Building Asset Hierarchies with PI AF

OSIsoft, LLC
1600 Alvarado Street
San Leandro, CA 94577

Table of Contents

Lesson 1 - Introduction to Asset Framework (AF)

What does AF do for me?

The *PI Data Archive* is focuses on a points database and is extremely good at storing vast amounts of data collected by interfaces. It allows easy and performant retrieval of time-series data. The PI Data Archive architecture is scalable, maintainable and highly available.

The *Asset Framework (AF)* supplements the architecture by providing a Meta-data structure for all data in the organization (“Data Directory”). Asset Framework (AF) has a rich set of features and functions to organize and enhance the data in the PI Data Archive. Because it offers user-friendly access to the data it is the preferred way for users to interact with their PI system data.

What are the benefits of Asset Framework?

Easy way to navigate throughout the system

The hierarchical asset structure gives a convenient way for navigation that can also be used by people, who are not familiar with the technical details of how the data is retrieved from the data source.

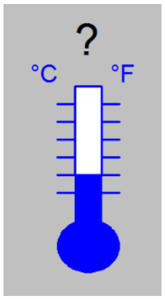
Unify data from disparate source systems

PI Point attributes get their data from PI Data Archives. AF is not limited to one PI Data Archive. Attributes can refer to multiple PI Data Archive Servers (either standalone or collective)

The origin of data can also be a relational database. While some attributes of an element representing a reactor are time-series data coming from PI tags (such as the temperature), data for some other attributes can be from an external database (such as the physical characteristics or inventory data).

All attributes are listed side by side giving the user a comprehensive insight into all the relevant data of the asset.

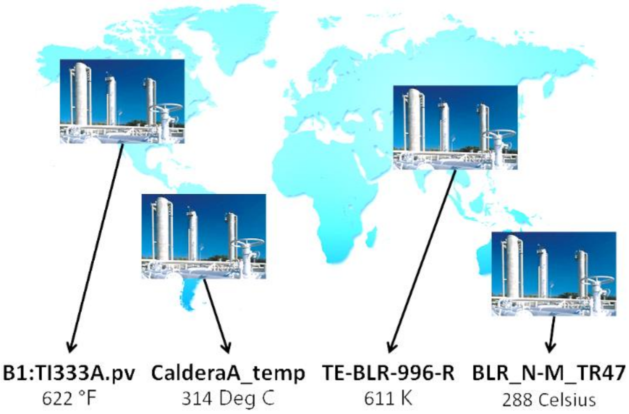
Supports different Units of Measure



PI Asset Framework (PI AF) attributes are associated with specific Units of Measure. AF is preloaded with numerous standard unit-of-measure classes and conversion factors based on the International System of Units (SI). It also supports user defined classes and Units of Measure. For example, the source unit for a temperature sensor attribute in a reactor in the USA can be associated with degrees Fahrenheit, the corresponding attribute for a reactor in Italy can be associated with degrees Celsius. When working with the data (doing calculations or displaying the data) this is properly taken into account.

Standardization with Templates

At many sites there are no rigorous naming standards for the points. There may be missing descriptions and/or engineering units. The PI System is often used to integrate information from different sources and these sources may not have been configured consistently:



Element templates in AF provide the basis for standardization. When applied for elements that represent the same type equipment, all elements have the same set of attributes with a consistent, user friendly naming. The attributes have same unit of measurement, same data type, the same description, etc. This allows a harmonized, consistent representation of your system.

Another benefit of using templates is the quick creation of many elements of the same type.

Powerful Calculation Options (Asset Analytics)

Users can configure, schedule and run calculations written using PI Performance Equation (PE) syntax acting on their PI Asset Framework (PI AF) attributes. PE expressions, Rollup calculations and generation of PI Event Frames based on trigger conditions are all supported analysis types. In addition, analysis templates enable users to manage their analyses in a standardized and consistent manner. Typical applications are Key Performance Indicator (KPI) calculations and condition-based maintenance (CBM).

AF Server Installation and Software Requirements

In a production environment, the AF Server component is typically installed on a dedicated Windows server machine.

For the database management the AF Server is using a **Microsoft SQL Server**, which has to be defined during the AF server installation. The default name of the SQL database used for AF is **PIFD**.

The client program for AF is PI System Explorer (PSE), which is used to configure and maintain the asset structure.

AF Server is compatible with several SQL Server versions (SQL Server 2008 or later; for a complete list refer to AF Server software requirements). The PI AF server works with 32-bit or 64-bit SQL Server versions.

Several SQL Server editions are supported: Express, Standard, Business Intelligence, Enterprise, Datacenter. The Express version is free of charge, but has several limitations. Depending on your AF asset structure size and the AF functions you are using (such as audit trail functionality), the Express version may not be sufficient. Refer to related OSIsoft Hardware System Sizing recommendations.

The SQL Server can reside on the same machine as the AF Server or on another machine. It is possible to use an existing SQL Server that is also used for other applications.

SQL Server Requirements for PI AF High Availability

To take advantage of the PI AF High Availability (HA Collective) feature, the primary collective member must run SQL Server Standard Edition, SQL Server Business Intelligence, SQL Server Enterprise Edition, or SQL Server Datacenter Edition. SQL Server does not need to reside on the same machine as PI AF Server.

SQL Server Requirements for PI AF Audit Trail Functionality

The PI AF Audit Trail feature requires SQL Server Enterprise edition for supported SQL Server versions of SQL Server 2014 and earlier.

The PI AF Audit Trail feature is also supported in:

- SQL Server 2016 (Windows Update or Microsoft KB 3164398 required)
- SQL Server 2016 SP1 Enterprise or Standard Editions
- SQL Server 2017 Enterprise or Standard Editions (Windows Version Only)

For complete information about the system requirements for the AF Server, refer to the AF Server software requirements.

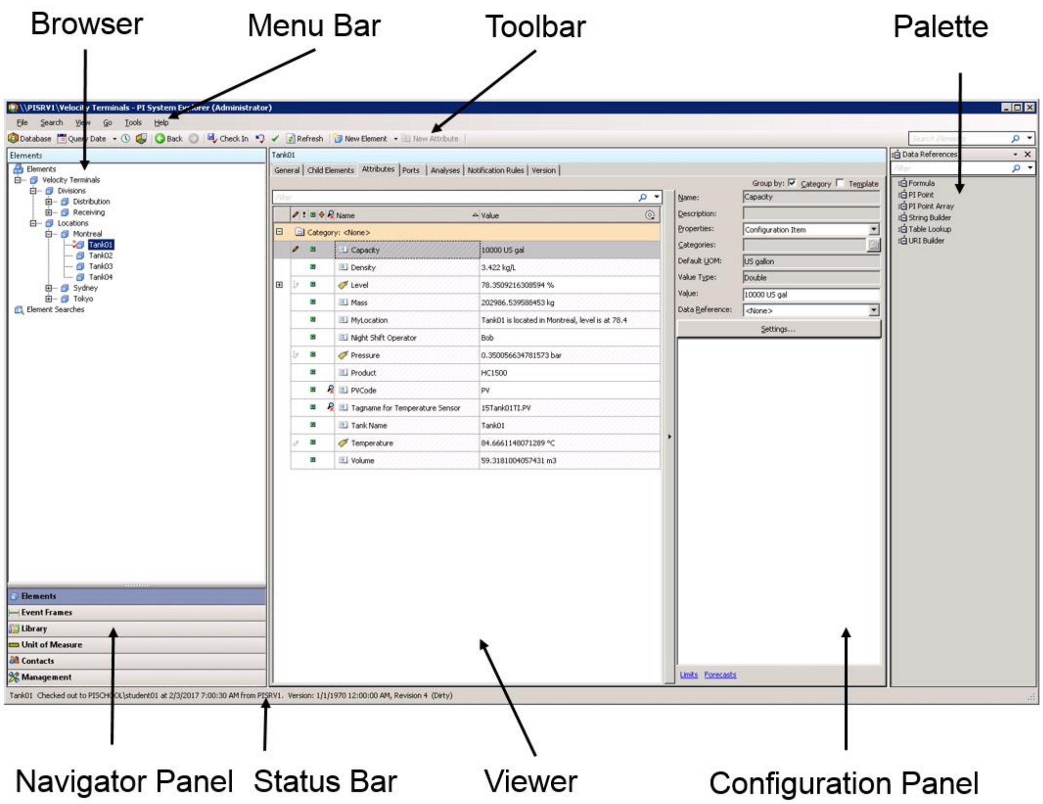
OSIsoft®

Lesson 2 - PI System Explorer Basics

Navigate in PI System Explorer (PSE)

Click on the PI System Explorer icon in the task bar to start the program.

Note: the very first start of PI System Explorer can take some time. In the beginning, no dialog box is displayed. Subsequent starts will be much faster.



Name	Value
Category: <None>	
Capacity	10000 US gal
Density	3.422 kg/L
Level	78.3509216308594 %
Mass	202986.539588453 kg
MyLocation	Tank01 is located in Montreal, level is at 78.4
Night Shift Operator	Bob
Pressure	0.350056634781573 bar
Product	HC1500
PVCode	PV
Tagname for Temperature Sensor	15Tank01TI.PV
Tank Name	Tank01
Temperature	84.6661148071289 °C
Volume	59.3181004057431 m3

Group by: ☑ Category □ Template

Name: Capacity
Description:
Properties: Configuration Item
Categories:
Default UOM: US gallon
Value Type: Double
Value: 10000 US gal
Data Reference: <None>

Settings...

Tank01 Checked out to PISCHOOL\student01 at 2/3/2017 7:00:30 AM from PISRV1. Version: 1/1/1970 12:00:00 AM, Revision 4 (Dirty)

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 Management.

Menu Bar / Toolbar

Use these options for such tasks as opening/creating a database, searching for elements or contacts, applying and checking in changes, setting view options, and so forth. Menus and the Toolbar are context sensitive and will present different options depending on what section is selected from 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 represent either physical or logical entities in your process. They can be organized in several hierarchies. An element can be created from a template or created from scratch. When based on a template, the element derives its initial attribute configurations from the template. Later modifications to the template will propagate to all elements based on said template.

- **Event Frames**: An event frame is any event, defined by a start time, an end time, and a context. Event frames also have attributes containing supporting data. Event Frames can represent downtime events, process and environmental excursions, material transfer events, equipment maintenance events, batch processing steps, safety incidents, and any other events important to your organization. A transfer is a special type of event frame that marks the movement of material in discrete quantities.

- **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. A UOM class is defined by the fundamental dimensions of its measurement. Examples of UOM classes are Mass, Volume, and Density. The UOM database comes preloaded with numerous standard unit-of-measure classes and conversion factors. You can extend these classes by adding new units of measure, as well as new measurement classes. The implementation of UOM is based on the International System of Units (SI).

- **Management**: This section provides a summary of all analyses and notification rules configured on the current AF database. It allows you to perform administrative tasks like starting, stopping and backfilling.

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.

Configuration Panel

The configuration panel is used to configure properties associated with attributes. These properties include categories, attribute references, units, data types, and values for static attributes.

Palette

The Palette displays templates, data references and contacts that can be associated with objects being defining in the Viewer. Often the Palette is hidden to optimize screen real estate.

Status Bar

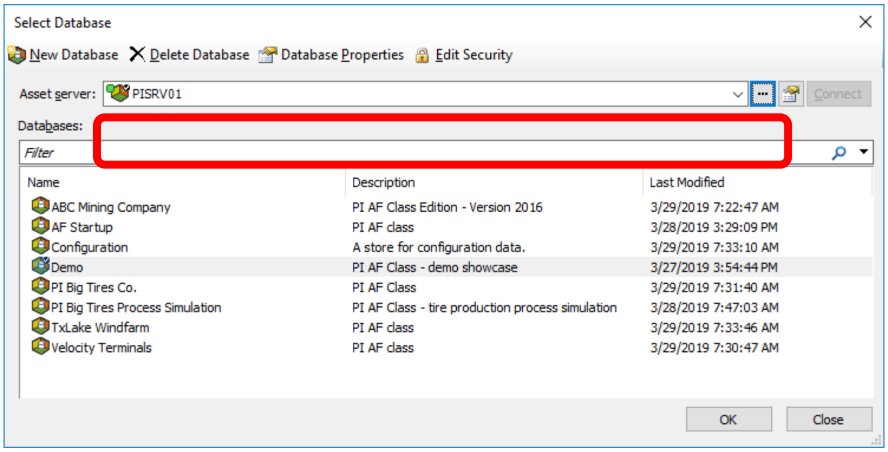
Check the status bar after clicking an item in the Browser to see its status. For example, last modification time, if the object is checked out or if a notification is currently loading.

