Techniques for Optimizing Asset Analytics

Presented by: Nathaniel Adebayo

Agenda

- Asset Analytics Overview
- Best Practices
  - Building expressions
  - Using Event Frames to capture critical events
  - Backfilling and Recalculations
  - Management
- Summary

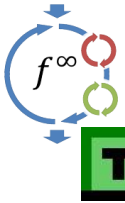
Asset Analytics Overview

Asset Analytics

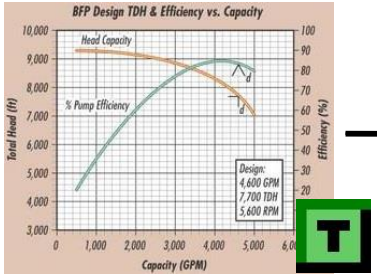
- 150+ built in functions
- Streaming calculations triggered by events or clock
- Output to PI Points for reporting and trending
- Create event frames and rollups
- Trigger notifications
- Supports templates, fully integrated into PI AF
- Backfill and Manual recalculation
- Auto-recalculation for late or out-of-order data

Workflow

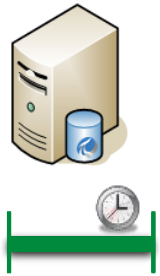
Configure

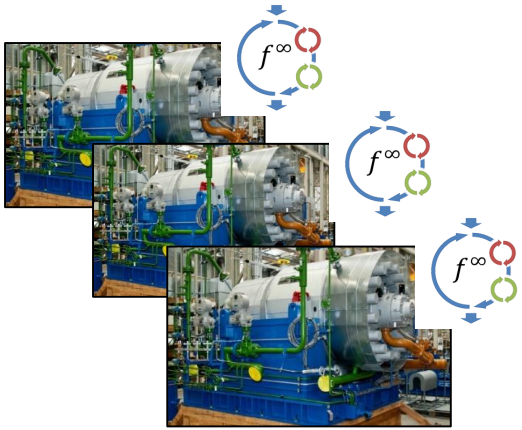


Test

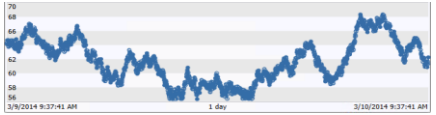


Schedule

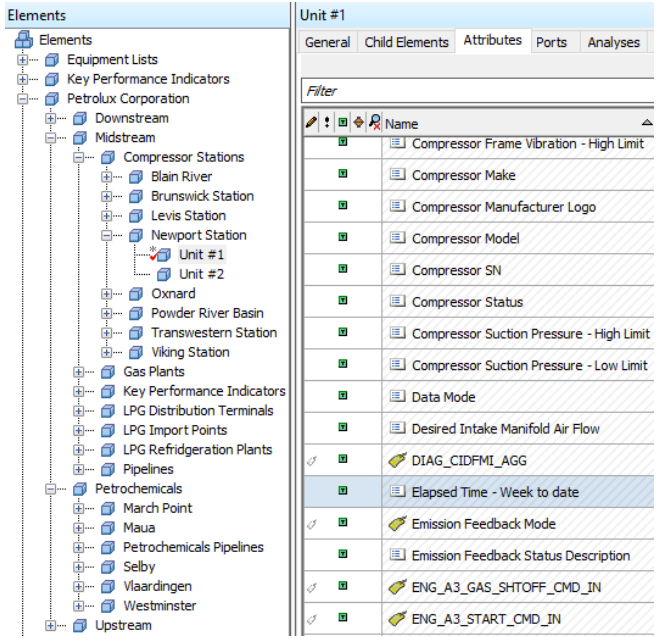




Backfill/Recalculate



A bit about Context…

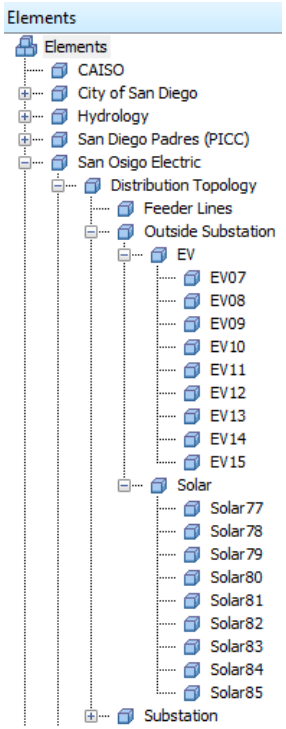


Typical use case:

I want to monitor and detect problems with my compressor, before they become catastrophic.

