Flexible Reports with Data Placement

Overview

Report content is driven by Data Connections. When defining a data connection for a template, two things need to be considered:

- What data source will provide content.
- Where will the content be placed in the report?

This document focuses on where the content will be placed and all the options that are offered by **XLReporter**.

Report as you Run

To understand why data placement is important, it is first important to understand "Report as you Run" technology. This is unique to **XLReporter.**

With **XLReporter**, a report is not a static entity that is generated in one pass. Rather, a report is dynamic and can be built up over time incrementally from many different data sources.

Let's take a simple example of a plant that only has a PLC and a PC, and needs a report with hourly values over the day from the PLC. There is no historian or database to hold this data; it needs to come straight from the PLC into the report every hour over the day.

Most reporting packages would require that data to be collected into some data repository and then, at the end of the day, generate the report with the hourly values collected. With **XLReporter**, that data can be retrieved from the PLC every hour and placed onto the report in the row designated for that hour throughout the entire day. At any time during the day the report can be viewed to see the data up through the previous hour.

Data Retrieval

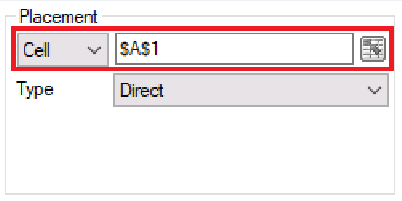
Before analyzing data placement, let's first identify how data is retrieved for reports when configured as data connections.

There are single value connection types like **Variables**, **Expressions** and **Real Time** tags. The data from these are always a single value which is written to a single cell.

Then there are table connections types like **Data Groups** from **Real Time**, **Historical** and **Database** sources. These connections return a table of data that can be both multiple columns and multiple rows.

Data Placement

Cell



Placement

Cell $A$1

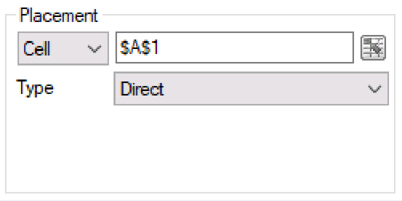
Type Direct

When configuring placement for a data connection, the **Cell** must be defined as a single cell on the worksheet. The simplest definition of **Cell** is that it is where the data is written to on the worksheet. For table connection types, **Cell** is the upper-left corner where data is written.

For example, if the table data connection returns 4 columns and 8 rows and **Cell** is set to *$A$1*, data is written to cell range *$A$1:$D$8*.

Placement Types

Direct



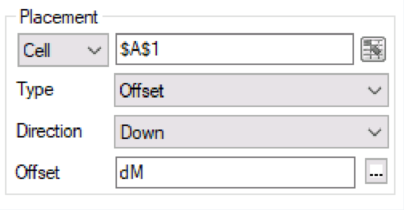
Placement

Cell: A1

Type: Direct

A data connection configured as *Direct* is always written to the **Cell** configured. This is done regardless if the cell already contains data.

Offset



Placement

Cell: A1

Type: Offset

Direction: Down

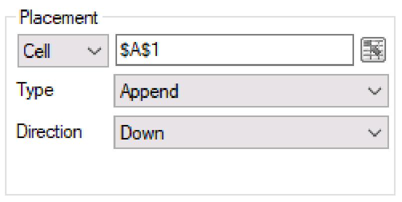
Offset: dM

A data connection configured as *Offset* is written to the sheet offset from the **Cell** configured based on the **Direction** and **Offset** specified.

As a simple example, consider a single value connection with **Cell** set to *A1*, **Direction** set as *Down* and **Offset** set as 2. When updated, the value for the connection is written to cell *A3*; two rows down from cell A1.

In practice, **Offset** is not a static number but rather an **XLReporter** Name Type which is a keyword for an interval of time like *hD* for the hour of the day (0-23) or *dM* for the zero based day of the month (0-30).

