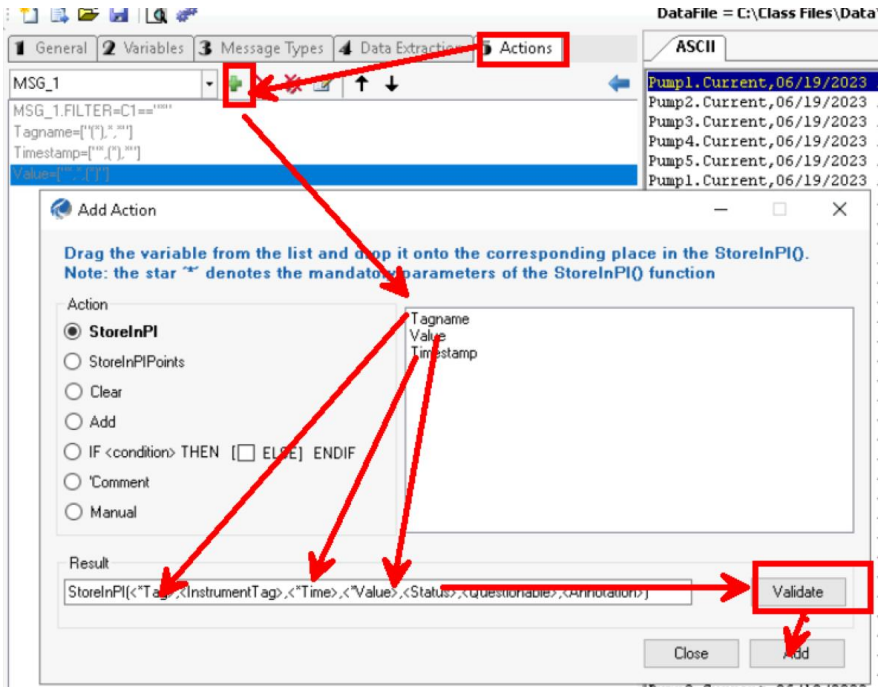
9. Select Comma and click Next.



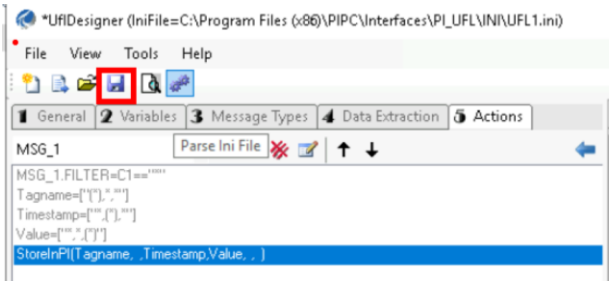
10. Drag and drop the variables to the corresponding boxes. Click Finish.



11. Click on Actions and add an action. Drag and drop the variables to the corresponding boxes. Click Validate > Add > Close.



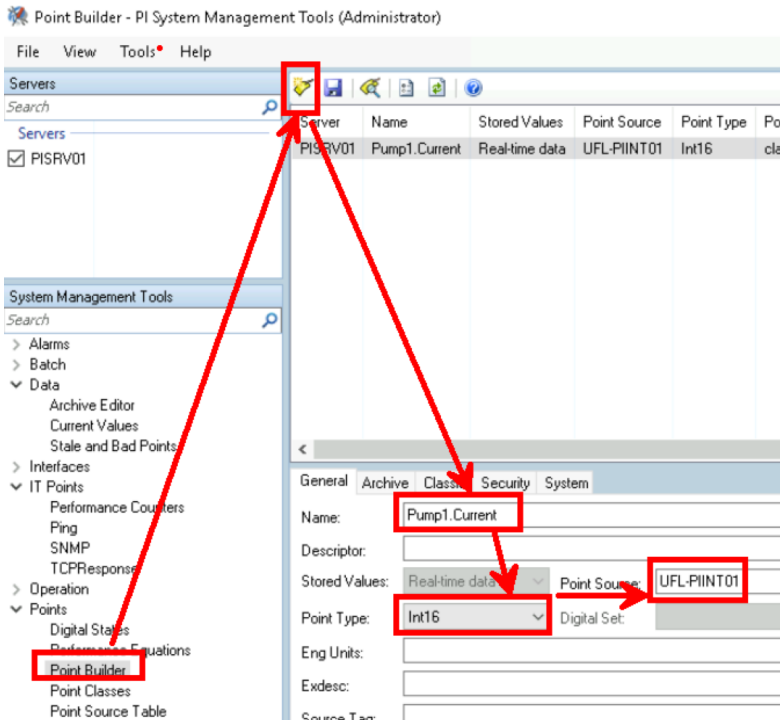
12. Click on Save to save the ini file. Close the UFLDesigner.exe.

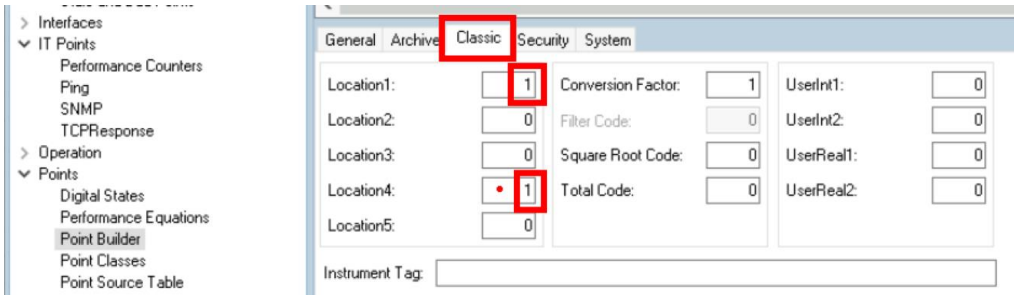


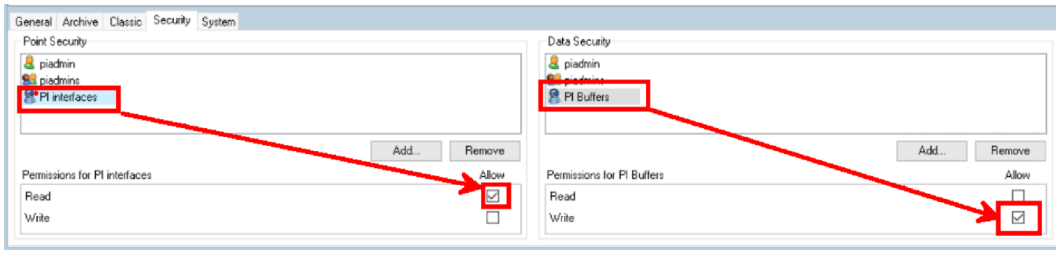
13. Run the interface from PI ICU.

Step 5 : Configure 5 new tags for PI UFL interface.

1. Tagname = PumpX.Current
2. Point Source = UFL-PIINT01.
3. PointType = Int16
4. Location1 = 1





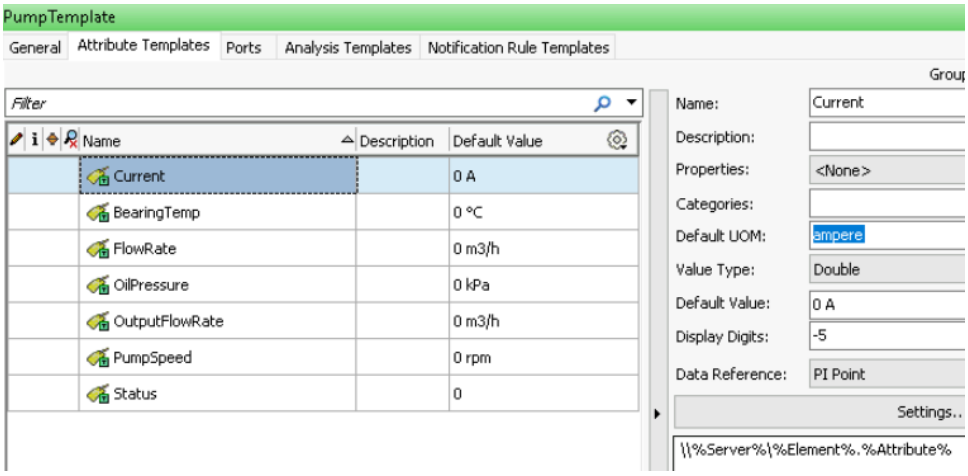


5. You can use PI Builder to create the remaining 4 tags.

Server	Name	Stored Values	Point Source	Point Type	Point Class	Descriptor	Point Security	Data Security
PISRV01	Pump1.Current	Real-time data	UFL-PIINT01	Int16	classic		piadmin: A(r,w) | piadmins: A(r,w) | PI interfaces: A(r)	piadmin: A(r,w) | piadmins: A(r,w) | PI Buffers: A(w)
PISRV01	Pump2.Current	Real-time data	UFL-PIINT01	Int16	classic		piadmin: A(r,w) | piadmins: A(r,w) | PI interfaces: A(r)	piadmin: A(r,w) | piadmins: A(r,w) | PI Buffers: A(w)
PISRV01	Pump3.Current	Real-time data	UFL-PIINT01	Int16	classic		piadmin: A(r,w) | piadmins: A(r,w) | PI interfaces: A(r)	piadmin: A(r,w) | piadmins: A(r,w) | PI Buffers: A(w)
PISRV01	Pump4.Current	Real-time data	UFL-PIINT01	Int16	classic		piadmin: A(r,w) | piadmins: A(r,w) | PI interfaces: A(r)	piadmin: A(r,w) | piadmins: A(r,w) | PI Buffers: A(w)
PISRV01	Pump5.Current	Real-time data	UFL-PIINT01	Int16	classic		piadmin: A(r,w) | piadmins: A(r,w) | PI interfaces: A(r)	piadmin: A(r,w) | piadmins: A(r,w) | PI Buffers: A(w)

Step 6 : Create a new attribute in your Pump element template for Current information.