Connect to and Search a PI System

Connecting to an AF Server

To connect to the AF Server simply open PSE. The default AF Server was set during installation, so you will be automatically connected to it. If a default AF database has already been defined, PSE will automatically connect to it as well. If no default AF database has been defined, a window will pop up and you will be invited to create a new AF database.

It is possible to see which AF server the user is connected to by selecting the Database button on the toolbar. The *Select Database* dialog box will show the current server name in the top drop-down “AF Server” list.



Asset server: PISRV01

Databases:

Filter

Name	Description	Last Modified
ABC Mining Company	PI AF Class Edition - Version 2016	3/29/2019 7:22:47 AM
AF Startup	PI AF class	3/28/2019 3:29:09 PM
Configuration	A store for configuration data.	3/29/2019 7:33:10 AM
Demo	PI AF Class - demo showcase	3/27/2019 3:54:44 PM
PI Big Tires Co.	PI AF Class	3/29/2019 7:31:40 AM
PI Big Tires Process Simulation	PI AF Class - tire production process simulation	3/28/2019 7:47:03 AM
TxLake Windfarm	PI AF class	3/29/2019 7:33:46 AM
Velocity Terminals	PI AF class	3/29/2019 7:30:47 AM

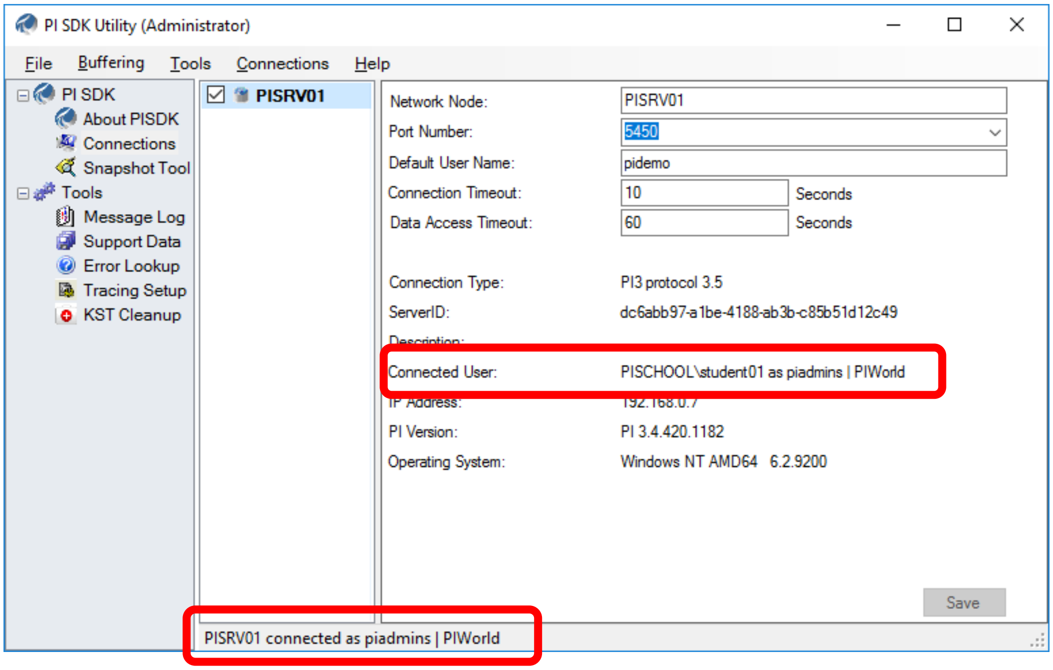
To connect to a different available AF server, simply select it from the drop-down AF Server menu and click on the *Connect* button.

Connecting to a Data Archive

You can validate your connection to the Data Archive via PSE. Notice AF Server icon ( ) is different than Data Archive server icon ( ).

The PI SDK utility provides you with more features to configure and diagnose connections to the Data Archive.

Open the PI SDK Utility (PI System > PISDKUtility (64-bit)). From the *PI-SDK > Connections* section, the Data Archive servers configured on the local machine are displayed. To validate a connection to a Data Archive, simply check the checkbox next to its name or IP address. If connection succeeds, connection information will be shown.



Network Node: PISRV01

Port Number: 5450

Default User Name: pidemo

Connection Timeout: 10 Seconds

Data Access Timeout: 60 Seconds

Connection Type: PI3 protocol 3.5

ServerID: dc6abb97-a1be-4188-ab3b-c85b51d12c49

Description:

Connected User: PISCHOOL\student01 as piadmins | PIWorld

IP Address: 192.168.0.7

PI Version: PI 3.4.420.1182

Operating System: Windows NT AMD64 6.2.9200

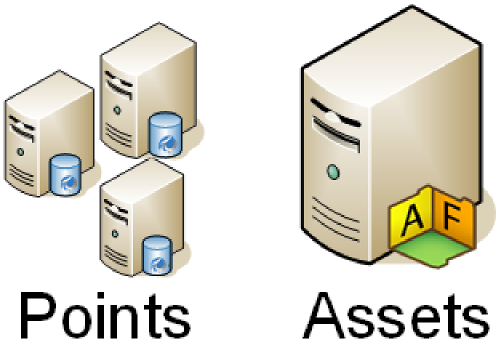
PISRV01 connected as piadmins | PIWorld

The user that was used to authenticate on the Data Archive will be shown in the status bar and the connection information section on the right.

Note: The PI SDK Utility is a particularly useful tool to troubleshoot permission issues.

PI Points in your Data Archive have a set of properties that define them (these properties are called attributes of the tag). Below are some of the parameters you can use to filter searches for PI Points in PI System Explorer.

Searching for PI Points in PSE: Basic PI Point Properties



PI Points in your Data Archive have a set of properties that define them (these properties are called attributes of the tag). Below are some of the parameters you can use to filter searches for PI Points in PI System Explorer.

Point Name

This is the unique name used to create points for storage in the Data Archive. Examples: **P294H.FI.PV,** M03_E1P1_MOTDRV1202_RUNSTAT

Descriptor

This is the human-friendly description of the Data Archive point. The descriptor is often a search criterion since the point name is not always intuitive. Often the point name is some sort of abbreviated convention and the descriptor captures the “full name.”

Point Source

Points can be related to their interfaces that collect the data by a point attribute called *pointsource*. Grouping by point source allows all of points associated with a particular device to be identified by searching for all points of a certain point source. This assumes that the user knows the point sources in use and that will not be true in some situations.

Point Type

This is the attribute that specifies the data type for the values that a point stores. The possible point types are:

- int16, int32, float16, float32, float64, digital, string, BLOB, timestamp

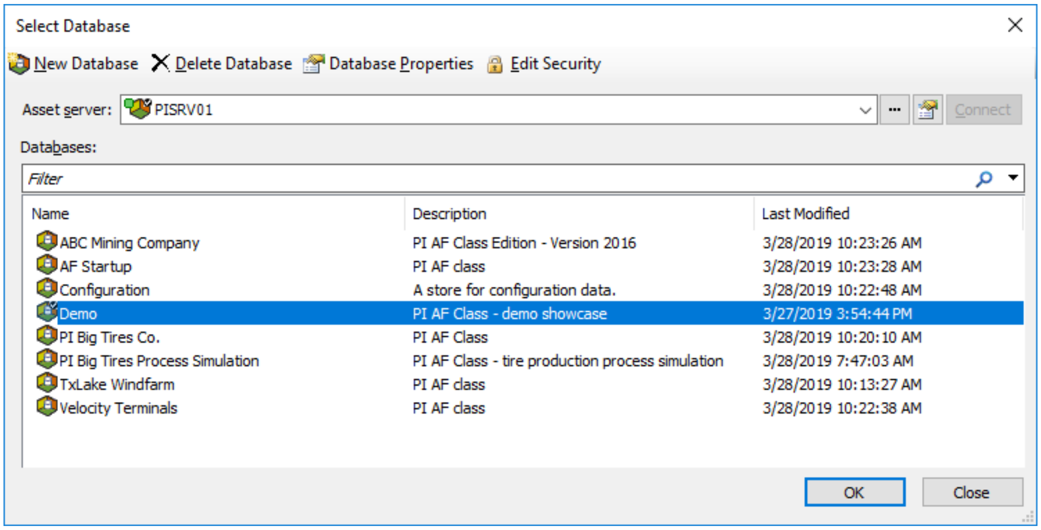
Connect to, Create and Share PI AF Databases

Open PSE from the start menu and, if necessary, connect to the AF server PISRV01.

If PSE opens normally, choose ***File > Database***. If PSE does not open and a pop-up appears, click on the ***Select*** option to get to the Select Database window.

To create a new database, right-click anywhere in the list of databases and choose ***New Database*** or click on the New Database button on the toolbar.

To select the existing database for entering the Velocity Terminals structure, select ***AF Startup*** and click the OK button or double-click on it to open the database.



Asset server: PISRV01

Databases:

Name	Description	Last Modified
ABC Mining Company	PI AF Class Edition - Version 2016	3/28/2019 10:23:26 AM
AF Startup	PI AF class	3/28/2019 10:23:28 AM
Configuration	A store for configuration data.	3/28/2019 10:22:48 AM
Demo	PI AF Class - demo showcase	3/27/2019 3:54:44 PM
PI Big Tires Co.	PI AF Class	3/28/2019 10:20:10 AM
PI Big Tires Process Simulation	PI AF Class - tire production process simulation	3/28/2019 7:47:03 AM
TxLake Windfarm	PI AF class	3/28/2019 10:13:27 AM
Velocity Terminals	PI AF class	3/28/2019 10:22:38 AM

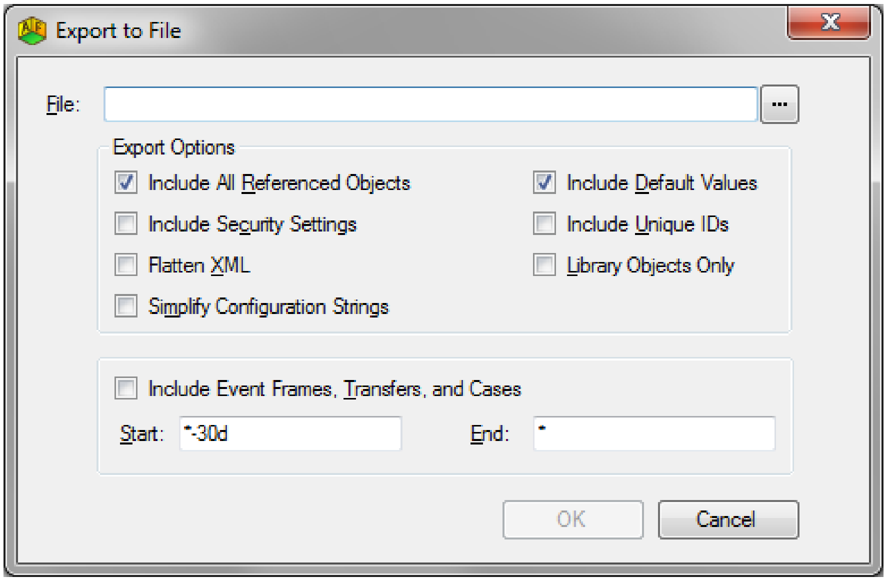
The Configuration Database

As an administrator, it is possible to view a database called Configuration. **Do not use this database to organize your assets** as it is only shown to administrators and used to store configuration data for PI System software.

Using the Import and Export Feature in PSE

From PSE, it is possible to use an Import and Export feature to export an entire AF database or just a AF object to an XML file. With the Import functionality, it is possible to import this AF object into another AF database. This can also be used to back up a AF structure before performing potential destructive actions.

It is possible to export almost any AF object by simply right clicking that object and selecting the Export to File... option. To export the whole AF hierarchy along with all the necessary information to rebuild it completely, simply use the *File > Export to File* option.



Export to File

File:

Export Options

☑ Include All Referenced Objects

☑ Include Default Values

☐ Include Security Settings

☐ Include Unique IDs

☐ Flatten XML

☐ Library Objects Only

☐ Simplify Configuration Strings

☐ Include Event Frames, Transfers, and Cases

Start: *-30d

End: *

In order to be able to rebuild the structure as is, it is required to check the ***Include All Referenced Objects***, so that Templates, Tables, UOMs, etc. are also exported to the .xml file.