- AF models assets and processes
- Inputs (typically) come from the asset
- Analysis outputs are mapped on the asset

A bit about Context…

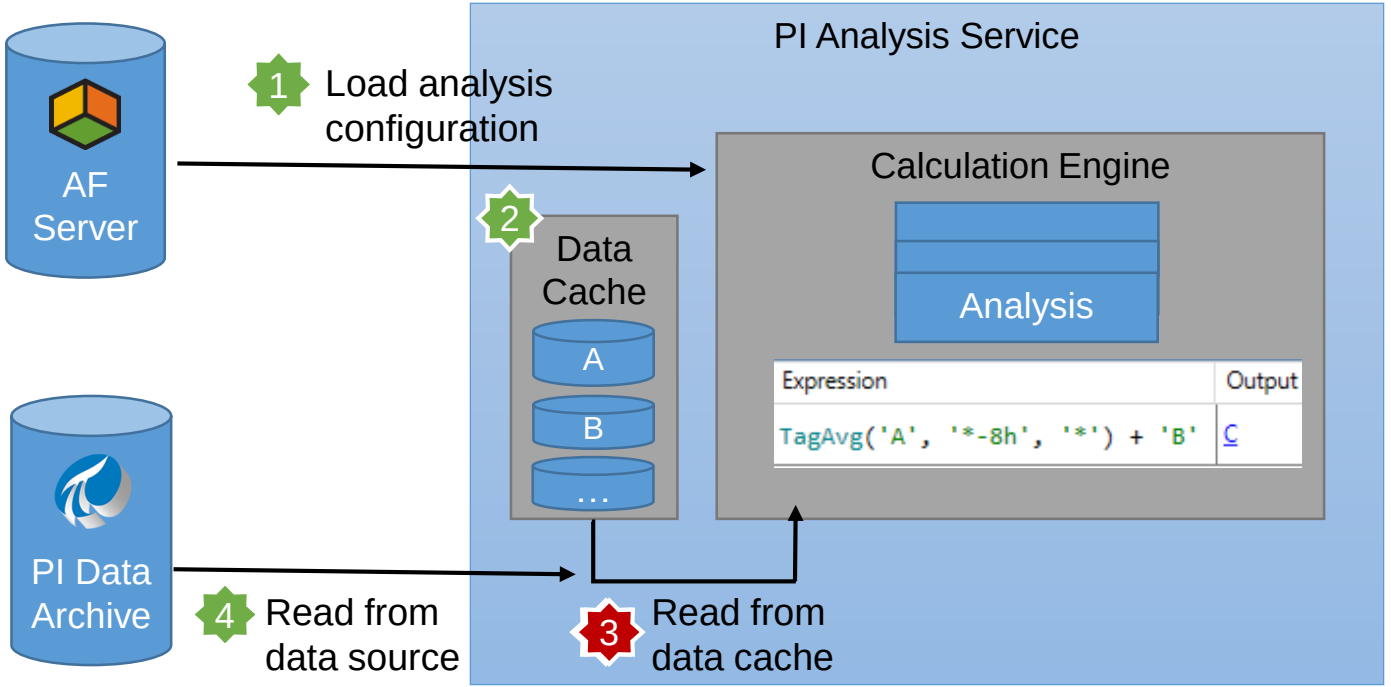


Typical use case:

I want to aggregate EV and Solar production for my substations.

- Inputs come from asset hierarchy
- Analysis outputs used in dashboards, CBM, KPIs & reports

PI Analysis Service - Overview



Design Tradeoffs

- Optimized for...
  - Streaming analytics use case
  - Real-time calculations take priority over recalculations
  - Easy configuration - no programming experience required

- Not suitable for...
  - Executing queries across really large number of attributes
  - Extracting large amounts of time series data
  - Ad-hoc calculations