1. Go to Pump Assets Database.
2. Open Pump template from the Library.
3. Create a new attribute and name it Current.
4. UoM is Ampere (A).
5. Data Reference is PI Point and tag name is %Element%.%Attribute% .



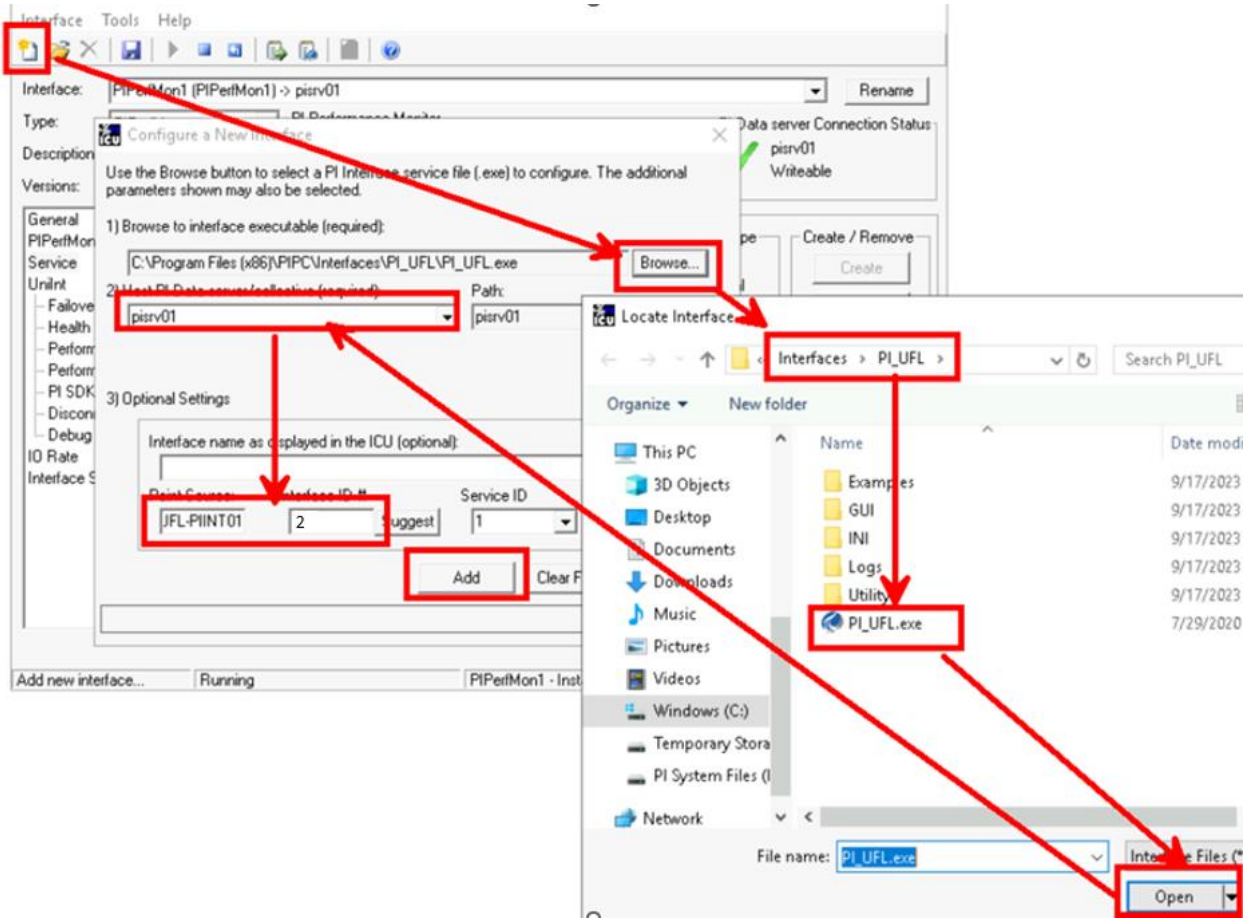
Part 2 – Collect data from **C:\Class Files\Data\Exercise2** folder. These files contain efficiency data coming from a third-party application.

Step 1 : Configure and instance of the UFL interface to collect data from Exercise1 folder. You can follow the documentation or the YouTube playlist.

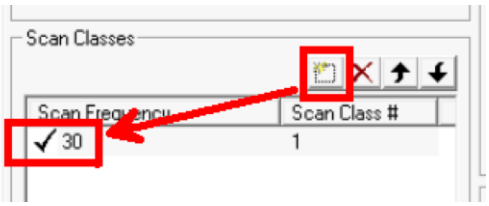
a. Open PI ICU. Click on “Create a new instance from EXE”.

b. Browse and find the UFL interface EXE file.

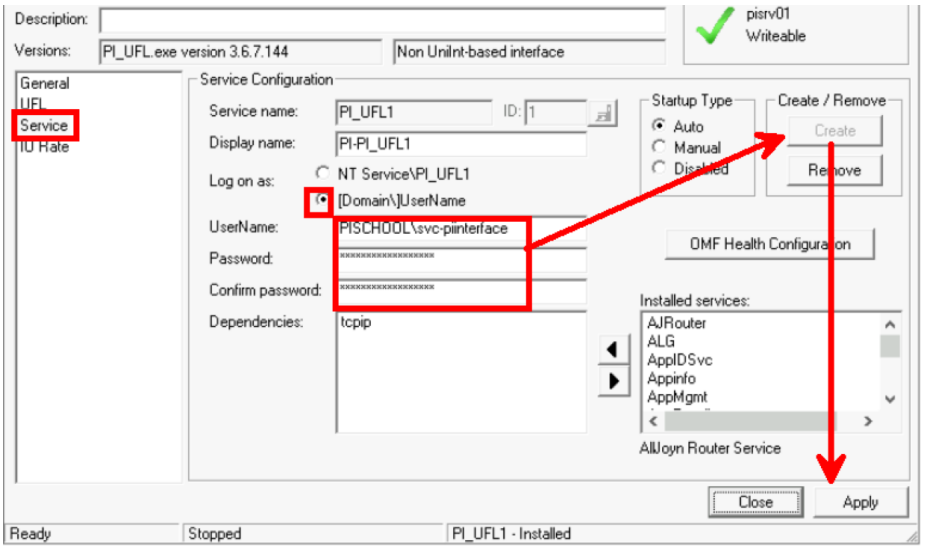
c. Select host as PISRV01. Set Point Source as **UFL-PIINT01** and interface ID as 2. Click Add.



d. Create a scan frequency of 30 seconds.

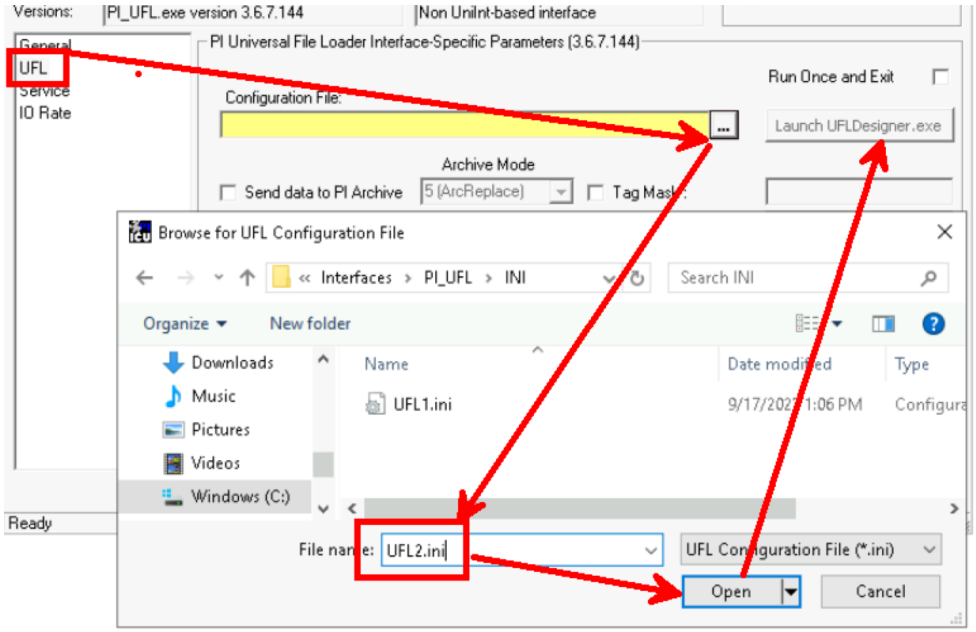


e. Go to Service tab. Click on [Domain\]UserName. Enter PISCHOOL\svc-piinterface and password: student. Click on Create and Apply.