Lesson 3 – Introduction to AF Modeling

Modeling Approaches in PI AF: Top-down Versus Bottom-up

Starting to Build an AF Structure

In a AF database, assets can be organized or structured into hierarchies. According to its definition a hierarchy is “a system of persons or things arranged in a graded order” (*thefreedictionary.com*). Logical hierarchies of assets or people are common in companies from any industry. They are used to convey the place/location of each entity in the company. Since the AF system is probably not the only enterprise system in your company, other systems in place might already make use of asset hierarchies (maintenance systems, etc.).

Those systems will often have an option to export the hierarchy information into a flat file (.csv, .txt, .xml, etc.). The content of this file can then be opened in Microsoft Excel and be used as a starting point to build the hierarchy in AF using the PI Builder add-in.

The whole company does not need to be modeled in AF for you to be able to get value out of it. In fact, the AF structure build-up is often project-driven. This means to begin with, a small structure containing your pumps might be built for use in PI ProcessBook displays. As more and more applications start making use of AF, the structure will continuously grow to include more and more of your assets.

That being said, it is still a good idea to define the various hierarchy levels that will be represented in the AF structure (divisions, locations, production lines, process, etc.) before gradually adding the assets to that hierarchy.

Styles of Approach for Designing Assets

There are several styles for designing asset models in AF. Even if the following approaches will result in a complete asset model, there will always be more information to add in the future. That is fine and is in fact a relief, since it means the AF model does not need to be perfect and complete the first time.

Bottom-up

Often a Data Archive is already in place so one approach is to group the assets by similar Data Archive points. These “similar object” groups become AF element templates, and the “objects” become AF elements. Any foreign data that is available is imported. Next, analytics that may seem useful are added, and finally consumers such as PI ProcessBook displays or PI Datalink reports.

The ***bottom-up*** approach has the advantage of being a clear, reality-based path to follow. The Data Archive points are known and analytics can be added on top of them.

However, a clear disadvantage is that this approach lacks vision. This approach does not guarantee the result to be useful since the asset model was not built with a particular goal in mind.

Top-down

Another approach is to start by asking theoretical questions:

- ✓ What is the goal of the asset model?
- ✓ What do the consumers need from AF?
- ✓ What are the business requirements?
- ✓ What analytics will be desirable?
- ✓ What sort of foreign data might be useful?

After answering these questions, the element templates and elements can be sketched, and organized in a hierarchy. Next, AF attributes can be added for the desired data points, even if the source for the data is not known yet. The first stage is simply to add the placeholders for that data, i.e. non-configured attributes. If these attributes are confirmed to be the ones needed, then the analytics can be added, the Data Archive tags can be mapped, and the model can be fully implemented.

This ***top-down*** approach has the advantage of planning for the usefulness of the AF model. Confidence that the model will be well designed and reusable is assured. This advantage cannot be understated and usually outweighs all of the following disadvantages. One disadvantage is that the design can drift away from reality and be incredibly hard to implement while having little benefit over an easier alternative, which is hard to tell beforehand. Another inherent disadvantage is that much of the available raw data that is not “necessary” will be ignored from the model.

Design top-down, then work bottom-up

The preferred approach will be a compromise: begin top-down, identifying goals and trying to identify a “good” design for everything, then quickly move to a mix of this theory and bottom-up experimentation. If a piece of data looks useful, add it to the model because it is rarely a mistake to do so.

This planned approach combines the advantages of top-down and bottom-up: assurance that the model will be useful, yet a good grounding in reality and completeness during construction. The steps of this approach are:

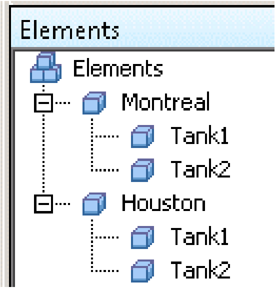
- Define the assets.
- Design the element templates and inheritance tree.
- Add attribute templates.
- Configure the attributes to point to foreign data or Data Archive points, all with proper units of measure.

- Add calculations/analytics.
- Test the model using consumers (sample reports, displays, etc.).

Organizing AF Elements in Hierarchies

Avoid same names for different devices

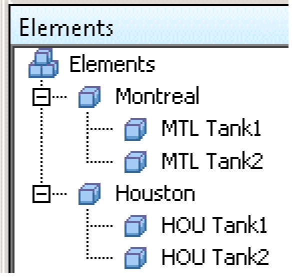
If you have multiple devices in different environments, do not assign the same name to them, as this can cause confusion. The following structure is possible but not recommended:



```
Elements
Elements
⊟ Montreal
    Tank1
    Tank2
⊟ Houston
    Tank1
    Tank2
```

not recommended

To avoid this situation, either assign unique names (such as Tank1, Tank2 in Montreal and Tank3 and Tank4 in Houston) or make the names unique by adding a location code:



```
Elements
Elements
⊟ Montreal
    MTL Tank1
    MTL Tank2
⊟ Houston
    HOU Tank1
    HOU Tank2
```

Different views for the Assets

Once a hierarchical structure has been built in AF and the assets have been defined, it is possible to organize the assets underneath that structure. Depending on the type of hierarchy that was created, the equipment will be organized by geographical locations, by enterprise divisions, by type of equipment, etc. However, having one kind of asset organization does not imply another type cannot also be used in the same AF database.

AF has this ability to let the system manager organize their assets in multiple different ways. It is then possible to have different “views” of the same information but without duplicating that information. This can be done using Element References (⧉).

Create Elements and Attributes in AF

About AF Elements

The element is the basic building block of AF. It is an organizational feature that can be thought of like a folder. It has relationships with other elements and has searchable characteristics. An elements has no Data Reference and does not have a value. Elements are generally organized hierarchically, though this is not required.

Elements represent physical or logical entities in your process, this can be: equipment, lines, products, systems, organizations or sites. They can be based on a template or created without one, although later sections will emphasize the fact that templates are highly recommended.

Elements in a AF database must be uniquely named to their path. An element named “Transformer” can exist under an element named “System A” while another element named “Transformer” can exist under “System B”. However, two elements named “Transformer” can not coexist under the same element “System A”. It is recommended that unique names are considered for all elements regardless of their position in the hierarchy.

The name of an element can contain any character except control characters or any of the following (; ? ` ‘ “ \ | {} []).

To **rename** an element, simply right-click on it in *PI System Explorer > Elements* and select *Rename*. Or use the *NewName* column in the PI Builder in Excel.

Because AF keeps relationships using the elements’ GUID, elements can be renamed without breaking relationships between elements or elements and their attributes or their templates. GUIDs are also used in the relationships of elements in data references as well.

About AF Attributes

As with elements, attributes must also be uniquely named along their path within the element they belong to. Attributes can have a configured constant value or can get a value dynamically from a Data Reference.

Value Type

The **Value Type** field for the attributes defines the format the value itself will have. The various types will be familiar to those with programming experience. In general, value type Double can be used for most analog/numeric values and value type String for everything else. Other types may also be useful in certain circumstances. For example, a value type of Boolean allows either a 0 or a 1, which is represented as False (0) or True (1).

Display Digits

The **Display Digits** field (new in AF 2018) can be used to control the number of digits you want to see for the attribute: zero or positive numbers indicate the number of digits to display to the right of the decimal point. A negative number indicates the number of significant digits to display. In this case, the absolute value of Display Digits is the number of significant digits. **Note**: If your AF Server version is **before 2018**, the functionality is identical to *Display Digits = -5 (default)* in the table below.

Example (Value= 23.45)

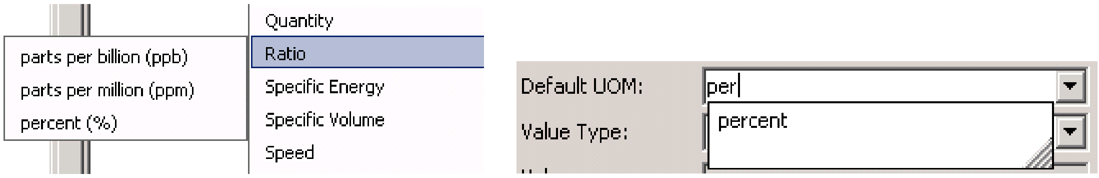
Display Digits	**Format**
3	23.450
2	23.45
1	23.5
0	23
-1	2E+001
-2	23
-4	23.45
-5 (default)	23.45

Units of Measure for Attributes

Attributes will generally be assigned units of measures (UOM). These UOM are organized into comparable classes. The Default UOM configuration field defines the default unit of measure to use to display the attribute’s values. This default unit of measure does not need to be the same as the instrumentation. Later sections will show how this can be used.

Note: **Always assign a unit of measurement!** Do not leave this undefined. This will avoid misinterpretation of the measurement in all subsequent operations.

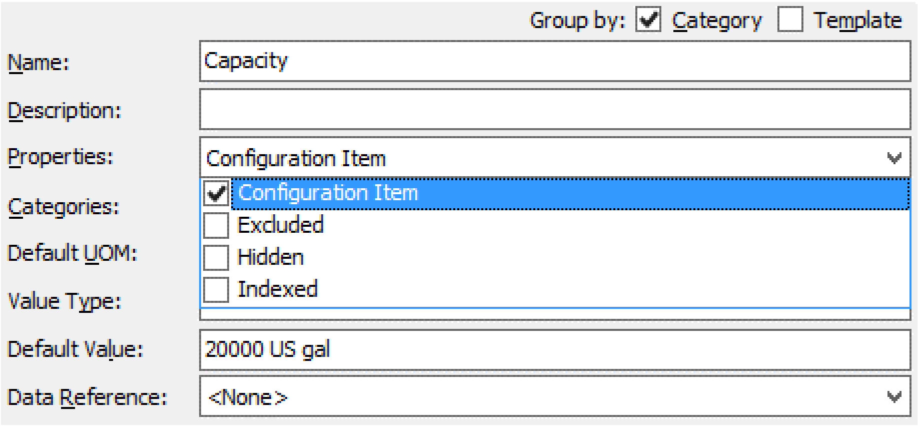
- To enter an UOM, select the UOM from the corresponding UOM Class (here: UOM class= Ratio, UOM= percent). Alternatively, enter the first characters (such as *per*), then select from the list of matching UOMs.



Default UOM: per
percent
Value Type:

Properties

When attributes are created, there are four associated properties (configuration item, excluded, hidden, indexed). You have the option to set those properties for each element attribute. In case of an element derived from a template, the properties of the attributes cannot be changed any more except for the Exclude property:



Group by: ☑ Category □ Template

Name: Capacity
Description:
Properties: Configuration Item
☑ Configuration Item
□ Excluded
□ Hidden
□ Indexed
Categories:
Default UOM:
Value Type:
Default Value: 20000 US gal
Data Reference: <None>

Configuration Item	You assign the **Configuration Item** property to an attribute with a constant value that represents inherent properties of an asset (e.g. a device serial number). In PI System Explorer, configuration attributes are marked with a pencil icon (✎). When you change the attribute value of a configuration item PI System Explorer automatically checks out the attribute. To commit the change you need to Check In.
Indexed	**Indexed** attributes are attributes that are optimized for fast search results and fast value retrieval. You can only index attributes whose values are stored in the AF database. This means that you cannot index attributes that get their values from PI point data references or from linked table references.

Excluded (new in AF Server 2015)	In situations where not all attributes in an element template apply, attributes that are not applicable can be excluded. **Example**: only some of the tanks have a second container coating. Set the attribute for the material of the second coating to **Excluded** property for the tanks with a single coating.
Hidden (new in AF Server 2015)	The **Hidden** property is useful if an attribute is being used to hold an intermediate result, such as a table lookup result that can then be retrieved by a PI point data reference, or is being used solely to populate a tag name in a substitution parameter.

Attribute Traits

Attribute traits hold characteristic information for their parent attribute. These can be limits, forecast values, geolocation information, reason codes, asset health, and analysis start triggers. They can either be defined as a fixed value or in combination with a Data Reference such as e.g. PI Point or Table Lookup, (Data References will be discussed in Lesson 4). **Attribute traits are child attributes**, which can be created by right-clicking on an attribute and selecting *New Child Attribute.* Define the Attribute Trait by using the *Properties* drop-down when the child attribute is selected.

Limit

Limits typically represent the expected range of a process variable. Following limit traits are defined:

- Minimum and Maximum (very lowest/highest possible value)
- LoLo and HiHi (very low/high value for an abnormal situation, typically initiates an alarm)
- Lo and Hi (low/high value that needs attention, typically initiates a warning)
- Target (aimed-for value such as setpoint)

Forecast

Forecasts contain predicted values, which allow to compare actual value with the parent attribute. It typically comes from a future PI point.