Best Practices

Building Expressions

Use of variables

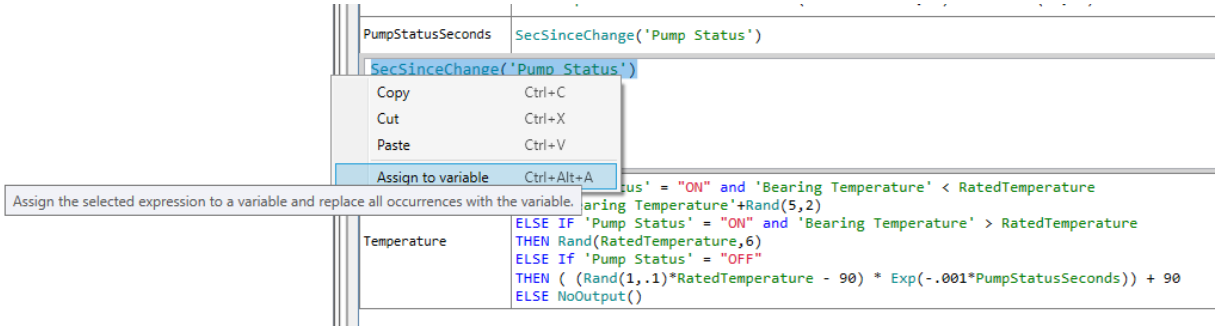
Add a new variable | Evaluate

Name	Expression	Output Attribute	
RatedFlowRate	// Typical Flow Rate of the Pump 450	Map	⊗
RatedPressure	// Typical Pressure of the Pump 150	Map	⊗
RatedTemperature	// Typical Temperature of the Pump 170	Map	⊗
FlowRate	IF 'Pump Status' = "ON" THEN Rand(RatedFlowRate,35) ELSE 0	Discharge Flow Rate	⊗
Pressure	IF 'Pump Status' = "ON" THEN Rand(RatedPressure,10) ELSE Rand(.1,.2)	Suction Pressure	⊗
PumpStatusSeconds	SecSinceChange('Pump Status')	Map	⊗
Temperature	IF 'Pump Status' = "ON" and 'Bearing Temperature' < RatedTemperature THEN 'Bearing Temperature'+Rand(5,2) ELSE IF 'Pump Status' = "ON" and 'Bearing Temperature' > RatedTemperature THEN Rand(RatedTemperature,6) ELSE If 'Pump Status' = "OFF" THEN ((Rand(1,.1)*RatedTemperature - 90) * Exp(-.001*PumpStatusSeconds)) + 90 ELSE NoOutput()	Bearing Temperature	⊗

Building Expressions

Use of variables

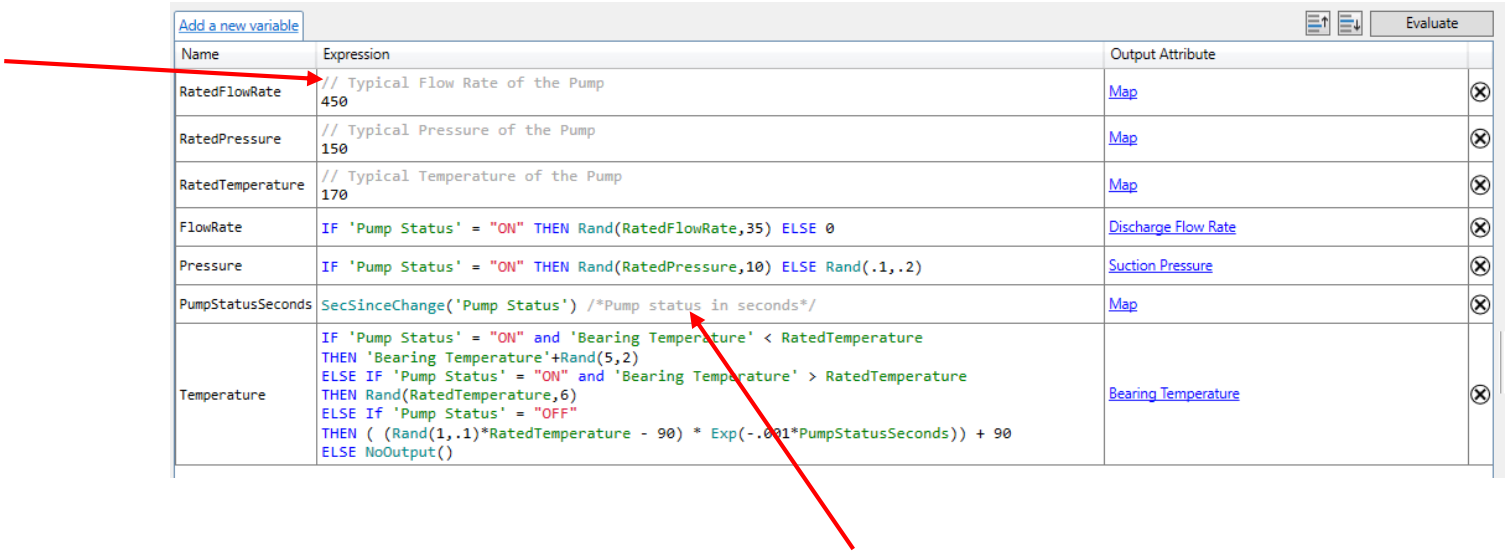
Easily assign variables to expressions, by highlighting and right-clicking on the highlighted text:



Building Expressions

Comments

Use of comments in expressions helps with readability:



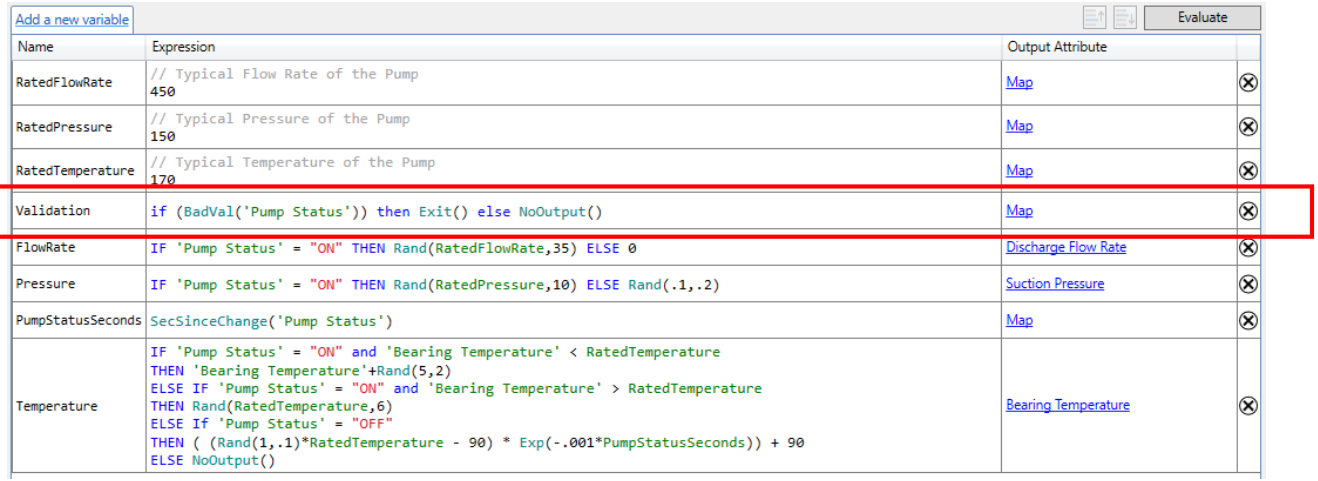
Add a new variable | Evaluate

Name	Expression	Output Attribute
RatedFlowRate	// Typical Flow Rate of the Pump 450	Map
RatedPressure	// Typical Pressure of the Pump 150	Map
RatedTemperature	// Typical Temperature of the Pump 170	Map
FlowRate	IF 'Pump Status' = "ON" THEN Rand(RatedFlowRate,35) ELSE 0	Discharge Flow Rate
Pressure	IF 'Pump Status' = "ON" THEN Rand(RatedPressure,10) ELSE Rand(.1,.2)	Suction Pressure
PumpStatusSeconds	SecSinceChange('Pump Status') /*Pump status in seconds*/	Map
Temperature	IF 'Pump Status' = "ON" and 'Bearing Temperature' < RatedTemperature THEN 'Bearing Temperature'+Rand(5,2) ELSE IF 'Pump Status' = "ON" and 'Bearing Temperature' > RatedTemperature THEN Rand(RatedTemperature,6) ELSE If 'Pump Status' = "OFF" THEN ((Rand(1,.1)*RatedTemperature - 90) * Exp(-.001*PumpStatusSeconds)) + 90 ELSE NoOutput()	Bearing Temperature

Building Expressions

Exit early

If some expressions are not needed to be evaluated based on same conditions, the Exit() function can be used:

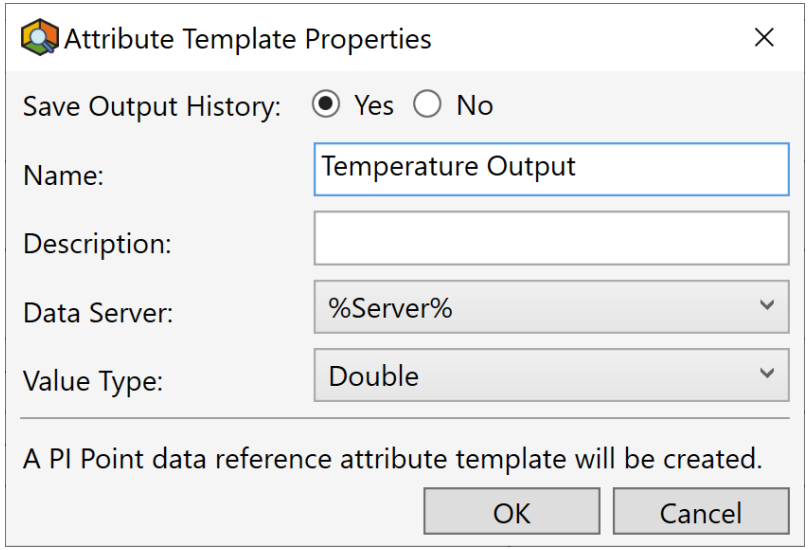


Add a new variable | Evaluate

Name	Expression	Output Attribute
RatedFlowRate	// Typical Flow Rate of the Pump 450	Map
RatedPressure	// Typical Pressure of the Pump 150	Map
RatedTemperature	// Typical Temperature of the Pump 170	Map
Validation	if (BadVal('Pump Status')) then Exit() else NoOutput()	Map
FlowRate	IF 'Pump Status' = "ON" THEN Rand(RatedFlowRate,35) ELSE 0	Discharge Flow Rate
Pressure	IF 'Pump Status' = "ON" THEN Rand(RatedPressure,10) ELSE Rand(.1,.2)	Suction Pressure
PumpStatusSeconds	SecSinceChange('Pump Status')	Map
Temperature	IF 'Pump Status' = "ON" and 'Bearing Temperature' < RatedTemperature THEN 'Bearing Temperature'+Rand(5,2) ELSE IF 'Pump Status' = "ON" and 'Bearing Temperature' > RatedTemperature THEN Rand(RatedTemperature,6) ELSE If 'Pump Status' = "OFF" THEN ((Rand(1,.1)*RatedTemperature - 90) * Exp(-.001*PumpStatusSeconds)) + 90 ELSE NoOutput()	Bearing Temperature