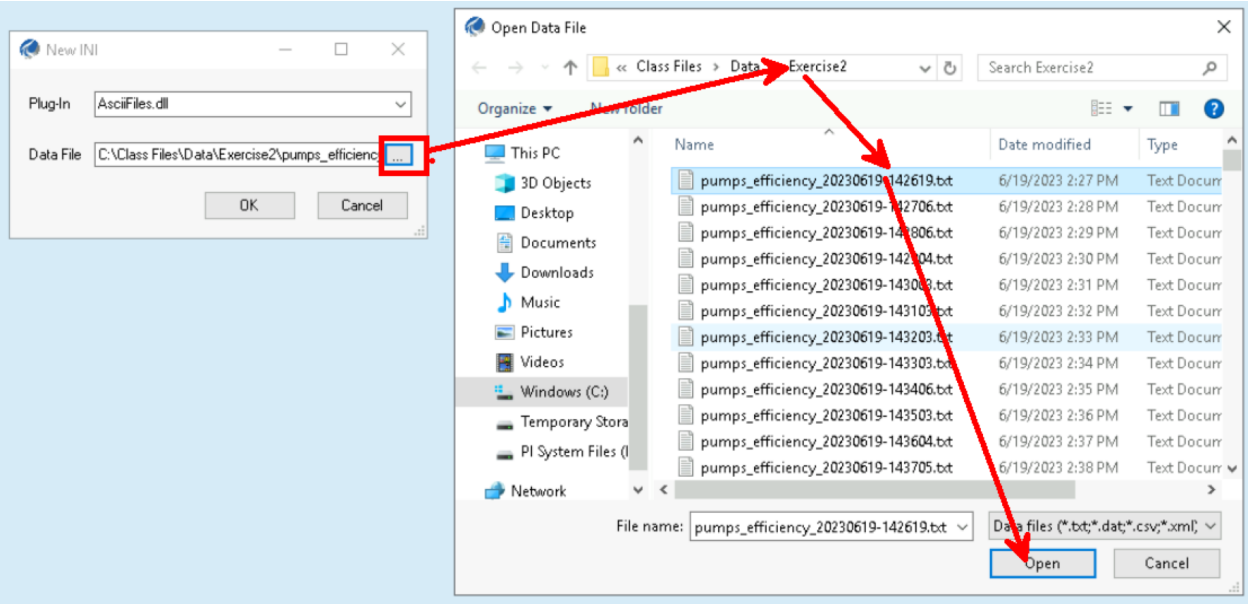
Step 2 : Configure the ini file.

1. Go to UFL Tab and create a new ini file as shown below.

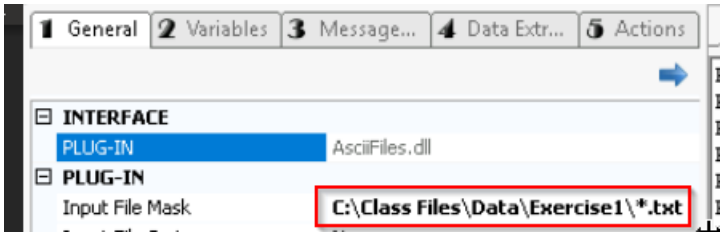


2. Click on Launch UFLDesigner.exe.
3. Click Yes on the warning pop-up.

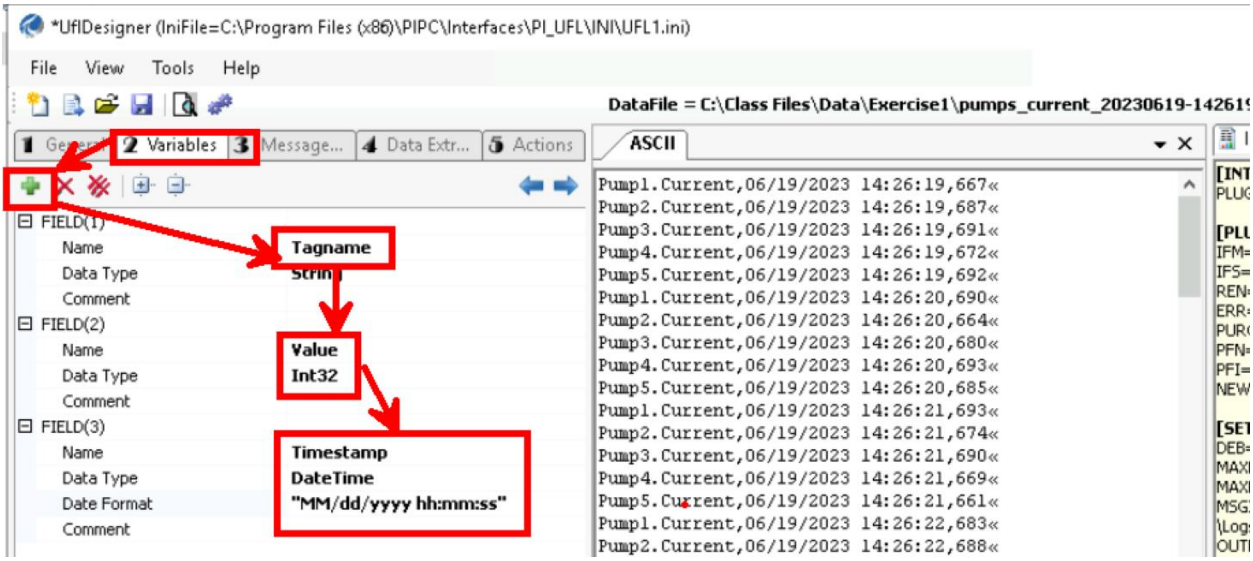
4. Select any file from Exercise2 folder for the next pop-up.



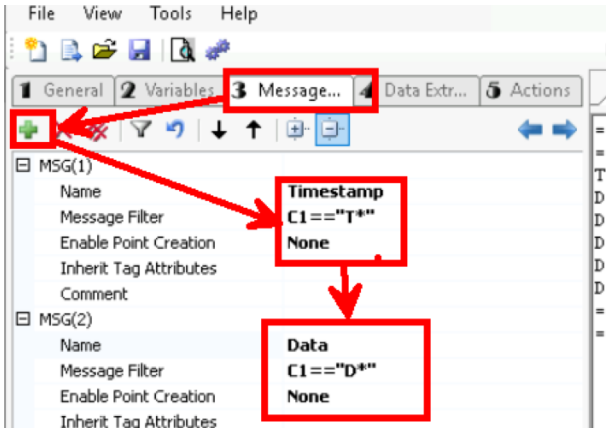
5. Change the Input File Mask to C:\Class Files\Data\Exercise1*.txt



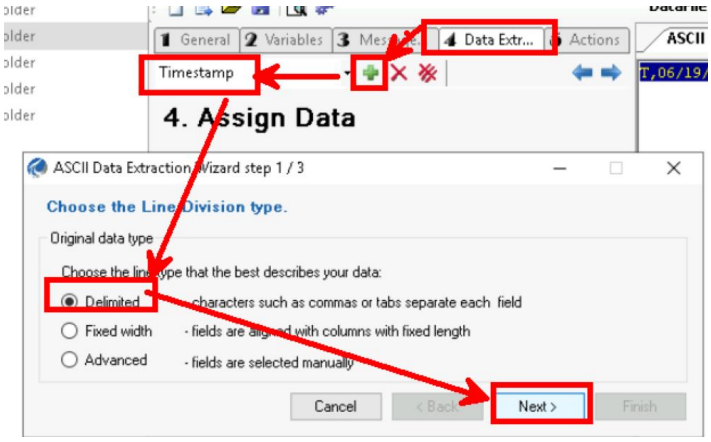
6. Click on Variables and add three variables as shown below.



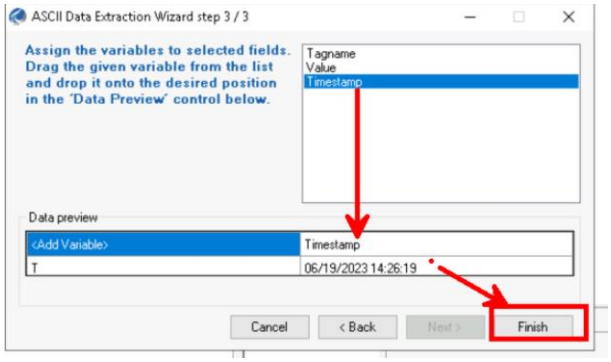
7. Click on Message Types and add two message types. One starts with T* and the other with D*.



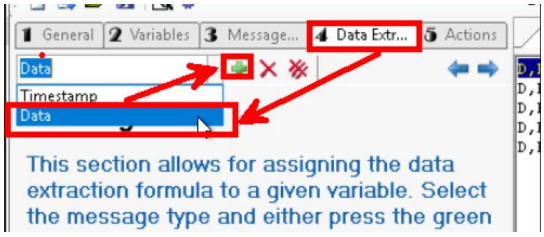
8. Go to Data Extraction and add a new rule for Timestamp Message Type.



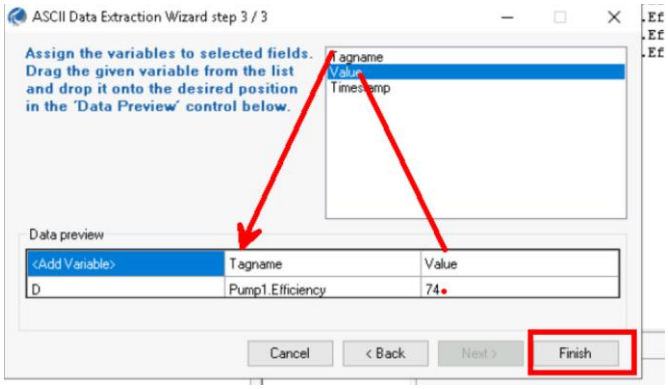
9. Select Comma in the next menu and click Next.
10. Drag and drop Timestamp into the corresponding box.