Location

Use location attributes to define longitude, latitude, and altitude information for an asset. You can use this information to identify the location of the asset on a map. Used by Integrator for ArcGIS.

Reason

Use reason attribute traits on event frames and transfers to enable users to select a reason code for excursions, downtime, and other events. The reason attribute trait must be an enumeration set that is previously defined, or a system enumeration set delivered with PI AF.

Analysis start-trigger

When users configure analytics to generate event frames, they can optionally elect to store the name of the start trigger in the value of an attribute (string) and mark that attribute with the analysis start trigger trait. This enables clients like PI Vision to indicate the start trigger that created that particular event frame.

Health

Use health attribute traits on elements and models to enable users to set a numeric health score and a health status (for example, healthy, out of service, in maintenance, warning, or error). The HealthStatus attribute trait uses values from the Health Status enumeration set, which is delivered with PI AF. Administrators can modify the Health Status enumeration set as required.

About Saving Modifications

The AF Server uses a sandbox concept for modifications. It holds the changes until they are published to the end users. When ✓Check In is pressed, the changes made in your active session are published as the single public version that others can now see.

Until you check in, other users will not be able to edit a checked out object. The checked out icon will be shown (). As for the user currently editing an object in AF, the object icon will signify some changes were made to the object, that are not yet published (✓). The “Undo all changes to the Database” button (↶) next to the check in button will reset your sandbox changes and set your sandbox back to the database state as it was before you started working

Lesson 4 – Attribute Data References

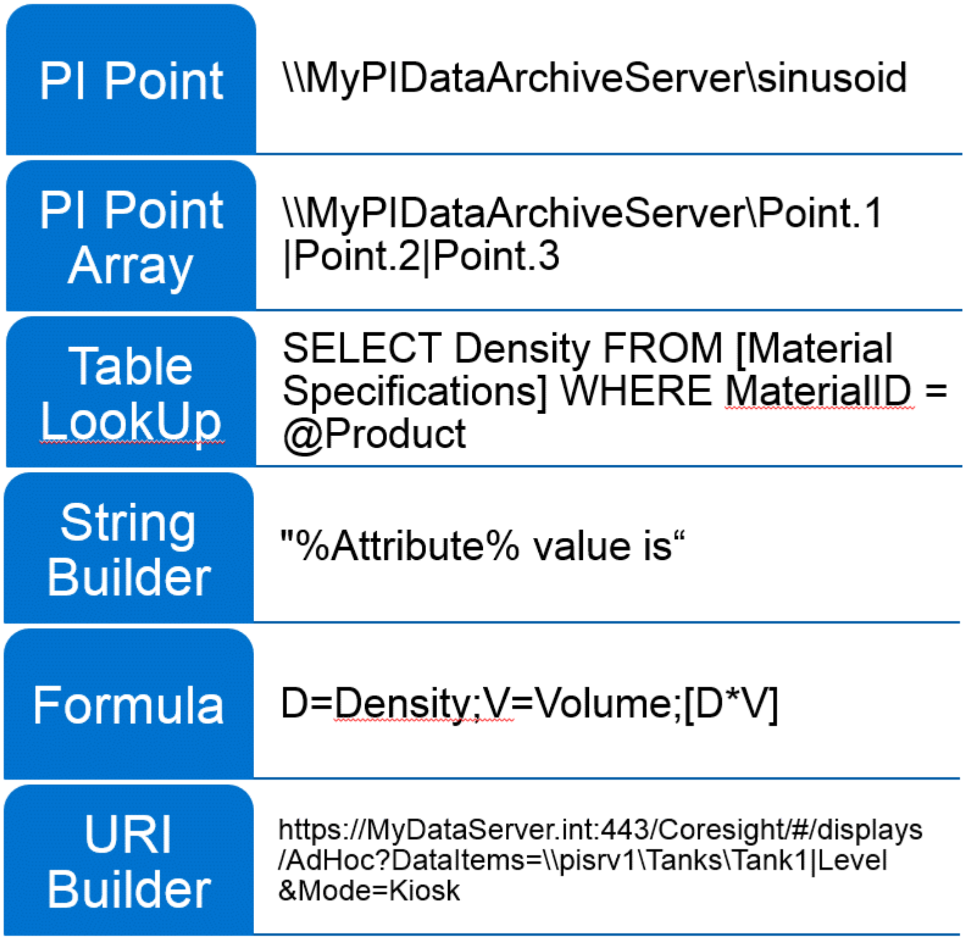
Attribute Data Reference Types

Attributes represent a unique property associated with an asset. The *Data Reference Type* of an attribute defines from where to get the data for the attribute. The following Data Reference Types exist:

- PI Point
- PI Point Array
- Formula
- Table Lookup
- String Builder
- URI Builder

Note: <none> means there is a static value for this attribute

Data Reference Type Examples



PI Point	\\MyPIDataArchiveServer\sinusoid
PI Point Array	\\MyPIDataArchiveServer\Point.1 |Point.2|Point.3
Table LookUp	SELECT Density FROM [Material Specifications] WHERE MaterialID = @Product
String Builder	"%Attribute% value is“
Formula	D=Density;V=Volume;[D*V]
URI Builder	https://MyDataServer.int:443/Coresight/#/displays /AdHoc?DataItems=\\pisrv1\Tanks\Tank1|Level &Mode=Kiosk

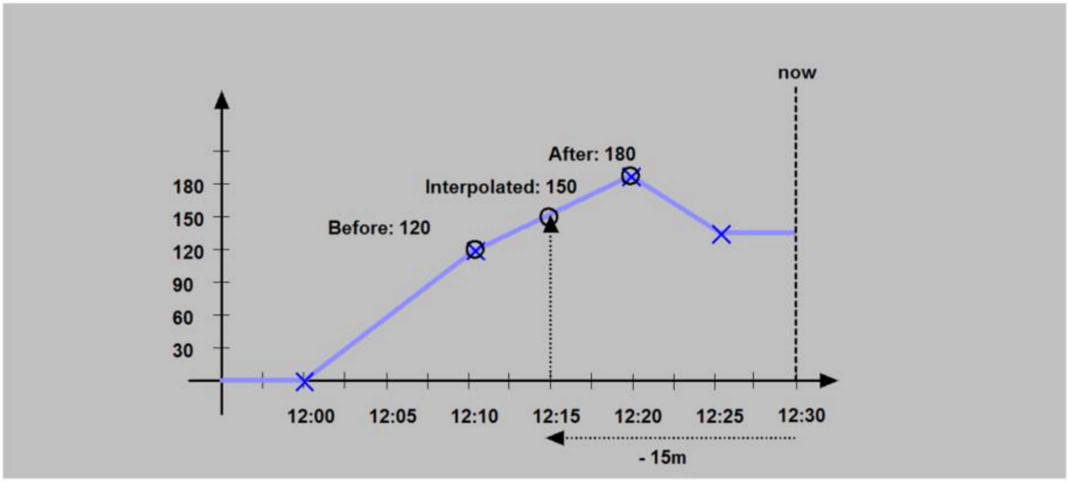
Use Alternative Value Retrieval Methods with PI Point Data Reference

Value Retrieval Modes

By default, the PI Point Data Reference retrieves the current value of a particular point. The value retrieval methods of the PI Point Data Reference can be configured so that the value will be either

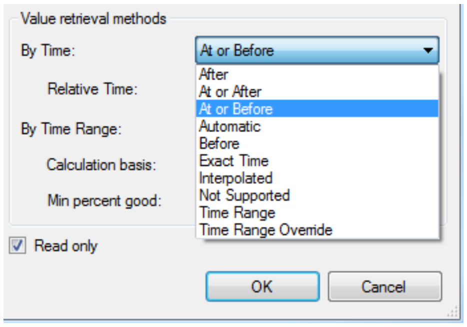
- the value of a point at a specific time (**By Time** retrieval mode)
- the result of a calculation on the point’s value over a time range, e.g. an average
(**By Time Range** retrieval mode).

By Time – Retrieving a value other than the current one



After | At or After | At or Before | Automatic | Before | Exact Time | Interpolated

By Time mode options are **After**, **At or Before**, **At or After**, **Automatic**, **Before**, **Exact Time** and **Interpolated**. To get a value other than the current value these options are used in context with the **Relative Time** field. Relative time expressions have to be in the PI System Time format (see examples below).



Value retrieval methods

By Time: At or Before

Relative Time:

By Time Range:

Calculation basis:

Min percent good:

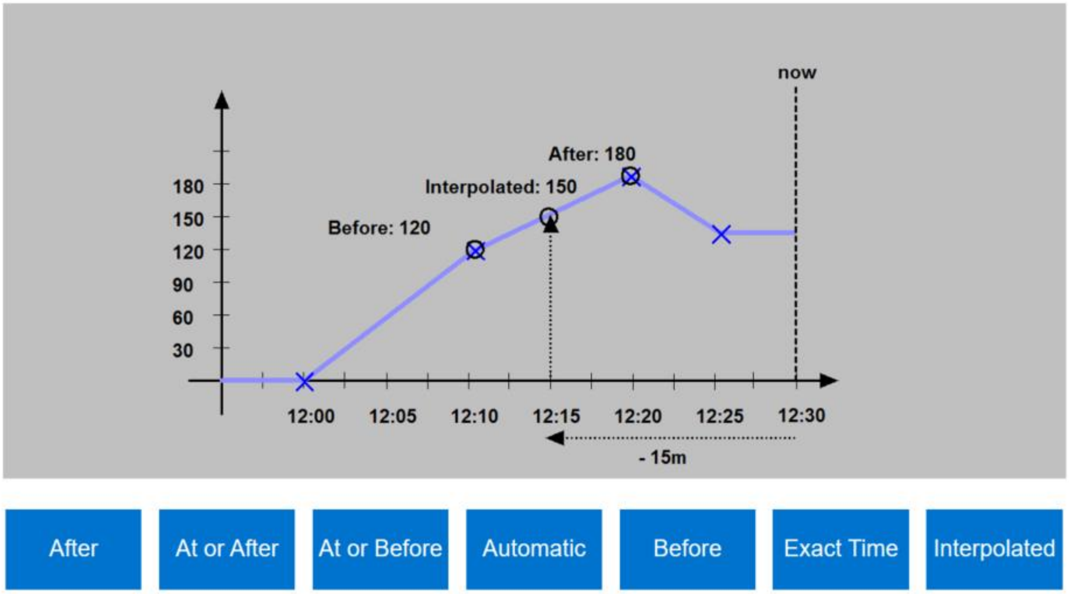
☑ Read only

By Time	Relative Time	Meaning
At or After	-15m	Returns the recorded value from 15 minutes before the current time. If no value exists at that time, the next recorded value is returned.
After	-2h	Returns the first recorded value after the point in time that is two hours ago. If a value exists exactly at that time, it is not returned.

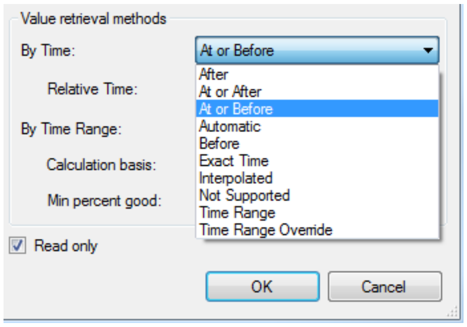
Exact	-2h	Returns the recorded value from two hours before the current time. If no value exists at that time, “No Data”-error is returned.
Interpolated	T+6h	Returns an interpolated time for 6:00:00 of the current day.

Note: Do not choose the Not Supported, Time Range, and Time Range Override options for **By Time**. These options are for attribute values based on time range calculations (see below**)**

By Time Range – Retrieving the result of an aggregation



By Time mode options are **After**, **At or Before**, **At or After**, **Automatic**, **Before**, **Exact Time** and **Interpolated**. To get a value other than the current value these options are used in context with the **Relative Time** field. Relative time expressions have to be in the PI System Time format (see examples below).



Value retrieval methods

By Time: At or Before

Relative Time:

By Time Range:

Calculation basis:

Min percent good:

☑ Read only

By Time	Relative Time	Meaning
At or After	-15m	Returns the recorded value from 15 minutes before the current time. If no value exists at that time, the next recorded value is returned.
After	-2h	Returns the first recorded value after the point in time that is two hours ago. If a value exists exactly at that time, it is not returned.
Exact	-2h	Returns the recorded value from two hours before the current time. If no value exists at that time, “No Data”-error is returned.
Interpolated	T+6h	Returns an interpolated time for 6:00:00 of the current day.

