

Limiting Child Event Frames

1. Description

Objective

The objective of this article is to explain how to configure the PI Event Frame connector in the SEEQ administration page to limit root/child condition to a specific hierarchy of event frames, in order to retrieve the conditions from a large number of events.

Context

In this article, we are going to proceed with the example use case of the site Pasadena for Novecare.

Here, the hierarchy refers to the hierarchy of child events for the event type "Procedure", in the section Event Frames of PI System Explorer, as listed in the picture below. Normally, the complete hierarchy contains 4 levels: "UnitProcedure" > "Operation" > "Phase" > "PhaseState".

[blocked URL](#)

In this case, the hierarchy requested by the user is : "UnitProcedure" > "Operation" > "Phase", without the last one "PhaseState".

So the main purpose is to exclude the "PhaseState" conditions for the site Pasadena, through the configuration of Event Frame Connector.

Ajouter un parent / STS-857

- create a hierarchy for the "SignalTables" that would have the same hierarchy as the Excel Pivot table : (OWNER <-- plant) / ST_NAME / SAMPLEPOINT / PA_NAME :
- Create a "SignalTables" to create two signals : SRESULT_PS , SRESULT_NONPS

Could you create to the LIMS connector the following tables (~ 100 products for Pasadena / 100 tables)

- Create in the product hierarchy (OWNER <--plant) / ST_NAME) per product, a "ConditionTables" that would contain all the LIMS table as per Excel.
- Add the 4 columns in the two sample tables : SAMPLEPOINT , PSEUDO , SRESULT_PS, SRESULT_NONPS

Regarding the PAS EF connector,

On the DeV server, could you create a EF connector that would :

- contain the partition condition by asset / product (the field is called "Product Name" for Pasadena)
- limit the created root/child condition to the hierarchy : "UnitProcedure" , "Product Name", "Operation" , "Phase"

2. Knowledge Reference : Indexing Exclusions

In order to realize this configuration, we can use the rule of **Indexing Exclusions** to exclude items from indexing (please see the picture below).

Indexing Exclusions

You can exclude items from indexing by applying a property transform that adds a property named "Exclude From Indexing" with a value of true. Any items that have this property after the transform will be excluded from posts to the Seeq database. The effect is to reduce the total time of indexing and to provide granular control over the indexing of databases that might include items that won't be of interest to end users. If the property is set to any other value than `true` or if it is removed entirely, the next index operation will once again index items that were previously excluded.

```
JS                                                                    Copy
1      "Transforms": [
2      {
3          "Inputs": [
4          {
5              "Property": "AF Path",
6              "Value": ".*\\|Flow_Computed$"
7          }
8          ],
9          "Outputs": [
10         {
11             "Property": "Exclude From Indexing",
12             "Value": true
13         }
14     ]
15 }
16 ]
```

You can go to the link below to check more information :

Seeq Connector Property Transforms

<https://support.seeq.com/kb/R58/cloud/connector-property-transforms#id-%28R58%29ConnectorPropertyTransforms-IndexingExclusions>

In our case, "PhaseState" conditions conclude 5 items to exclude : *Running, Held, Restarting, Holding, Complete*.

[blocked URL](#)

3. EF Connector Configuration

Prerequisites

We need to collect first the following informations regarding the AF database & EF template:

- AF server.
- AF database.
- EF template.

Configuration

Go to the directory `C:\ProgramData\Seeq\data\configuration\link`

- Make a copy of the configuration file (for back-up reason)
- Edit the configuration of the OSIsoft AF Event Frames Connector

[blocked URL](#)

Go to the section "Name": "ACEW1PSTEKPAF02 Event Frames", then set "Enabled" to true.

[blocked URL](#)

Check connector in SEEQ \ ADMINISTRATION TAB:

[blocked URL](#)

Note: Starting with Seeq Server version R54, connector configuration JSON files do not need to be updated manually on the machine running the agent and can instead be managed through the administration page.

In the datasources tab, select the datasource of interest and click "Manage" for the associated connection card:

[blocked URL](#)

Go to the section "Name": "PAS_NOVECARE", then set "Enabled" to true:

[blocked URL](#)

Blocks to add

Now we can add blocks using the **Indexing Exclusions** transform (see *2.Knowledge Reference*).

For the site Pasadena Novecare, as "PhaseState" contains 5 conditions : *Running, Held, Restarting, Holding, Complete*. We add 1 block for each condition.

[blocked URL](#)

Here are the specific blocks to add as transforms :

- **Held**

[blocked URL](#)

- **Running**

[blocked URL](#)

- **Complete**

[blocked URL](#)

- **Holding**

[blocked URL](#)

- **Restarting**

[blocked URL](#)

How to get Data ID

As you can see, in each block we need to specify the Value of Data ID :

[blocked URL](#)

In order to get this Data ID, we need to go to [Seeq](#), open your workbook in the Seeq Workbench Analysis environment, find the exact condition you are looking for, and then click on the "i" [blocked URL](#) icon in front of it.

The Value of Data ID will appear on the left panel.

[blocked URL](#)