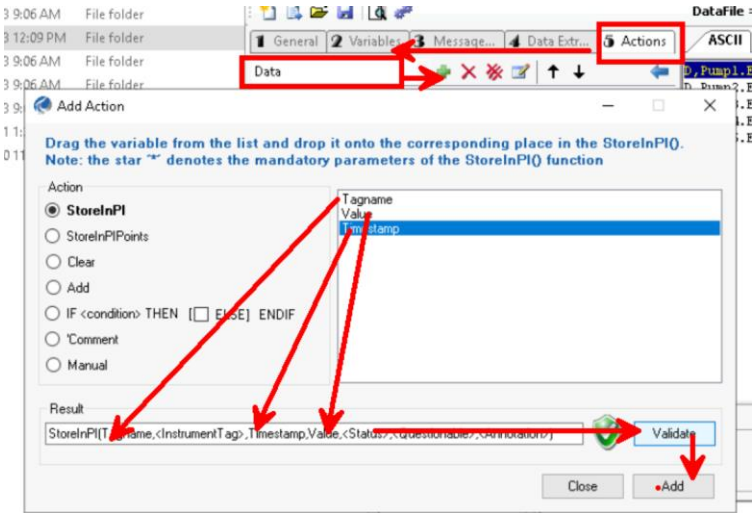
11. Change the message type to Data and add a new rule.



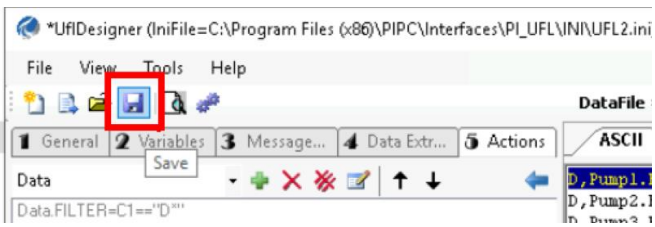
12. Drag Tagname and Value to the corresponding boxes.



13. Click on Actions tab. While data is selected add an action. Drag variables into positions as shown below. Click Validate and Add.



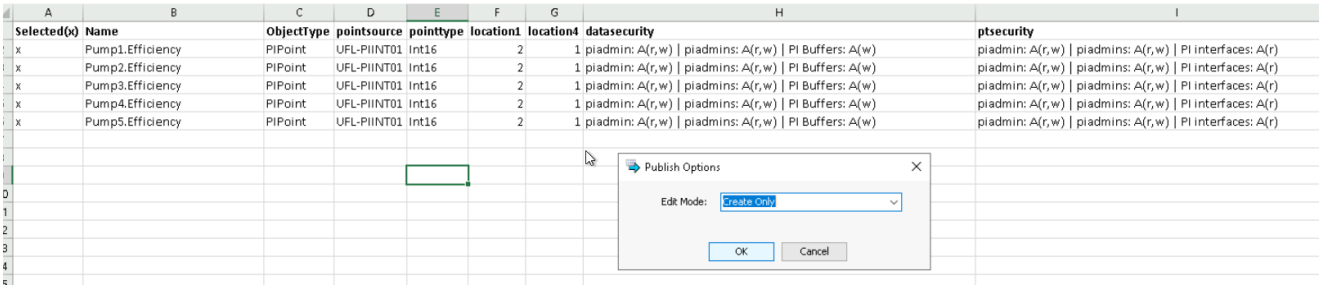
14. Save the ini file.



15. Run the interface.

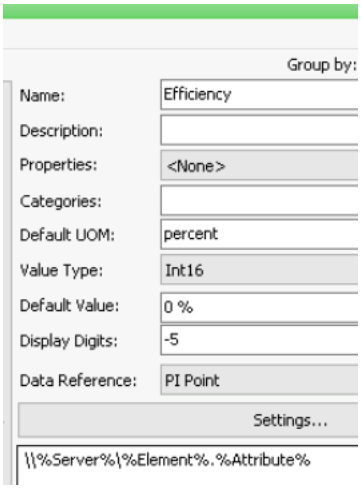
Step 3 : Configure 5 new PI Points for PI UFL interface. You can edit the Excel file you created earlier to create these PI Points.

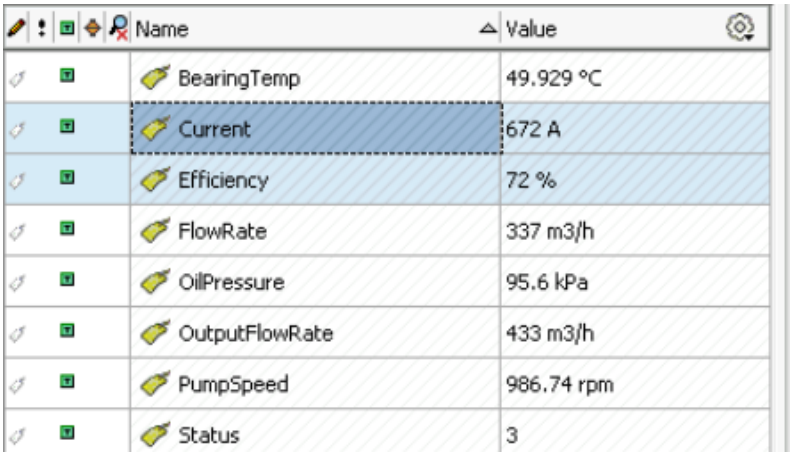
6. Tagname = PumpX.Efficiency
7. Point Source = UFL-PIINT01.
8. PointType = Int16
9. Location1 = 2
10. Location4 = 1.



Selected(x)	Name	ObjectType	pointsource	pointtype	location1	location4	datasecurity	ptsecurity
x	Pump1.Efficiency	PIPoint	UFL-PIINT01	Int16	2	1	piadmin: A(r,w) | piadmins: A(r,w) | PI Buffers: A(w)	piadmin: A(r,w) | piadmins: A(r,w) | PI interfaces: A(r)
x	Pump2.Efficiency	PIPoint	UFL-PIINT01	Int16	2	1	piadmin: A(r,w) | piadmins: A(r,w) | PI Buffers: A(w)	piadmin: A(r,w) | piadmins: A(r,w) | PI interfaces: A(r)
x	Pump3.Efficiency	PIPoint	UFL-PIINT01	Int16	2	1	piadmin: A(r,w) | piadmins: A(r,w) | PI Buffers: A(w)	piadmin: A(r,w) | piadmins: A(r,w) | PI interfaces: A(r)
x	Pump4.Efficiency	PIPoint	UFL-PIINT01	Int16	2	1	piadmin: A(r,w) | piadmins: A(r,w) | PI Buffers: A(w)	piadmin: A(r,w) | piadmins: A(r,w) | PI interfaces: A(r)
x	Pump5.Efficiency	PIPoint	UFL-PIINT01	Int16	2	1	piadmin: A(r,w) | piadmins: A(r,w) | PI Buffers: A(w)	piadmin: A(r,w) | piadmins: A(r,w) | PI interfaces: A(r)

Step 4 : Create a new attribute in your Pump element template for Efficiency information. UoM is percent (%). Confirm that you are getting data for both attributes.





10.3 Final Exercise – Import data from SQL Server

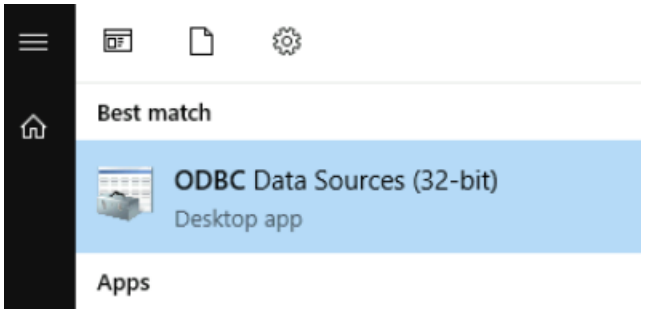
Part 1 – Collect data from **DATA.dbo.POWER** table. This table shows total power consumption for all the pumps.

Step 1 : On PIINT01, install the PI Interface for Relation Database Management Systems (RDBMS). Click OK or Next when prompted.

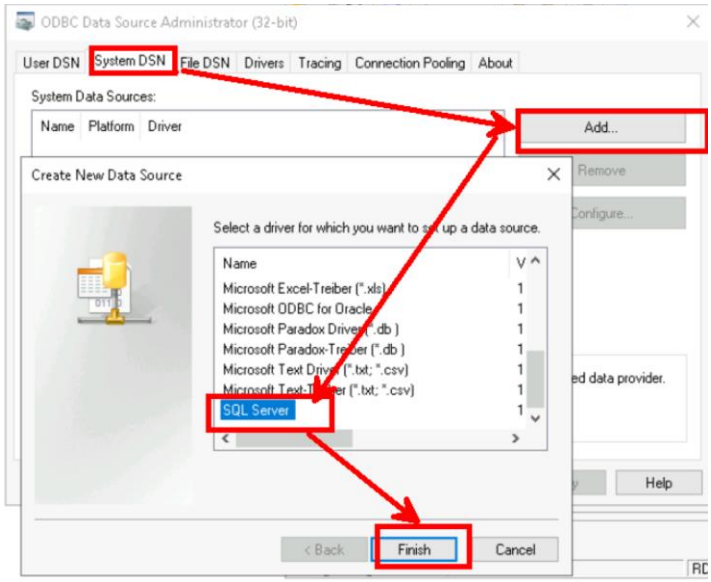
1. Installation kit is in C:\Class Files\Installation Kits\RDBMSPI_x.x.x.x.exe.
2. Once asked, reboot the machine.

Step 2 : Configure a DSN for PIAF01.