Note: Do not choose the Not Supported, Time Range, and Time Range Override options for **By Time**. These options are for attribute values based on time range calculations (see below**)**

Lesson 5 – Create Templates in AF

Introduction to and Advantages of AF Templates

The Power of Templates

The term *template* suggests a cookie-cutter way to create new objects with consistency. AF element templates do this and more for elements. They ensure similar equipment stay alike in AF.

When changes (in default values, attribute definition, or structure) are made to the template, the elements created using that template are automatically updated.

Note: If changes were made at the element's attribute level, a change to the attribute template will not replace the modifications made by the user. That is, if you uniquely set values or configuration strings in the data references of unique elements' attributes, they will not be overwritten by a modification of those attributes' default values at the element template level.

Attributes derived from a template have a *Reset to Template* option on their right-click menu to remove any unique value or configuration string definition for this attribute and reset it to the default values established at the template level.

This allows to create, but also to maintain many elements with the template, knowing that changes are applied to all appropriate related elements.

Templates also allow the AF database to grow into an analysis tool when ready to do so. It is possible to add calculations and analytics into the template at any time since that information will be propagated to all of the elements automatically. It is not necessary to have the complete analysis at the time of template creation.

In this example we created a template from an existing element, but it is possible to go directly to the Library to create a new template by right-clicking Element Templates and selecting New Template or by clicking the New Template button from the toolbar.

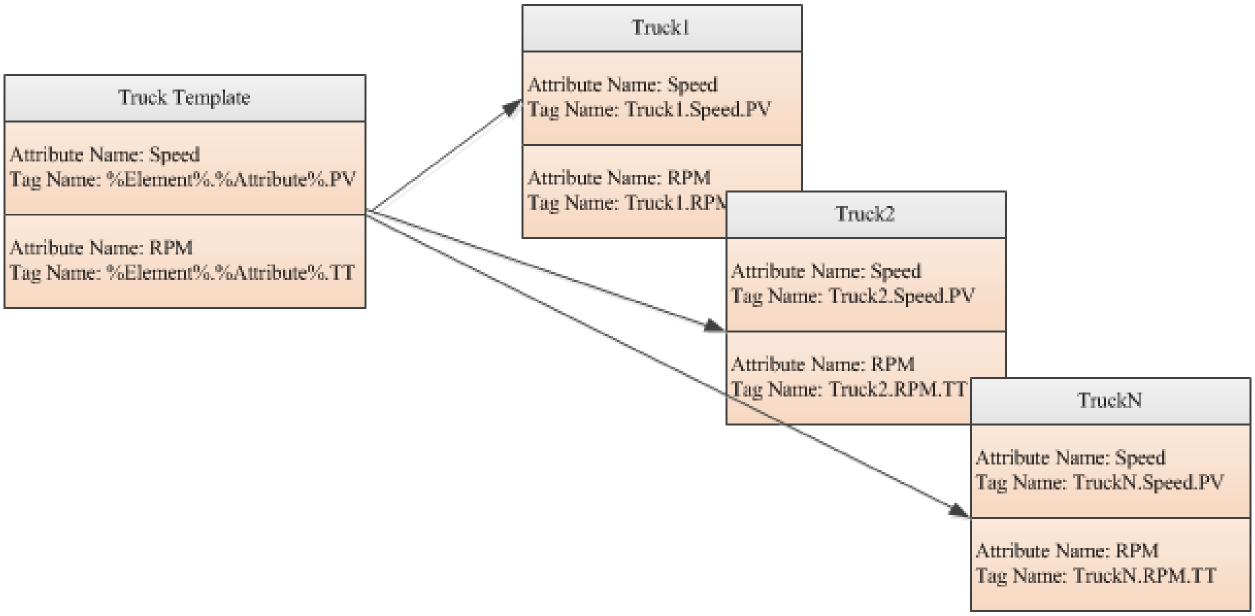
AF Templates have an □ Allow Extensions check box under the General tab. Checking this box allows for additional attributes to be created at the element level, outside of the element template. Obviously, this will no longer guarantee all elements have the same set of attributes.

Templates may be the most powerful and unique feature in AF. Very few applications would not benefit of the use of templates when building AF elements.

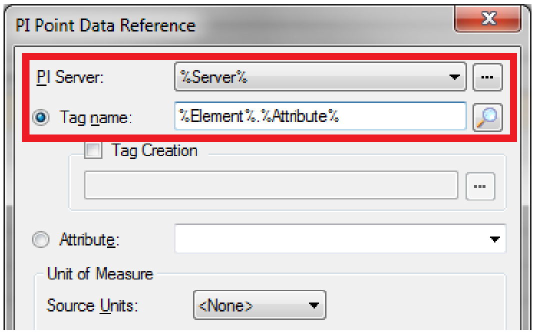
Use Substitution Parameters with Attribute Templates

Substitution parameters are variables that you place in attribute templates for PI point data references. AF resolves a substitution parameter when elements are created. As an example, the substitution parameter *%Element%* would resolve to the attribute's element name.

For example, in a truck template, the tag for the speed attribute is configured as *%Element%.%Attribute%.PV*. Once Truck1 is created from this template, AF would try to find a tag named *Truck1.Speed.PV*.



Notice the default configuration for a PI Point Data Reference attribute template: *\\%Server%\%Element%.%Attribute%.*


PI Point Data Reference

PI Server: %Server%

● Tag name: %Element%.%Attribute%

□ Tag Creation

○ Attribute:

Unit of Measure

Source Units: <None>

Substitution parameters can also be applied in Table Lookup and String Builder Data References.

Defining Substitution Parameters

The substitution parameters are listed in the following table. The ones in bold are the commonly used “Name” substitution parameters.

Parameter	Will be replaced by this object’s name:
%..\Element%	The name of the owning element of the element in which the attribute resides. To retrieve further ancestors, use the ‘..\’ notations, such as %..\..\Element%.
%|Attribute%	Name of the root attribute or attribute template that holds this data reference.
%..|Attribute%	The name of the owning attribute in which the attribute resides. To retrieve further ancestors, use the ‘..|’ notations, such as %..|..|Attribute%.
%@Attribute%	The value of the attribute referenced. To retrieve further ancestors, use the ‘..|’ notations, such as %@..|..|Attribute%.
%\Element%	The name of the root AF Element in which the attribute resides.
%<Environment Variable>%	The matching System Environment Variable’s value. For example %COMPUTERNAME% is replaced with the name of the computer on which the Data Reference is executing.
%Analysis%	The name of the analysis if it can be obtained from the context.
%Attribute%	The name of the attribute that holds this data reference.
%AttributeId%	The attribute ID that holds this data reference.
%Database%	The name of the AF Database in which the attribute resides.
%Description%	The description of the attribute that holds this data reference.

%Element%	The name of the AF Element in which the attribute resides.
%ElementDescription%	The description of the element in which the attribute resides.
%ElementId%	The element ID that holds this data reference.
%EndTime%	The local end time if it can be obtained from the time context.
%Model%	The name of the model if it can be obtained from the context.
%Server%	The name of the default Data Archive of the AF Database in which the attribute resides.
%StartTime%	The local start time if it can be obtained from the time context.
%System%	The name of the PI System in which the attribute resides.
%Time%	The local time if it can be obtained from the time context.
%UtcEndTime%	The coordinated universal (UTC) end time if it can be obtained from the time context.
%UtcStartTime%	The coordinated universal (UTC) start time if it can be obtained from the time context.
%UtcTime%	The coordinated universal (UTC) time if it can be obtained from the time context.
.\	The current reference
[.]	The default object of the parent collection. For example .\Elements[.] |Temperature returns the temperature attribute from the primary element of the current reference’s Elements collection.
[@filter=text]	The search string in text (e.g. Tank*) matches the given filter. Supported filters are: @Name, @Index, @Template, @Category, @ReferenceType, @Description, @Type, @UOM.

[@Index=#]	Returns the result at location # from the collection result.

The following table lists the substitution parameters that are useful for event frames naming patterns.

Parameter Name	Substitution
%EventFrame%	Name of the event frame in which the attribute resides.
%..\EventFrame%	Name of the parent event frame of the event frame in which the attribute resides. To retrieve further ancestors, use the ..\ notation such as %..\..\EventFrame%.
%StartTime%, %EndTime%	Local start or end time, if obtainable from the time context.
%UtcStartTime%, %UtcEndTime%	Coordinated universal (UTC) start or end time if it can be obtained from the time context.

.

Reading Substitution Parameters

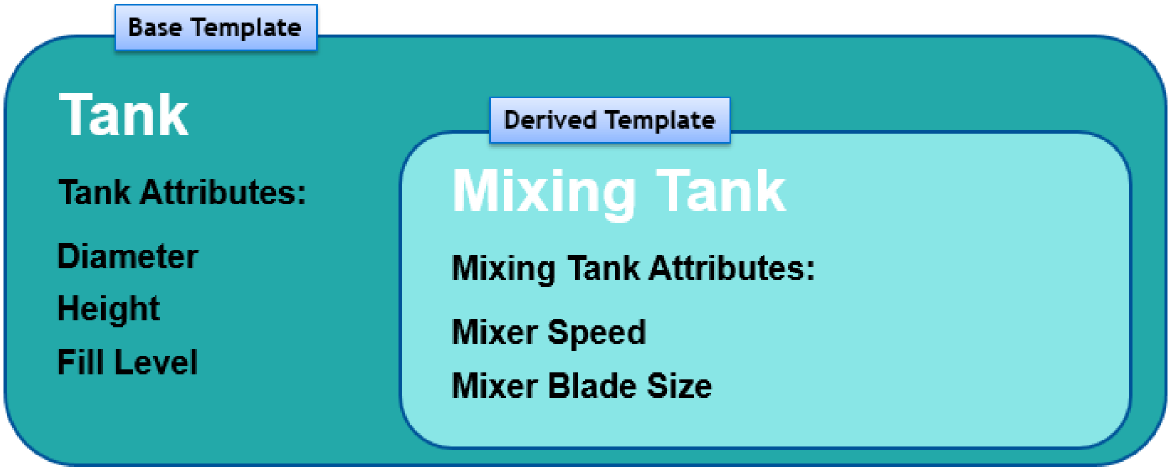
In order to better understand configuration strings in substitution parameters, refer to the following table:

<table>
<tr><th>Symbol</th><th>Definition</th><th>Examples</th></tr>
<tr><td>%[…]%</td><td>Consider the expression as a substitution parameter.</td><td>**%**Element**%**,
%Attribute**%**</td></tr>
<tr><td>.</td><td>Navigate a level down.</td><td>%@.\<ChildElement>|<Attribute>%</td></tr>
<tr><td>..</td><td>Navigate a level up.</td><td>%**..****..**\Element%, %**..**|Attribute%</td></tr>
<tr><td>\</td><td>References an element.</td><td>%..****Element%</td></tr>
<tr><td>|</td><td>References an attribute.</td><td>%..**|**Attribute%, %@**|**<Attribute>%</td></tr>
<tr><td>@</td><td>References the value of the object instead of its name.</td><td>%@..\..\|<Attribute>%</td></tr>
</table>

When referring attribute values there are differences between substitution parameters used in PI Point Data References and the syntax in the String Builder. %@..\|SiteID% works in the PI point reference but has to be ‘..\|SiteID’ instead for the String Builder.

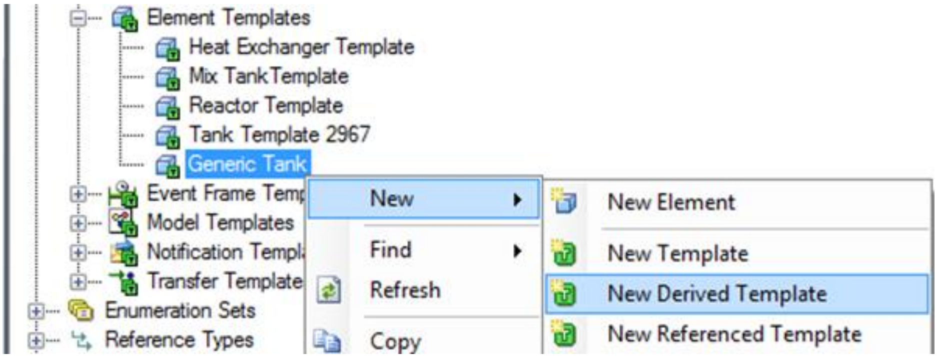
Build Derived Templates in AF

A powerful feature of the element template is the ability to set a base template. Once a base template is created, it can be used to create a number of derived templates. When an element is created from a derived template, the element contains all attributes from both the base template and the derived template.