Historizing Data

Store data history into PI Points:



Working with Multiple Values

Retrieve and filter data in your calculations:

Add a new variable | Evaluate

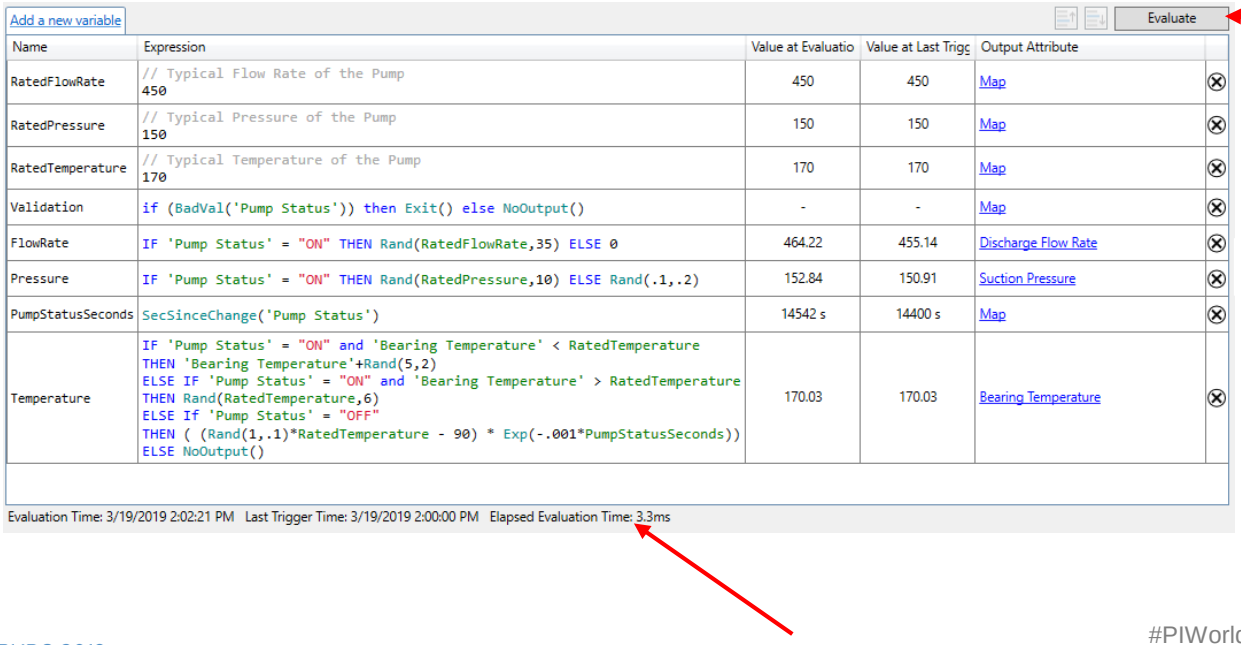
Name	Expression	Value at Evaluatio	Value at Last Trigc	Output At	
BearingTemperature20	RecordedValuesByCount('Bearing Temperature', '*', 20)	[93.889 °F, ..., 1	[93.889 °F, ..., 1	Map	⊗
BearingTemperatureDay	RecordedValues('Bearing Temperature', 't', '*')	[109.93 °F, ..., 9	[109.93 °F, ..., 9	Map	⊗
BearingTemperatureFiltered	FilterData(BearingTemperature_20, $val > 110)	[114.48 °F, ..., 1	[114.48 °F, ..., 1	Map	⊗
BearingTemperatureDayNew	MapData(BearingTemperatureDay, if $val < 100 then 100 else $val)	[109.93 °F, ..., 1	[109.93 °F, ..., 1	Map	⊗
BearingTemperatureDayNewMin	Min(BearingTemperatureDayNew)	100 °F	100 °F	Map	⊗
BearingTemperatureDayNewMax	Max(BearingTemperatureDayNew)	175.52 °F	175.52 °F	Map	⊗

Templates

- Provide manageability, consistency and governance
- Use templates for any repetitive work or for future extensions.
- A modification to the template is applied to all analyses from that template.
- Searching and filtering in UI is also easier with templates.
- More performant.