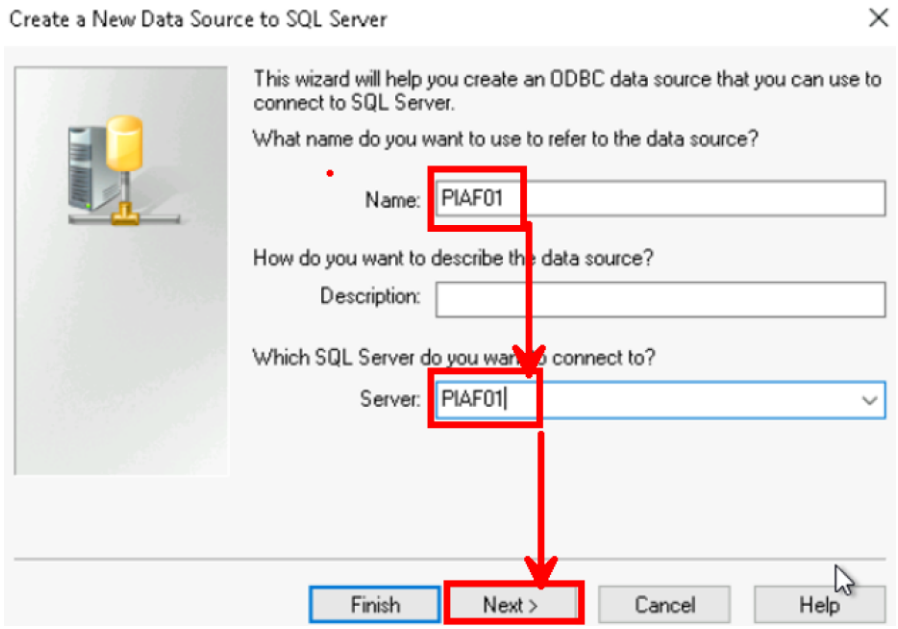
1. Open ODBC Data Sources (32-bit)



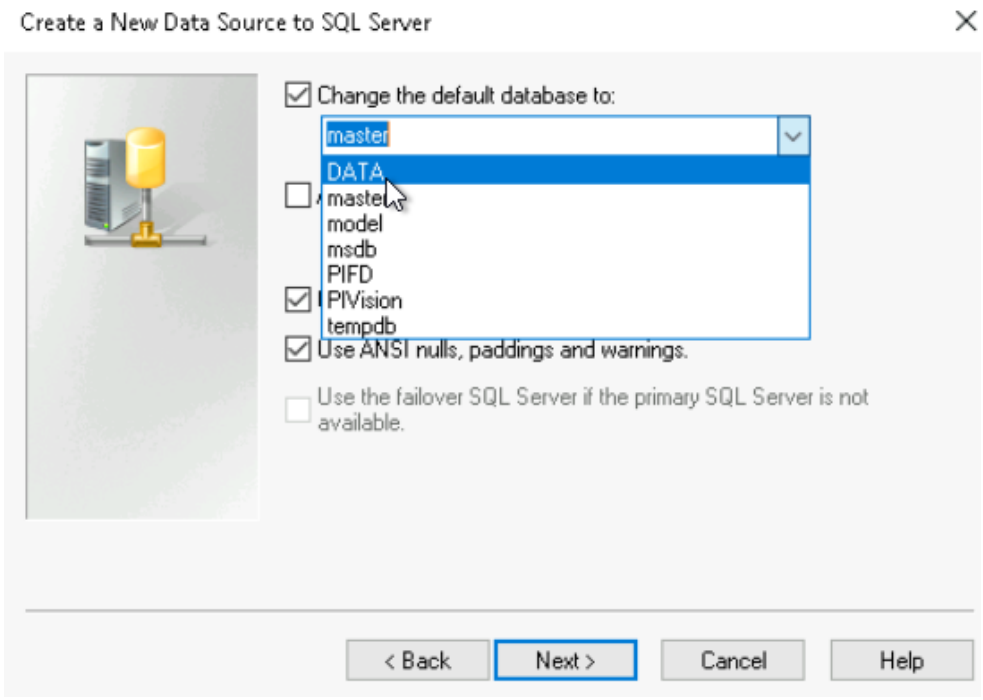
2. Go to System DSN and add a new SQL Server DSN.



3. Use name PIAF01 and enter Server as PIAF01. Click Next. C



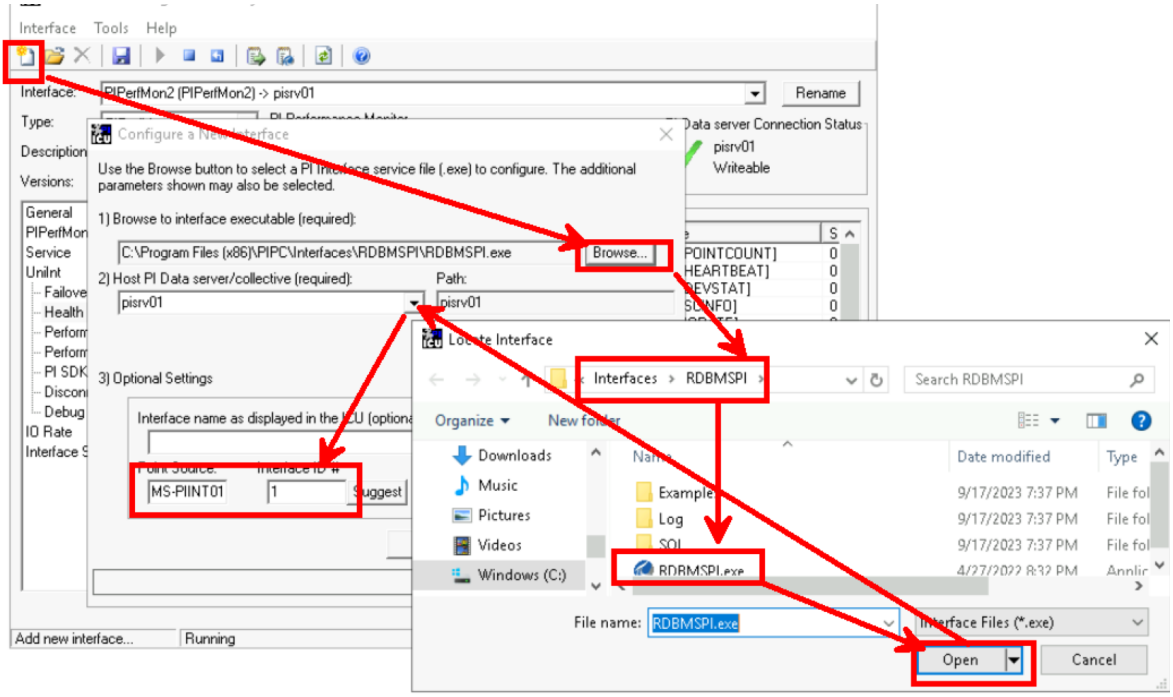
4. Click Next.
5. Change the default database to DATA. Click Next.



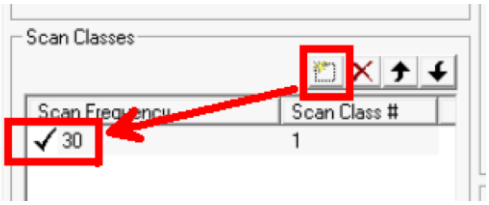
6. Click Finish. Click Test Data Source. If test is successful, click OK and then OK again.

Step 3 : Configure and instance of the RDBMS interface to collect data from Power table.

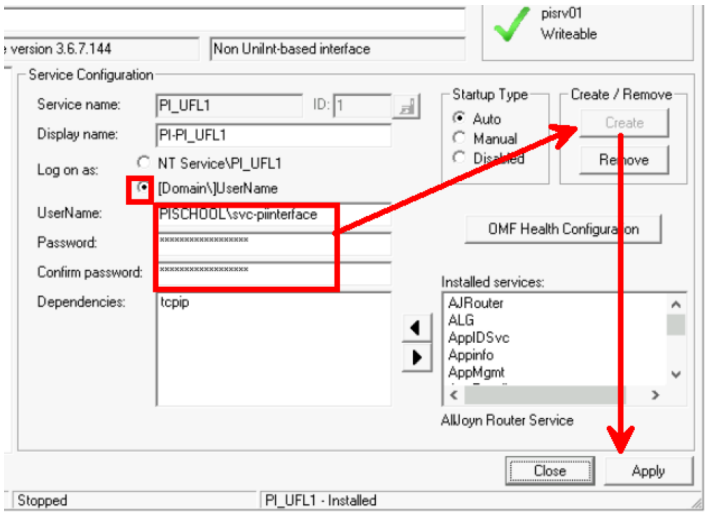
1. Set Point Source as RDBMS-PIINT01 and interface ID as 1.



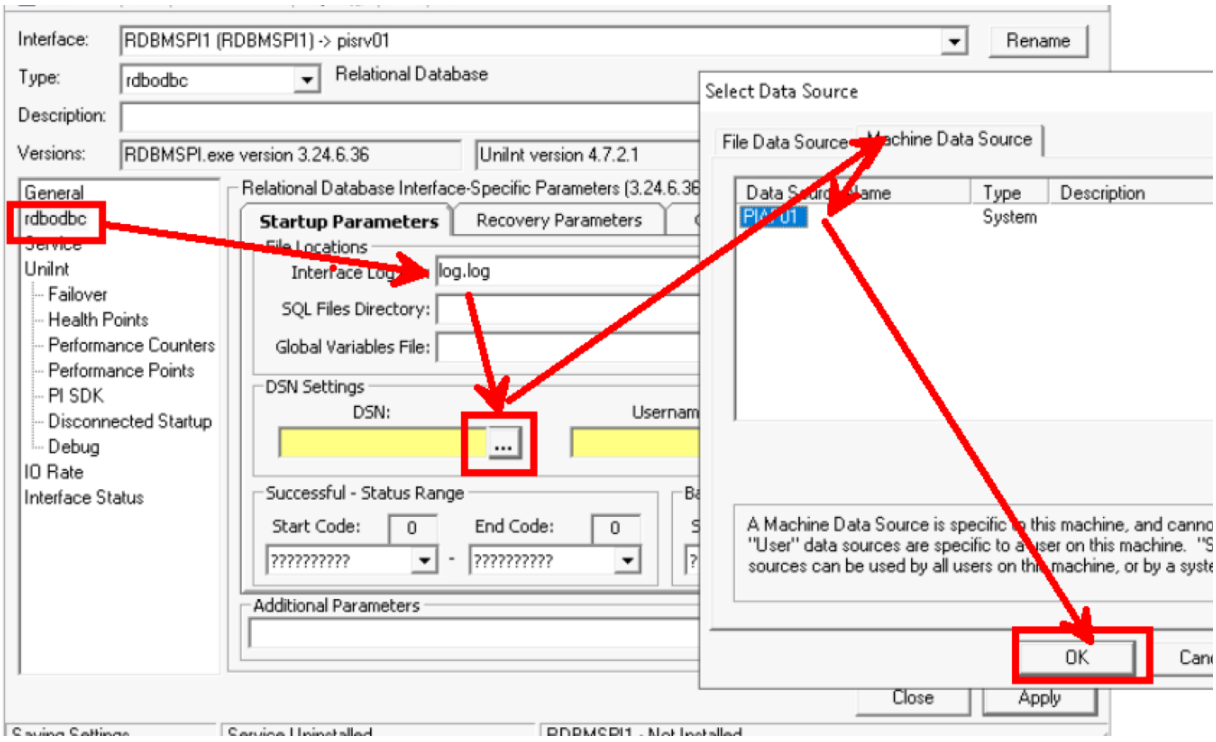
2. Create a scan frequency of 30 seconds.