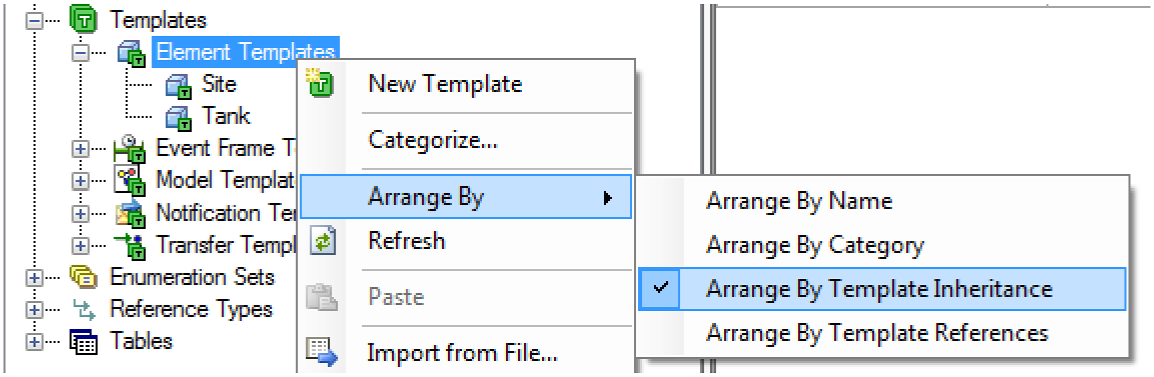


- An element created from the Tank element template has three attributes: Diameter, Height and Fill Level.
- An element created from the Mixing Tank element template has five attributes: Diameter, Height, Fill Level, Mixer Speed and Mixer Blade Size

A base template is best used when you are modeling elements that have a set of attributes in common with a few attributes that differ. For example, if you have a set of tanks, some with two valves and some with one valve, you can create an element template for the one-valve models and use that as the base template for the two-valve models. Set the base template of an element template in the *General* tab; alternatively, you can set the base template at creation time by right clicking the base template and select *New Derived Template*.



To view the template inheritance tree from the PSE Library, simply organized the templates by inheritance.



Lesson 6 – Build Tags in AF

Use Advanced Substitution Parameters with Attribute Templates

Reading Substitution Parameters

In order to better understand configuration strings in substitution parameters, refer to the following table:

<table>
<tr><th>Symbol</th><th>Definition</th><th>Examples</th></tr>
<tr><td>%[…]%</td><td>Consider the expression as a substitution parameter.</td><td>%Element%,
%Attribute%</td></tr>
<tr><td>.</td><td>Navigate a level down.</td><td>%@.\&lt;ChildElement>|<Attribute&gt;%</td></tr>
<tr><td>..</td><td>Navigate a level up.</td><td>%..\..\Element%, %..|Attribute%</td></tr>
<tr><td>\</td><td>References an element.</td><td>%..\Element%</td></tr>
<tr><td>|</td><td>References an attribute.</td><td>%..|Attribute%, %@|<Attribute&gt;%</td></tr>
<tr><td>@</td><td>References the value of the object instead of its name.</td><td>%@..\..\|<Attribute&gt;%</td></tr>
</table>

When referring attribute values there are differences between substitution parameters used in PI Point Data References and the syntax in the String Builder. %@..\|SiteID% works in the PI point reference but has to be ‘..\|SiteID’ instead for the String Builder.

Lesson 7 – Organize Elements and Attributes

Move and Reference Elements in AF

You can move elements around in the Viewer using the mouse and keyboard. This will result in creating an element reference, or making a copy of the element or simply moving it.

Dragging an element while pressing the...	Action
Ctrl Key	Copies the original element under another parent element. The original and the copy will not be linked
Shift Key	Moves the element
No Key	Creates an Element Reference. The element will now exist in two places in the hierarchy. In this case, the element icon has a "link" designation (⧉), like in the case of a file shortcut

Tip: When moving an object, be deliberate with the shift key, drag and drop manipulation. It is safer to right click the element, copy it to the new location, and then delete the original.

Element Reference Types

Composition

The c*omposition* relationship binds two elements together so that whenever one is changed or calculated, the other is too. Deleting the parent element deletes the child as well.

For example, a valve attached to a tank may be represented as a separate element in AF, but is really a part of the tank asset and cannot exist outside of it. Removing the tank from a site removes the valve as well.

Typically, an element having a composition relationship with its parent would not be referenced elsewhere by itself

Parent-Child

With a *Parent-Child* reference, the child can have multiple parents. The child element can thus be part of multiple hierarchies.

Parent-Child is the default for a new child element creation. When you use that reference type substitution parameters in the element attributes are resolved according to the hierarchy the element is in.

Weak

A *Weak* element reference is like a Parent-Child element reference, but a weak referenced element cannot exist on its own.

Weak is applied for element references when you have a main view (e.g. master view organized according to geographical location) and you create additional views for your assets (e.g. organized according to business function). Different to a parent-child reference the element reference in an additional view will not exist on its own when removed in the master view.

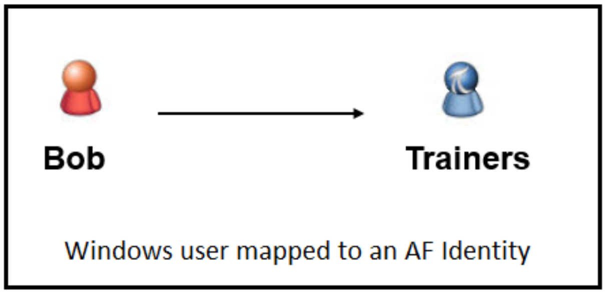
Lesson 8 - AF Security

AF Identities and Mappings

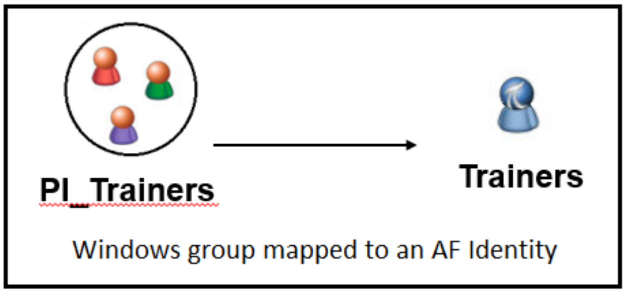
Beginning with AF version 2.7, a security model similar to Data Archive security was implemented. This model relies on Windows integrated security for authentication, but provides its own authorization to AF objects using **AF Identities** and **Mappings**. In AF version 2.6 and before, permissions in AF were set based on Windows users and groups.

An **AF identity** represents a set of access permissions on the AF server. Each AF mapping points from a Windows user or group to an AF identity. Built-In AF Identities are **Administrators**, **Engineers** and **World.** According to the specific security requirements, AF identities are created on the AF Server, and the permissions for the AF server resources (such as an element collection or objects) are granted for these AF Identities.

With **AF Mappings**, Windows users and groups are mapped to the AF Identities:



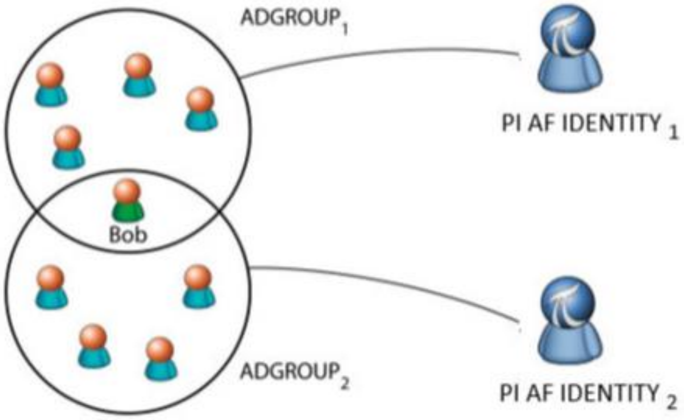
Windows user mapped to an AF Identity



Windows group mapped to an AF Identity

Note: It is preferable to define the AF Mappings for Active Directory (AD) groups instead of individual windows users. Since it is inefficient to maintain individual user accounts directly, it is recommended that the mapping on a user basis be the exception

Members of the Windows groups that are mapped to an AF identity are automatically granted the access permissions for that AF identity. For example, the Active Directory (AD) group Engineering Team is mapped to Engineers, so all the members in that AD group have read/write permission for the Elements collection. In case of multiple identities user is granted permissions based on all AF Identities to which he or she is mapped. In example below, Bob has all permissions from PI AF Identity 1 and PI AF Identity 2.

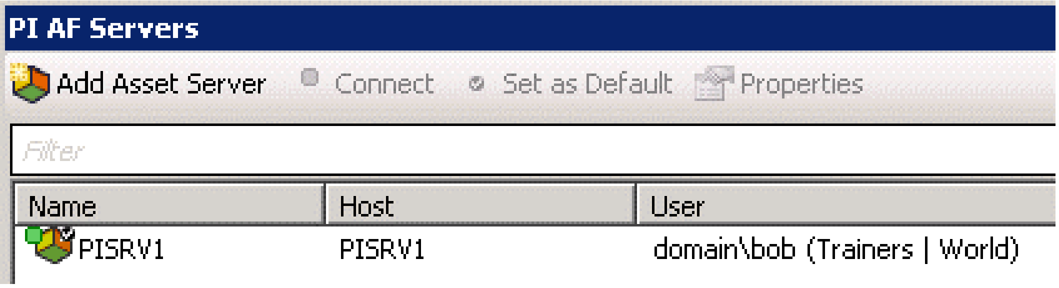


Built-in AF Identities:

AF Identity	Description
Administrators	By default, this identity has all access permissions to every collection and object on the AF server, including all databases. It cannot be modified or deleted. It is recommended that access to this identity is restricted to only a few users.
Engineers	This identity has the same privileges as *Administrators*, with the exception of the *Admin (a)* permission. This identity is also not allowed to delete AF databases. It is recommended that this identity be restricted to those users who are defining the asset database. Additional identities should be created to narrow the scope of access within AF.
World	This identity has read access permissions to every collection and object on the AF server. More information see below.

World Identity

By default, World identity is mapped to the Windows Everyone users group:

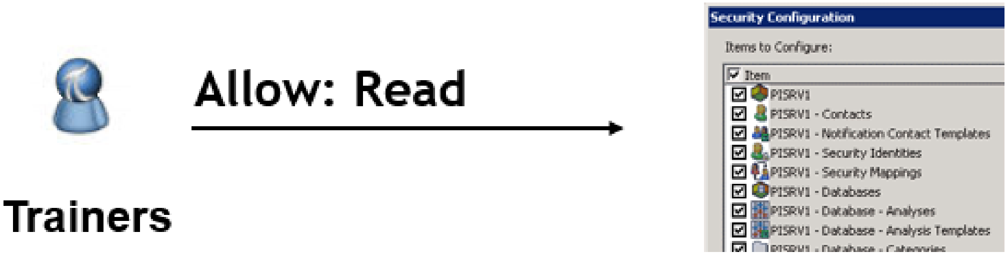


Name	Host	User
PISRV1	PISRV1	domain\bob (Trainers | World)

By default, World Identity has Read permissions on all items. World identity cannot be modified or deleted. However, Mappings for World can be removed and Permissions for World can be removed.

AF Access Rights

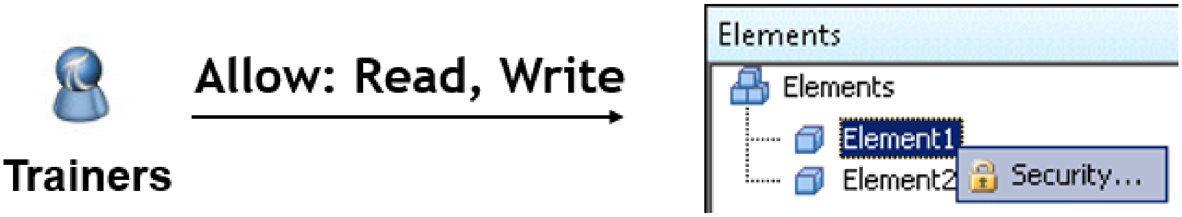
Access permissions can be granted for all AF Objects. Examples:



Security Configuration

Items to Configure:

- ☑ Item
- ☑ PISRV1
- ☑ PISRV1 - Contacts
- ☑ PISRV1 - Notification Contact Templates
- ☑ PISRV1 - Security Identities
- ☑ PISRV1 - Security Mappings
- ☑ PISRV1 - Databases
- ☑ PISRV1 - Database - Analyses
- ☑ PISRV1 - Database - Analysis Templates
- ☑ PISRV1 - Database - Categories



The following table describes the access permissions you can assign to AF identities for all objects in the AF hierarchy.