Evaluate

Perform an Evaluate to identify possible issues during configuration:



Add a new variable

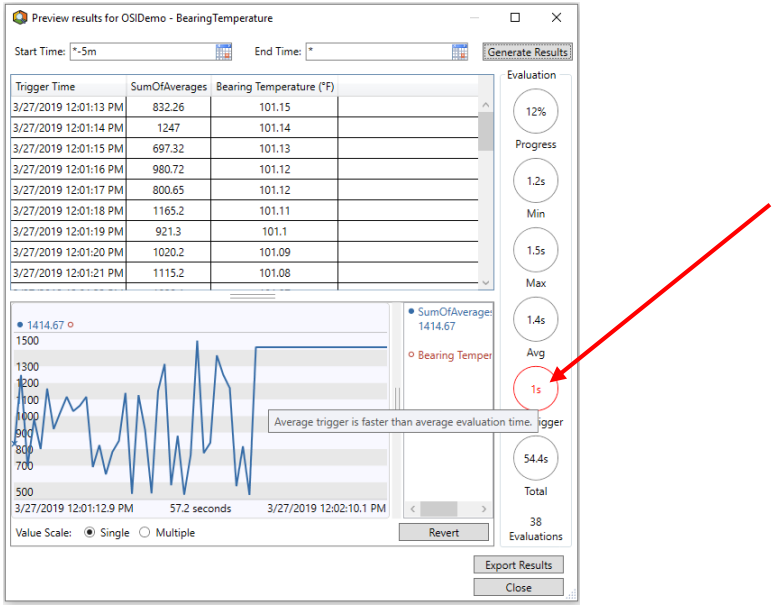
Evaluate

Name	Expression	Value at Evaluatio	Value at Last Trigg	Output Attribute
RatedFlowRate	// Typical Flow Rate of the Pump 450	450	450	Map
RatedPressure	// Typical Pressure of the Pump 150	150	150	Map
RatedTemperature	// Typical Temperature of the Pump 170	170	170	Map
Validation	if (BadVal('Pump Status')) then Exit() else NoOutput()	-	-	Map
FlowRate	IF 'Pump Status' = "ON" THEN Rand(RatedFlowRate,35) ELSE 0	464.22	455.14	Discharge Flow Rate
Pressure	IF 'Pump Status' = "ON" THEN Rand(RatedPressure,10) ELSE Rand(.1,.2)	152.84	150.91	Suction Pressure
PumpStatusSeconds	SecSinceChange('Pump Status')	14542 s	14400 s	Map
Temperature	IF 'Pump Status' = "ON" and 'Bearing Temperature' < RatedTemperature THEN 'Bearing Temperature'+Rand(5,2) ELSE IF 'Pump Status' = "ON" and 'Bearing Temperature' > RatedTemperature THEN Rand(RatedTemperature,6) ELSE If 'Pump Status' = "OFF" THEN ((Rand(1,.1)*RatedTemperature - 90) * Exp(-.001*PumpStatusSeconds)) ELSE NoOutput()	170.03	170.03	Bearing Temperature

Evaluation Time: 3/19/2019 2:02:21 PM Last Trigger Time: 3/19/2019 2:00:00 PM Elapsed Evaluation Time: 3.3ms

Preview

Preview calculation before checking in:

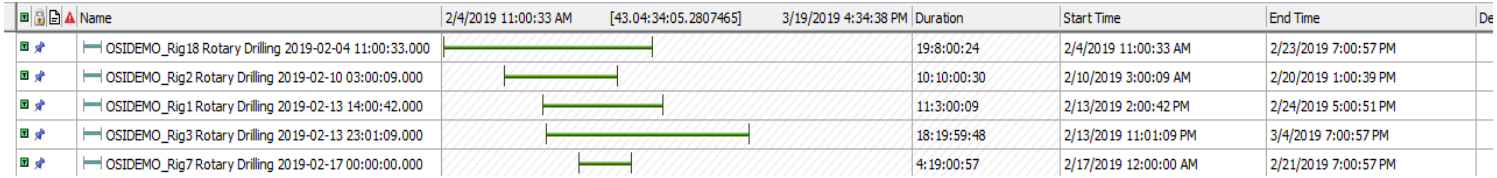


Expensive Functions

- Some functions such as summary functions (TagAvg etc.) might require a lot of data in the past to perform a calculation.
- Optimizations can be made by using a smaller range or evaluating less often if the range is larger.
- Compression on PI points can also help with density of data.