3. Go to Service tab on the left pane. Click on [Domain\]UserName. Enter PISCHOOL\svc-piinterface and password: student. Click on Create and Apply.



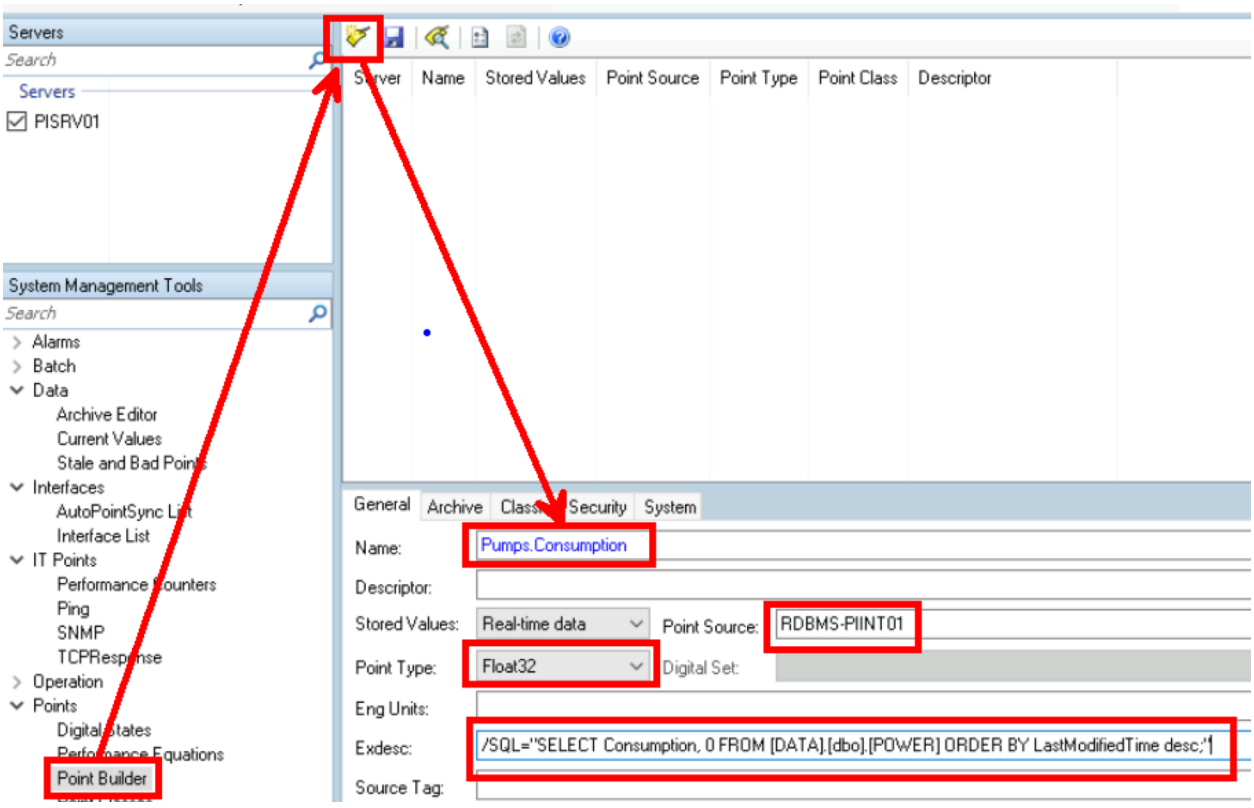
4. Go to rdbodbc tab on the left pane. Enter a file name into interface log file space. Select the DSN we created earlier.



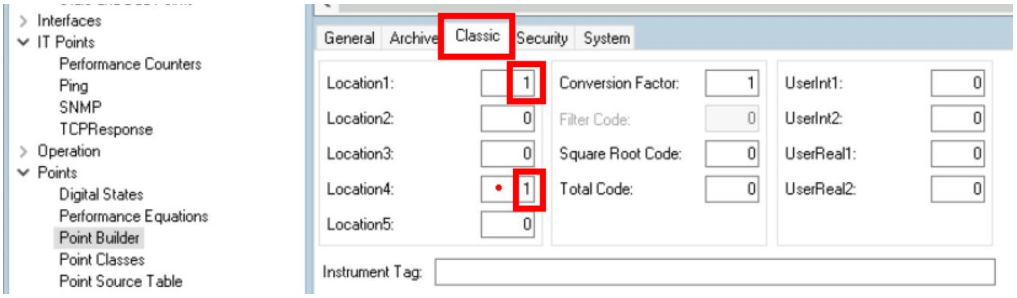
5. Enter PISCHOOL\svc-piinterface and password: student. Click on Apply.
6. Run the interface.

Step 4 : Configure a new tag for PI RDBMS interface using PI System Management Tools. Watch this video to configure the SQL query.

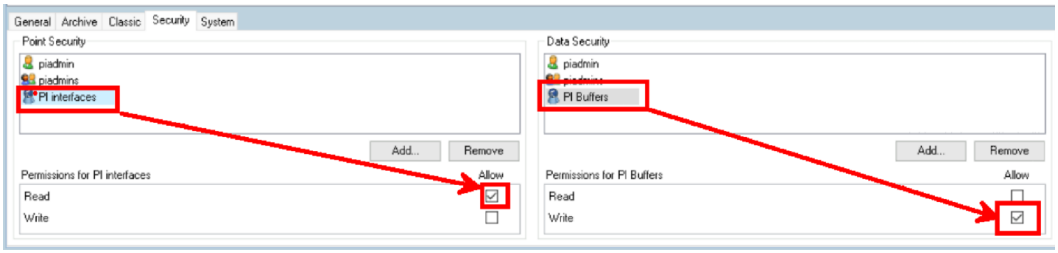
1. Tagname = Pumps.Consumtion
2. Point Source = RDBMS-PIINT01
3. PointType = Float32
4. ExDesc = /SQL="SELECT Consumption, 0 FROM [DATA].[dbo].[POWER] ORDER BY LastModifiedTime desc;"