Access right	Abbreviation	Definition
Read	r	Enables a user to view the object.
Write	w	Enables a user to create and modify an object. The exception is that event frames and transfers also require Write Data permission on the element template from which they are created. Additionally, if users do not have Write permission on the AF database, they cannot modify any object within the database, regardless of the specific permission on that object.
Read/Write		Enables a user to read and write to the associated object.
Read Data	rd	Enables a user to read element's attribute values (non-configuration items).
Write Data	wd	Enables a user to modify element's attribute values (non-configuration items). Additionally, this permission controls whether a user can create or modify event frames.
Subscribe	s	Enables a user to subscribe and unsubscribe to a notification.
SubscribeOthers	so	Enables a user to subscribe and unsubscribe other users to a notification.
Delete	d	Enables a user to delete an object.
Execute	x	Enables a user to perform most actions on an analysis case. Only used in Pimsoft SigmafineTM data reconciliations. The PI Analysis Service does not use this permission. The Write permission is required to modify, run, and stop asset analyses.
Admin	a	Enables a user to modify the security settings, or owner, of an object. Also allows to force an Undo Check Out on an object that is checked out to another user, as well as to lock and unlock an event frame..

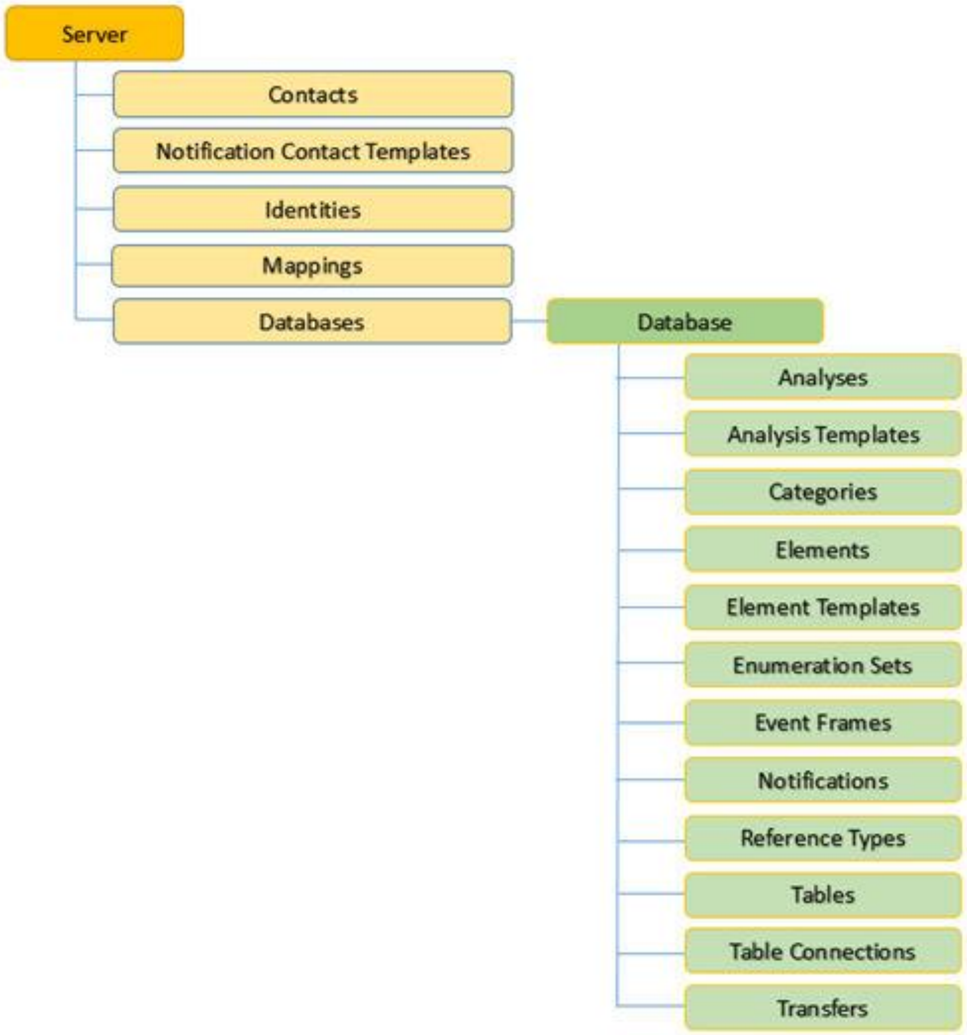
Setting permissions can be done for individual AF objects or for collections of objects. When you create new objects, except for child elements, the collection security is used as the default security. When you create a child element, the security descriptor of the parent element becomes its default security.

AF Server Security

AF Security Hierarchy

The following chart shows the structure of the AF objects in a AF Server. Each securable AF object (element, event frame, and notification, and so on) throughout the hierarchy has an associated security descriptor that contains the access permissions information for that object.

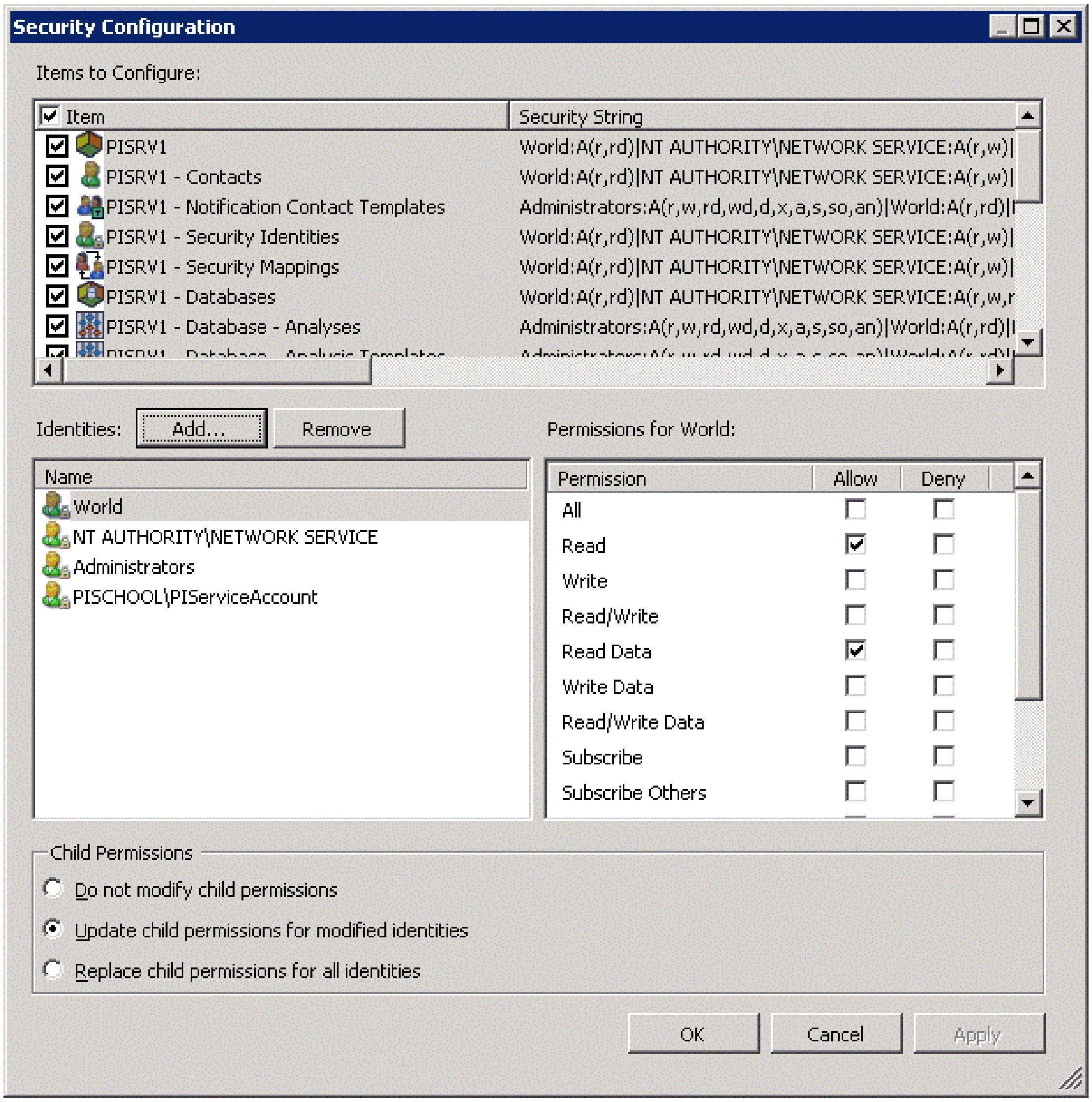
All AF objects of the same type belong to a collection. For example, every AF element in a database belongs to the Elements collection for that database. Each collection also has an associated security descriptor that contains access permission information.



Setting the security for a AF Server

To open *Security Configuration* window for AF Server: On the toolbar, click the **Database** button. In the *Select Database* window, click the **Edit Security** button.

Alternatively, click the AF Server Properties button to open the AF Server Properties window, then click the blue **Security** link below the Aliases field. The *Security Configuration* window displays the defined access rights and allows to change them.



Security Configuration

Items to Configure:

☑ Item	Security String
☑ PISRV1	World:A(r,rd)|NT AUTHORITY\NETWORK SERVICE:A(r,w)|
☑ PISRV1 - Contacts	World:A(r,rd)|NT AUTHORITY\NETWORK SERVICE:A(r,w)|
☑ PISRV1 - Notification Contact Templates	Administrators:A(r,w,rd,wd,d,x,a,s,so,an)|World:A(r,rd)|
☑ PISRV1 - Security Identities	World:A(r,rd)|NT AUTHORITY\NETWORK SERVICE:A(r,w)|
☑ PISRV1 - Security Mappings	World:A(r,rd)|NT AUTHORITY\NETWORK SERVICE:A(r,w)|
☑ PISRV1 - Databases	World:A(r,rd)|NT AUTHORITY\NETWORK SERVICE:A(r,w,r
☑ PISRV1 - Database - Analyses	Administrators:A(r,w,rd,wd,d,x,a,s,so,an)|World:A(r,rd)|
☑ PISRV1 - Database - Analysis Templates	Administrators:A(r,w,rd,wd,d,x,a,s,so,an)|World:A(r,rd)|

Identities: Add... Remove

Name
World
NT AUTHORITY\NETWORK SERVICE
Administrators
PISCHOOL\PIServiceAccount

Permissions for World:

Permission	Allow	Deny
All	□	□
Read	☑	□
Write	□	□
Read/Write	□	□
Read Data	☑	□
Write Data	□	□
Read/Write Data	□	□
Subscribe	□	□
Subscribe Others	□	□

Child Permissions

○ Do not modify child permissions

● Update child permissions for modified identities

○ Replace child permissions for all identities

The **Items to configure** list for the AF Server contains the following items:

- AF Server
- Contacts Collection
- Notification Contact Templates Collection
- Identities Collection
- Mappings Collection
- Databases Collection (entire AF hierarchy)
- Analyses Collection (entire AF hierarchy)
- Analysis Templates Collection (entire AF hierarchy)
- Categories (entire AF hierarchy)
- Elements Collection (entire AF hierarchy)
- Element Templates Collection (entire AF hierarchy)
- …

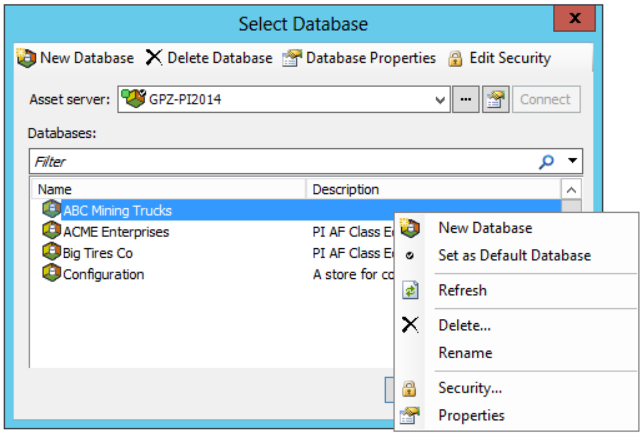
The Security Configuration allows adding, removing or changing the permissions for AF Identities.