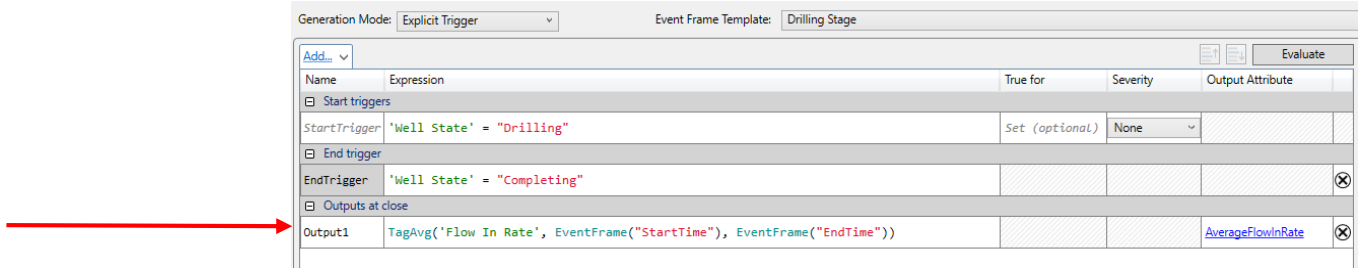
Event Frames and Notifications

- Utilize Event Frames to capture events.
- Good for reports, comparisons and tracking.



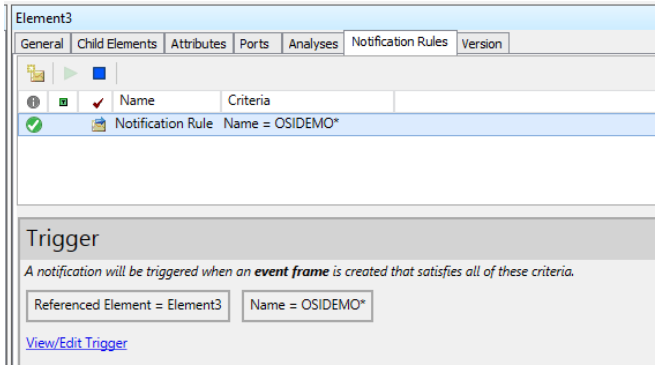
Name	2/4/2019 11:00:33 AM	[43.04:34:05.2807465]	3/19/2019 4:34:38 PM	Duration	Start Time	End Time
OSIDEMO_Rig18 Rotary Drilling 2019-02-04 11:00:33.000				19:8:00:24	2/4/2019 11:00:33 AM	2/23/2019 7:00:57 PM
OSIDEMO_Rig2 Rotary Drilling 2019-02-10 03:00:09.000				10:10:00:30	2/10/2019 3:00:09 AM	2/20/2019 1:00:39 PM
OSIDEMO_Rig1 Rotary Drilling 2019-02-13 14:00:42.000				11:3:00:09	2/13/2019 2:00:42 PM	2/24/2019 5:00:51 PM
OSIDEMO_Rig3 Rotary Drilling 2019-02-13 23:01:09.000				18:19:59:48	2/13/2019 11:01:09 PM	3/4/2019 7:00:57 PM
OSIDEMO_Rig7 Rotary Drilling 2019-02-17 00:00:00.000				4:19:00:57	2/17/2019 12:00:00 AM	2/21/2019 7:00:57 PM

- Capture data points at the close of event frame.