5. Set Location1 = 1 and Location4 = 1

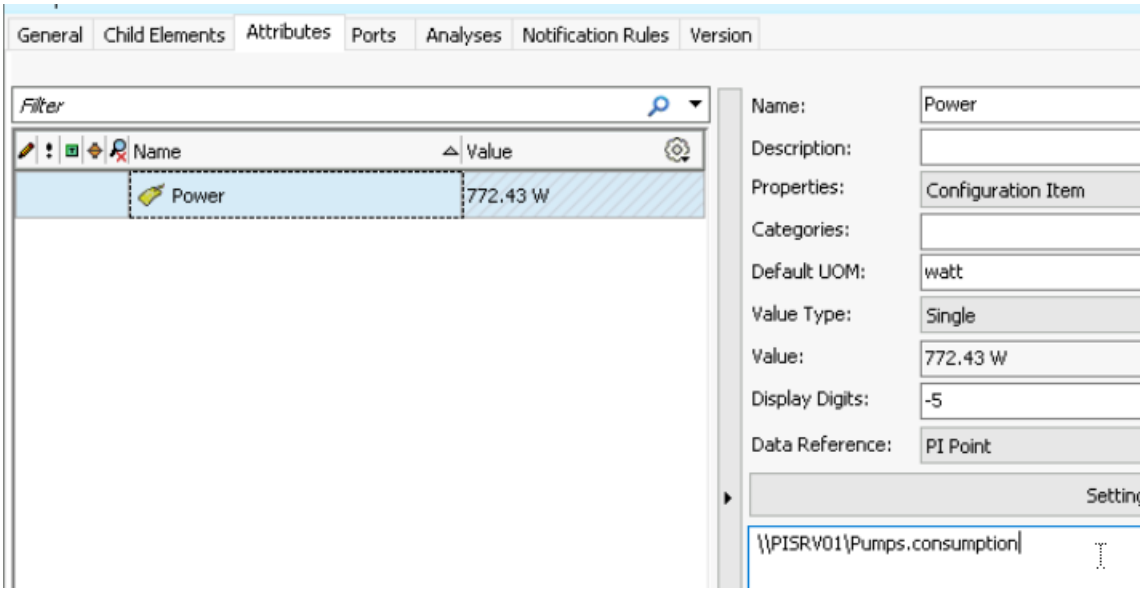


6. Edit Security settings to give Buffering write access to the tag.



7. Check data on PI Data Archive. Do you see historical data? How can you import all the historical data?

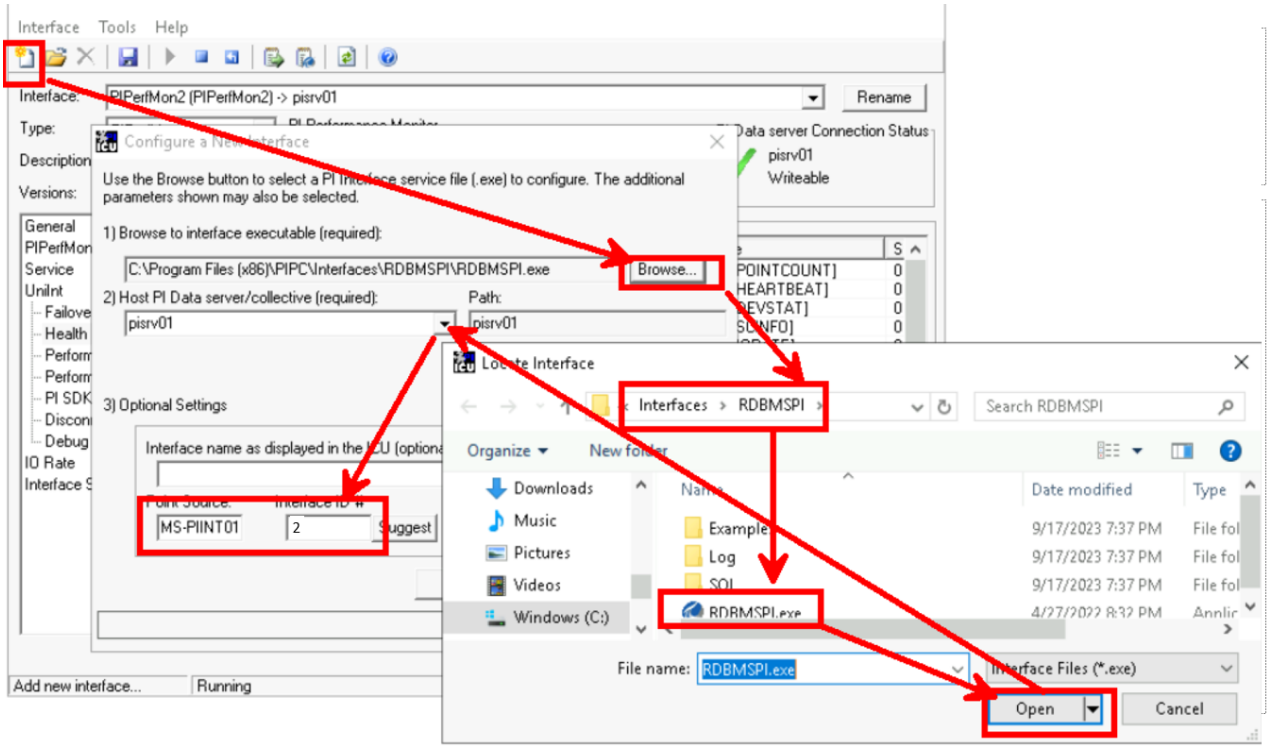
Step 5 : Create a new attribute on Pumps element on AF to see the total Power. UoM is watt.



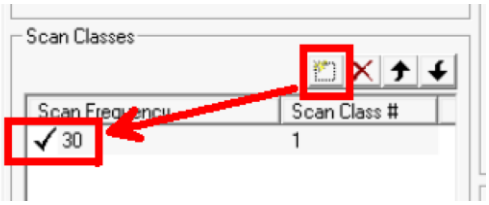
Part 2 – Collect data from **DATA.dbo.PUMPS_EFFICIENCY** table.

Step 1 : Configure a new instance of PI RDBMS interface.

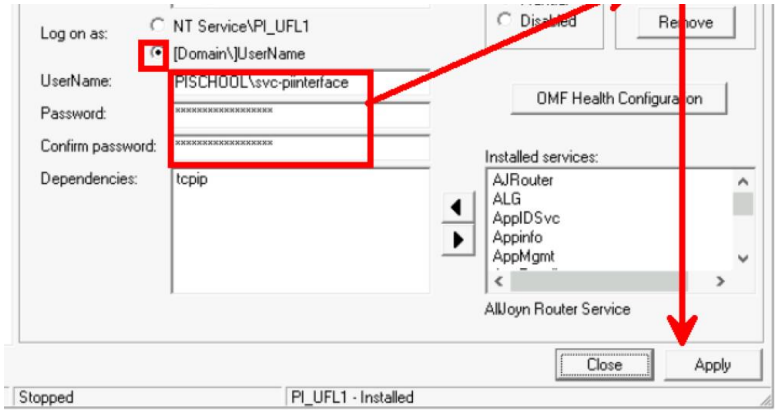
1. Set Point Source as RDBMS-PIINT01 and interface ID as 2.



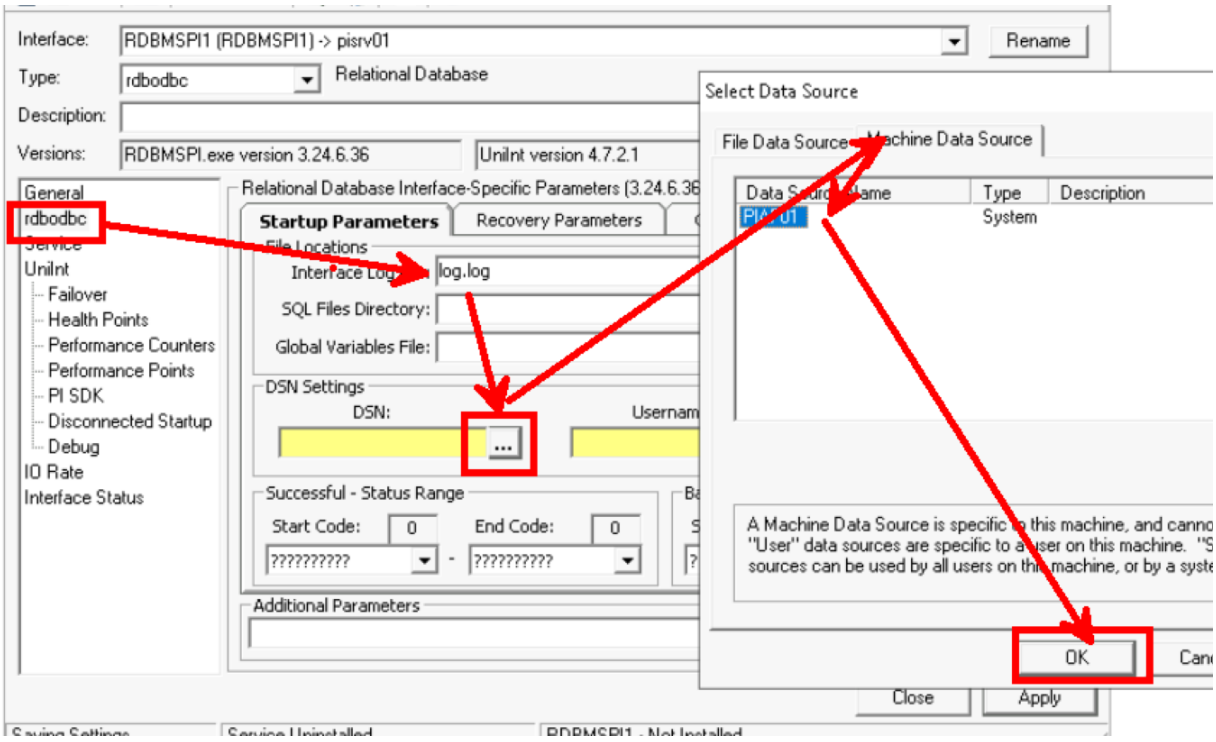
2. Create a scan frequency of 30 seconds.



3. Go to Service tab on the left pane. Click on [Domain\]UserName. Enter PISCHOOL\svc-piinterface and password: student. Click on Create and Apply.

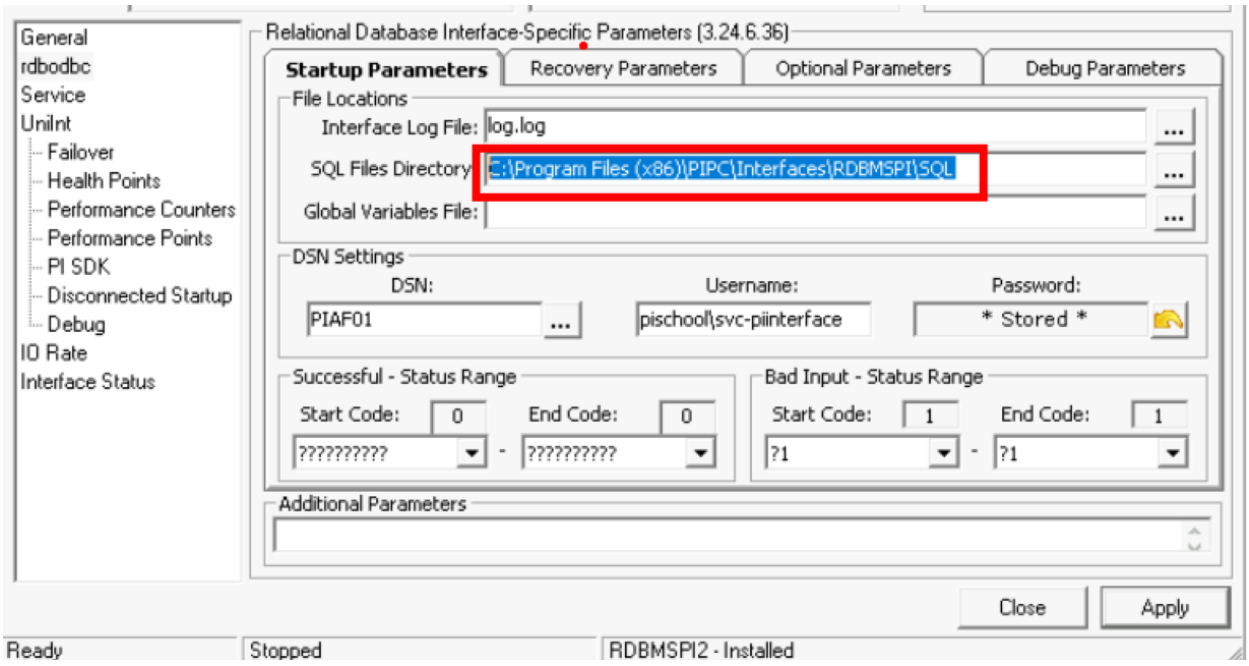


4. Go to rdbodbc tab on the left pane. Enter a file name into interface log file space. Select the DSN we created earlier.