- You can uncheck/check items to control the scope of the changes you are going to apply.
- You can modify permissions for one of the AF identities that is listed, or add or remove identities, as needed.
- The child permissions option defines the handling concerning permission inheritance.

Setting the security for a AF Database

To open *Security Configuration* window for an AF database:

On the toolbar, click the *Database* button. In the *Select Database* window, right-click a database in the **Databases** list and select **Security**.

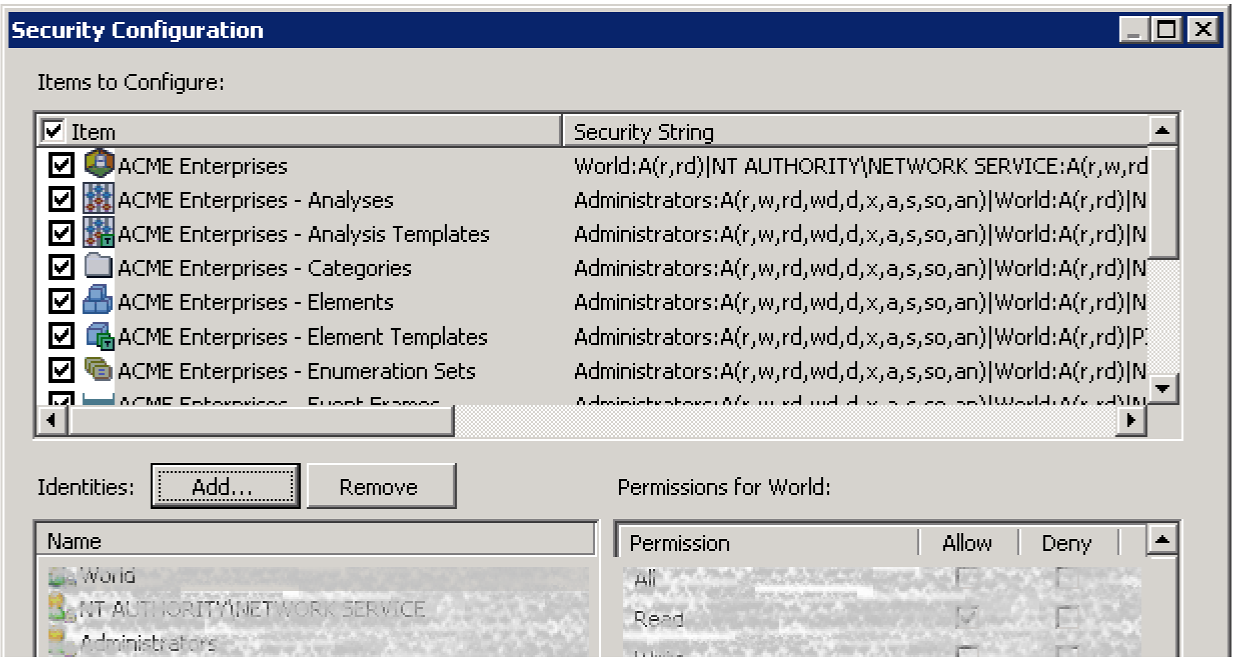


Asset server: GPZ-PI2014

Databases:

Name	Description
ABC Mining Trucks	
ACME Enterprises	PI AF Class E
Big Tires Co	PI AF Class E
Configuration	A store for c

In the **Items to Configure** list of the *Security Configuration* window, the selected database and every collection is checked.



Security Configuration

Items to Configure:

☑ Item	Security String
☑ ACME Enterprises	World:A(r,rd)|NT AUTHORITY\NETWORK SERVICE:A(r,w,rd
☑ ACME Enterprises - Analyses	Administrators:A(r,w,rd,wd,d,x,a,s,so,an)|World:A(r,rd)|N
☑ ACME Enterprises - Analysis Templates	Administrators:A(r,w,rd,wd,d,x,a,s,so,an)|World:A(r,rd)|N
☑ ACME Enterprises - Categories	Administrators:A(r,w,rd,wd,d,x,a,s,so,an)|World:A(r,rd)|N
☑ ACME Enterprises - Elements	Administrators:A(r,w,rd,wd,d,x,a,s,so,an)|World:A(r,rd)|N
☑ ACME Enterprises - Element Templates	Administrators:A(r,w,rd,wd,d,x,a,s,so,an)|World:A(r,rd)|P:
☑ ACME Enterprises - Enumeration Sets	Administrators:A(r,w,rd,wd,d,x,a,s,so,an)|World:A(r,rd)|N
☑ ACME Enterprises - Event Frames	Administrators:A(r,w,rd,wd,d,x,a,s,so,an)|World:A(r,rd)|N

Identities: Add... Remove

Permissions for World:

Name
World
NT AUTHORITY\NETWORK SERVICE
Administrators

Permission	Allow	Deny
All	□	□
Read	☑	□
Write	□	□

The **Items to configure** list for the AF Database contains the following items:

AF Database

Analyses Collection

Analysis Templates Collection

Categories

Elements Collection

Element Templates Collection

…

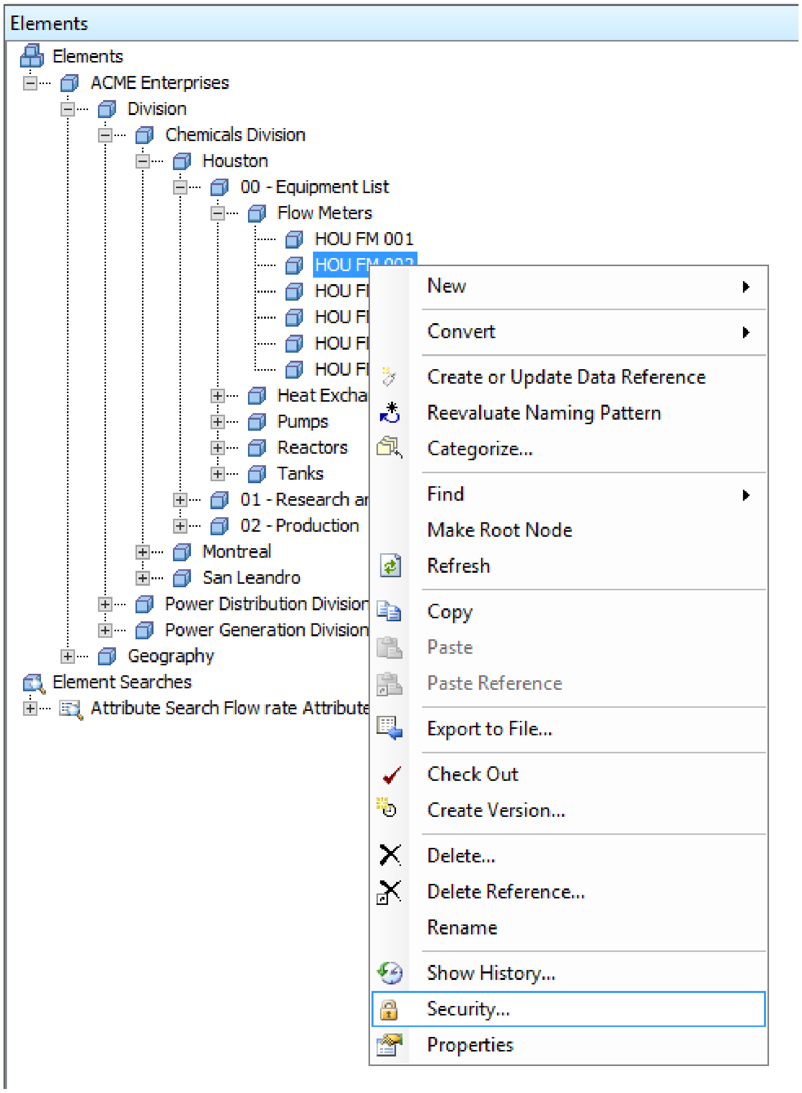
The Security Configuration allows adding, removing or changing the permissions for AF Identities. You can uncheck/check items to control the scope of the changes you are going to apply. You can modify permissions for one of the AF identities that is listed, or add or remove identities, as needed. The child permissions option defines the handling concerning permission inheritance.

Setting the security for a AF Collection

You can configure access permissions to collections (Elements collection, Event Frame collection, Templates collection...) at several points in the AF hierarchy. You can set them at the server level or at the database level. If set at the server level, the permissions assigned to identities on the server are also assigned to the same identities in every database.

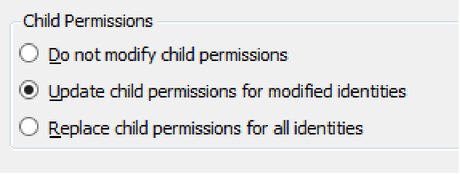
AF Object Security

You can set specific access permissions for an identity that differ from the default settings inherited from elsewhere in the AF hierarchy on any object (or object group) and collection in a database.



Permission Inheritance

When you change the access permissions for an element, the following applies for access permissions of child elements:

Reference Type	Behavior
Composition	Access permissions for child and parent are always the same.
Weak	Access permissions are never inherited.
Parent – Child	When access permissions are set on a parent, the Child Permission settings in the Security Configuration window depend on option used  Child Permissions ○ Do not modify child permissions ● Update child permissions for modified identities ○ Replace child permissions for all identities

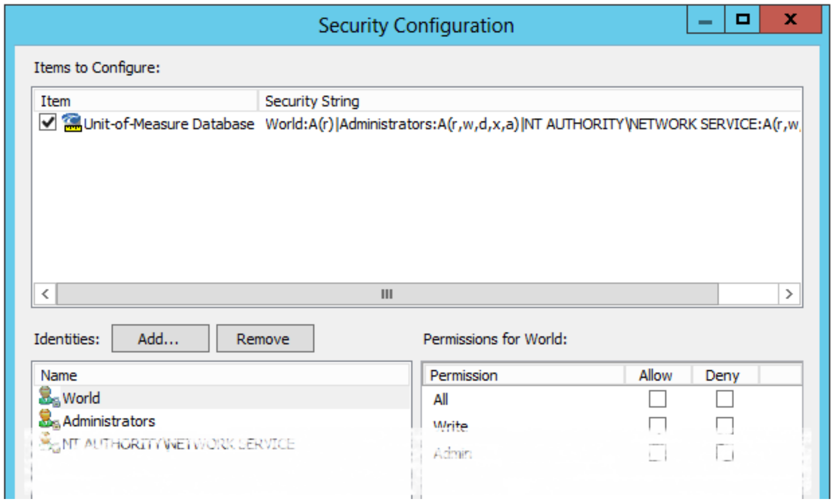
Option	Description
Do not modify child permissions	Prevents access permissions that have been set for the current object or collection from being replicated to child collections and objects in the AF hierarchy. Default for AF server 2.5 and earlier
Update child permissions for modified identities	For 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. Default for AF server 2.6 and later.
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. Hint: Before you apply this option, review access permission settings for all items on the Items to Configure list to avoid unintentionally overwriting custom permissions that may have been applied elsewhere in the collection hierarchy!

UOM Database Security

The AF Unit of Measure (UOM) database is shared across all AF databases. You cannot set permissions for individual UOMs or UOM classes. Permissions for individual UOMs or UOM classes. Permissions can only be set for the entire UOM database.

To open UOM *Security Configuration* window:

1. In the Navigator pane, select **Unit of Measure.**
2. On the toolbar, click the **UOM Security** button.



Security Configuration

Items to Configure:

Item	Security String
☑ Unit-of-Measure Database	World:A(r)|Administrators:A(r,w,d,x,a)|NT AUTHORITY\NETWORK SERVICE:A(r,w,

Identities: Add... Remove

Name
World
Administrators
NT AUTHORITY\NETWORK SERVICE

Permissions for World:

Permission	Allow	Deny
All	□	□
Write	□	□
Admin	□	□

Security Tips

- Administrator privileges at the server level provides access to **every object**, regardless of their security settings
- If you want to edit an element, you need *write* permissions on the Elements collection **and** on the particular element
- Library objects, such as templates, enumeration sets, UOMs, and references types **always** have read permission regardless of their security settings
- Deny settings override any granted allow permissions