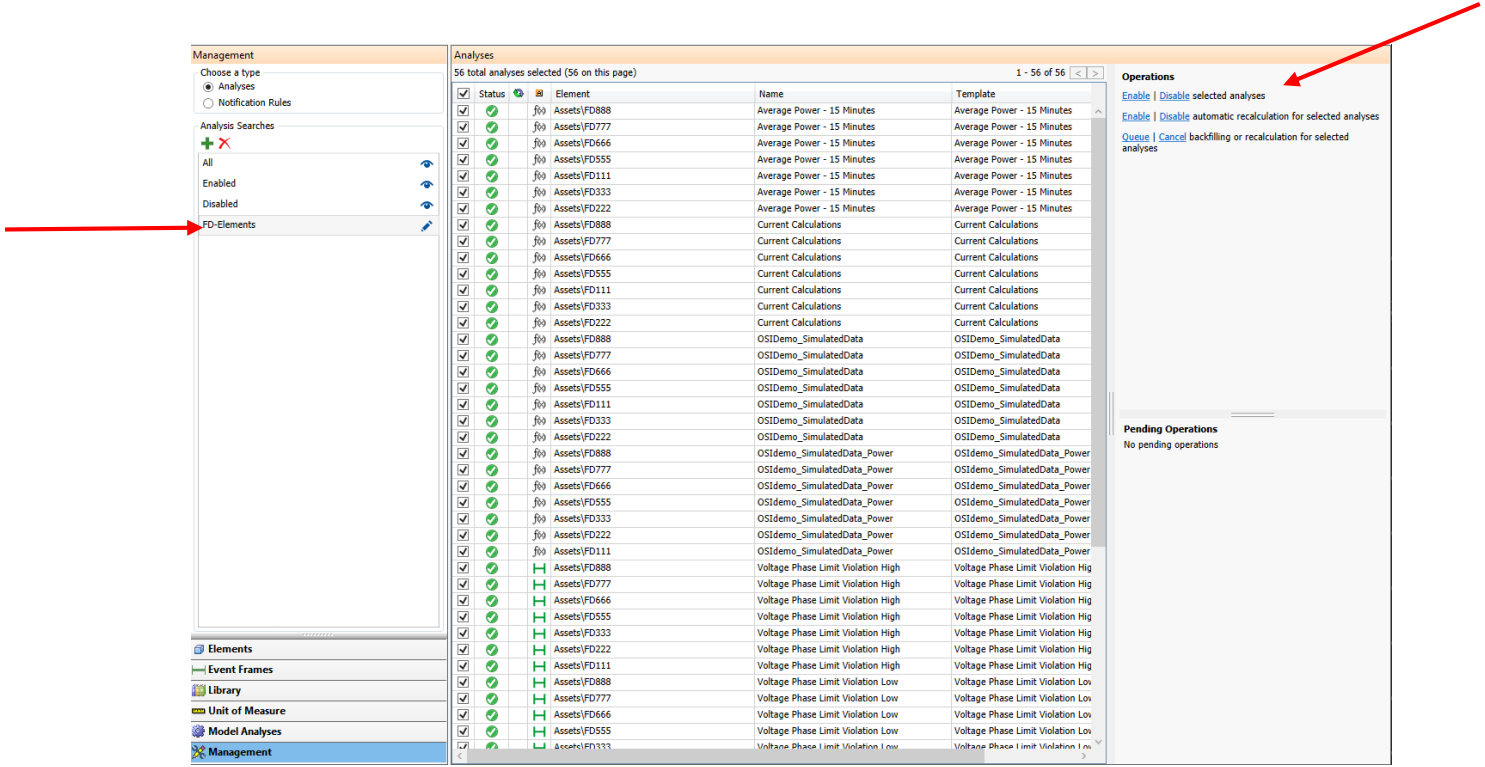


Event Frames and Notifications

- Get notified when an event starts and/or ends.

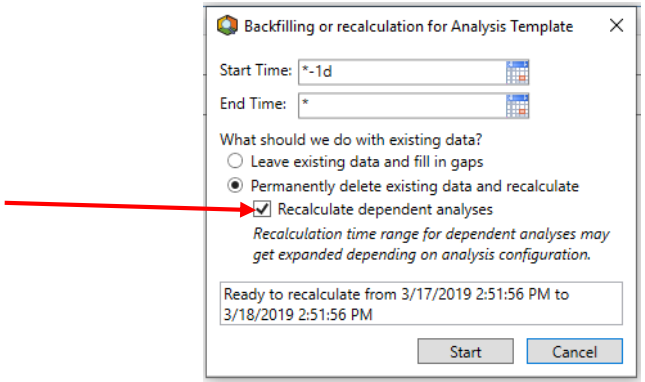


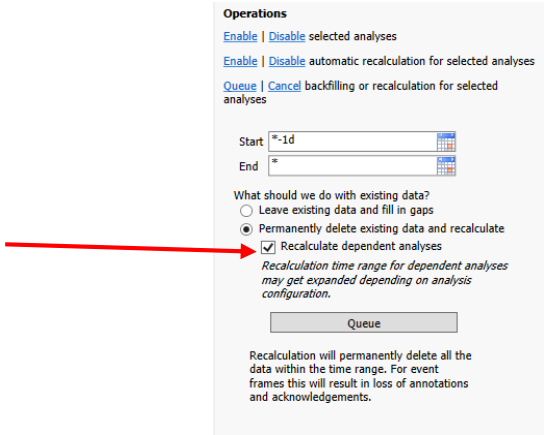
Bulk Operations



Recalculation/Backfilling

- Real-time vs Recalculation: Independent workers.
- Recalculate dependent analyses:





Auto-Recalculation

- Enable auto-recalculation when:
  - Out of order data is expected and
  - It is required to recalculate past data.
- Don't enable auto-recalculation when:
  - Past data shouldn't be modified
  - Past data from dependent calculations shouldn't be modified.

Change Management

- Consider setting up Dev or Test environment
  - Allows for experimentation while developing calculations
  - Many engineers could be writing calculations
  - Frequent changes can cause some churn for the production system
  - Easier to isolate issues

High Availability

- PI Analysis Service supports failover using Windows Server Failover Clustering (WSFC)
- Use buffering for writing PI Point outputs (PI Buffer Subsystem)

More Resources

- Incorporate Event Frames into Your Operations (up next!)
- PI Square (Asset Analytics Best Practices) https://pisquare.osisoft.com

High Availability

- PI Analysis Service supports failover using Windows Server Failover Clustering (WSFC)
- Use buffering for writing PI Point outputs (PI Buffer Subsystem)

Summary

Summary

1. Use variables
2. Use comments
3. Use the Exit() function
4. Save analysis output data to PI Points
5. Build Analytics on a template
6. Avoid expensive, frequently evaluated queries using summary functions such as TagAvg()

Nathaniel Adebayo
Systems Engineer
OSIsoft LLC
nadebayo@osisoft.com

Questions?

Please wait for the **microphone**

State your **name & company**

Please remember