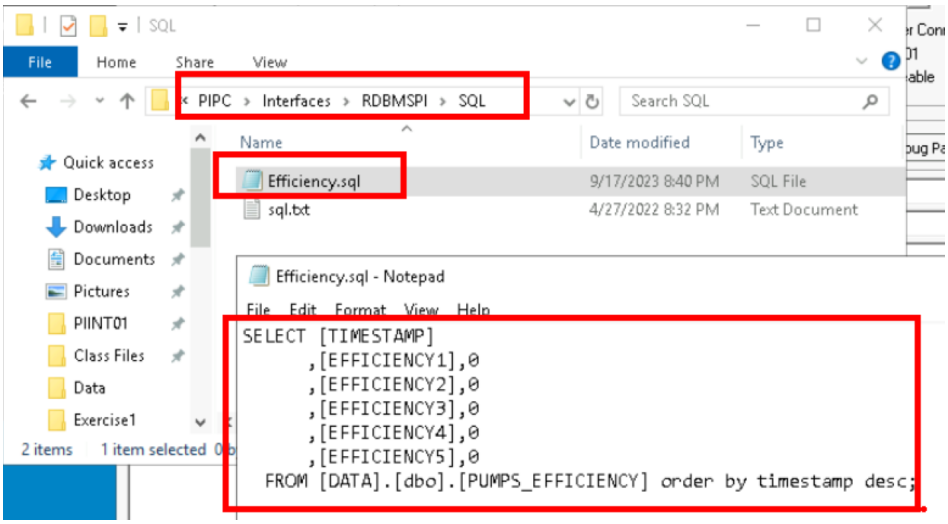
5. Enter PISCHOOL\svc-piinterface and password: student. Click on Apply.
6. For the SQL Files directory, enter

   “C:\Program Files (x86)\PIPC\Interfaces\RDBMSPI\SQL”



7. Go to C:\Program Files (x86)\PIPC\Interfaces\RDBMSPI\SQL and create a new file called Efficiency.sql. Open it using Notepad and enter the query below:

```
SELECT [TIMESTAMP]
      ,[EFFICIENCY1],0
      ,[EFFICIENCY2],0
      ,[EFFICIENCY3],0
      ,[EFFICIENCY4],0
      ,[EFFICIENCY5],0
  FROM [DATA].[dbo].[PUMPS_EFFICIENCY] order by timestamp desc;
```



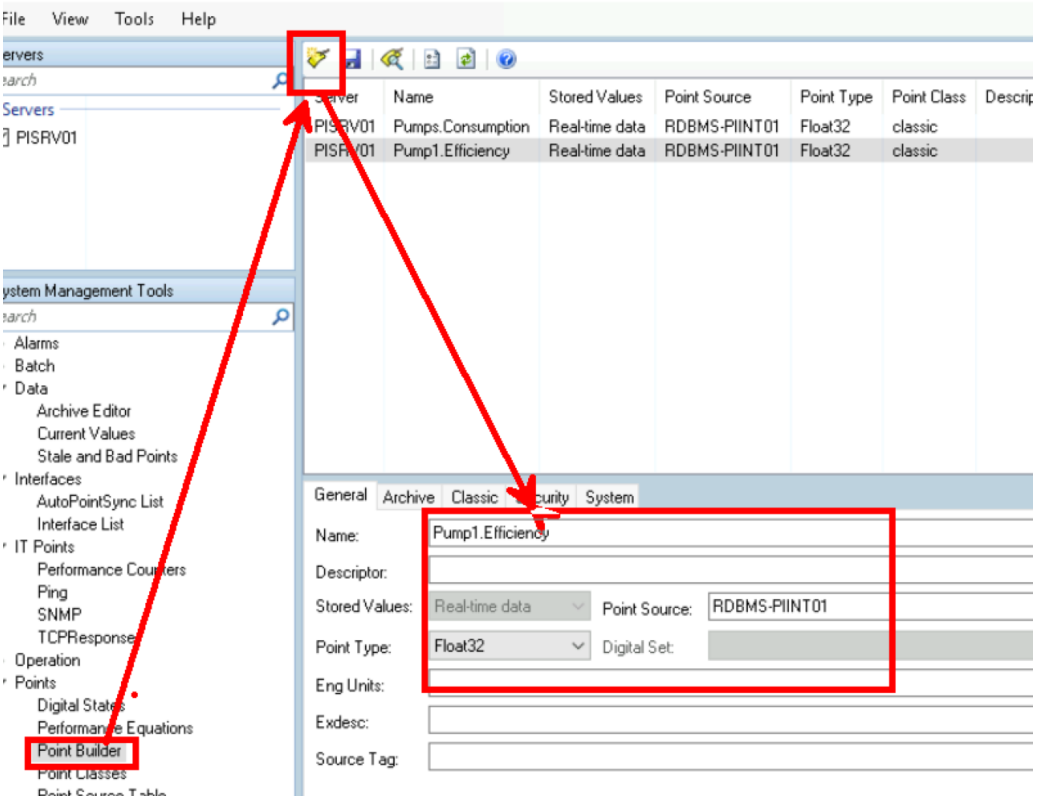
8. Save the file and close it.
9. Run the interface.

Step 2 : Configure new tags for PI RDBMS interface. Watch this video to configure the SQL file.

a. Tagname = PumpX.Efficiency

b. Point Source = RDBMS-PIINT01.

c. PointType = Float32

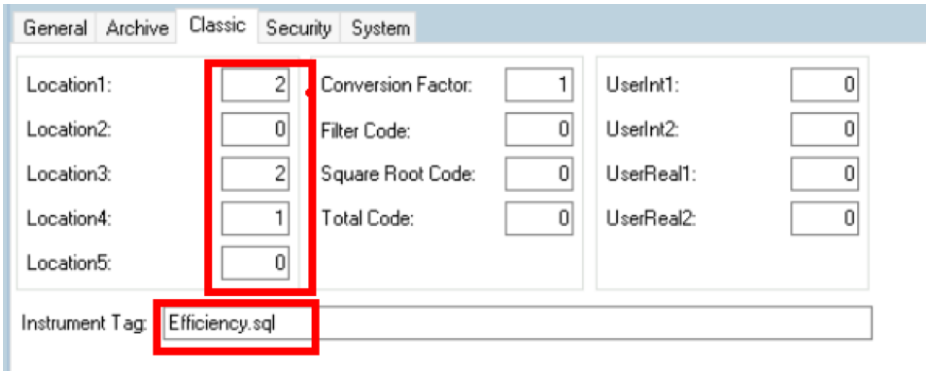


d. Location1 = 2

e. Location3 = [2 for Pump1.Efficiency, 4 for Pump2.Efficiency, 6 for Pump3.Efficiency, 8 for Pump4.Efficiency, 10 for Pump5.Efficiency,]

f. Location4 = 1

g. Instrumenttag = Efficiency.sql



You can use PI Builder to create remaining PI Points.

	Selected(x)	Name	ObjectType	pointsource	pointtype	location1	location4	instrumenttag	location3	datasecurity	ptsecurity
2	x	Pump1.Efficiency	PIPoint	RDBMS-PIINT01	Float32	2	1	Efficiency.sql	2	piadmin: A(r,w) | piadmins: A(r,w) | PI Buffers: A(w)	piadmin: A(r,w) | piadmins: A(r,w) | PI interfaces: A(r)
3	x	Pump2.Efficiency	PIPoint	RDBMS-PIINT01	Float32	2	1	Efficiency.sql	4	piadmin: A(r,w) | piadmins: A(r,w) | PI Buffers: A(w)	piadmin: A(r,w) | piadmins: A(r,w) | PI interfaces: A(r)
4	x	Pump3.Efficiency	PIPoint	RDBMS-PIINT01	Float32	2	1	Efficiency.sql	6	piadmin: A(r,w) | piadmins: A(r,w) | PI Buffers: A(w)	piadmin: A(r,w) | piadmins: A(r,w) | PI interfaces: A(r)
5	x	Pump4.Efficiency	PIPoint	RDBMS-PIINT01	Float32	2	1	Efficiency.sql	8	piadmin: A(r,w) | piadmins: A(r,w) | PI Buffers: A(w)	piadmin: A(r,w) | piadmins: A(r,w) | PI interfaces: A(r)
6	x	Pump5.Efficiency	PIPoint	RDBMS-PIINT01	Float32	2	1	Efficiency.sql	10	piadmin: A(r,w) | piadmins: A(r,w) | PI Buffers: A(w)	piadmin: A(r,w) | piadmins: A(r,w) | PI interfaces: A(r)

Step 4 : Create a new attribute in your Pump element template for Efficiency information. UoM is percent (%).