Append



Placement

Cell: A1

Type: Append

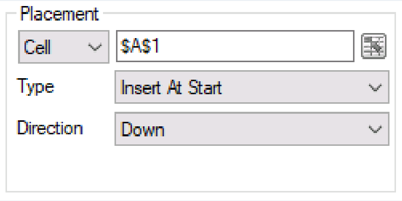
Direction: Down

A data connection configured as *Append* is written to the sheet to the next empty cell starting at **Cell** configured based on the **Direction** specified.

For example, a single value connection configured to **Cell** *A1* with **Direction** as *Down*. Whenever this is updated, the value is written to the next empty cell starting at A1 going down.

This gets a bit more complex with a table connection. Let's take a table connection with 4 columns and 8 rows with **Cell** as *A1*. If **Direction** is *Down*, the values are written to the first row where all 4 columns (A-D) are empty. If **Direction** is *Across*, the values are written to the first column where all 8 rows (1-8) are empty.

Insert At Start/Insert At Start (Full)



Placement

Cell: A1

Type: Insert At Start

Direction: Down

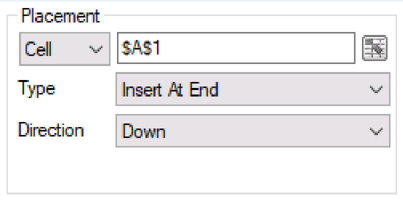
A data connection configured as *Insert At Start* or *Insert At Start (Full)* is written to the sheet by first having the appropriate amount of rows (**Direction** = *Down*) or columns (**Direction** = *Across*) inserted to the worksheet starting at the **Cell** specified and then the data is written into the inserted range.

For example, consider a table data connection that returns 4 columns and 8 rows with **Cell** set to *$A$1*. If **Type** is *Insert At Start* and **Direction** is *Down*, when updated, if *$A$1:$D$1* is empty, a range 7 rows by 4 columns is inserted at cell *$A$2*.

If *$A$1:$D$1* is not empty, a range of 8 rows by 4 columns is inserted at cell *$A$1*. Anything beneath this range in columns *A:D* are pushed downwards. Then, the data is written into the range *$A$1:$D$8*.

If the above example was configured with **Type** as *Insert At Start (Full)*, the same thing would happen, except instead of inserting a range 4 columns wide, 7 or 8 entire rows would be inserted before the data is written.

Insert At End/Insert At End (Full)



Placement

Cell $A$1

Type Insert At End

Direction Down

A data connection configured as *Insert At End* or *Insert At End (Full)* is written to the sheet by first finding the next empty row (**Direction** = *Down*) or column (**Direction** = *Across*), then the appropriate amount of rows (**Direction** = *Down*) or columns (**Direction** = *Across*) is inserted to the worksheet at the empty row or column and finally the data is written into the inserted range.

For example, consider a table data connection that returns 4 columns and 8 rows with **Cell** set to *$A$1*. If **Type** is *Insert At End* and **Direction** is *Down*, when updated, if *$A$1:$D$1* is empty, a range 7 rows by 4 columns is inserted at cell *$A$2*.

If *$A$1:$D$1* is not empty, the next empty row between *A:D* is found and a range 8 rows by 4 columns is inserted at the *A* column of that row. Anything beneath this range in columns *A:D* are pushed downwards. Then, the data is written into the inserted range. If the above example was configured with **Type** as *Insert At End (Full)*, the same thing would happen, except instead of inserting a range 4 columns wide, 7 or 8 entire rows would be inserted before the data is written.

Excel and Inserted Rows, Columns and Ranges

When ranges, rows or columns are inserted into a worksheet, it influences other items within the worksheet. This can be especially useful when the amount of data returned from a data connection varies.

Formatting

When rows or columns are inserted into a worksheet, the formatting of row above or column to the left of the range (depending on which way the cells are shifted) being inserted is applied to the inserted range.

This includes number formatting, borders, fonts, background colors and even extends to any conditional formatting that may be applied.

Charts

If the range for any series of a chart spans over a range in the worksheet that is inserted into, the series range is automatically resized with the number or rows or columns inserted.

For example, if the series of a chart is set up initially as *$A$1:$A$2* and 8 rows are inserted into *$A$2,* the series becomes *$A$1:$A$10*.

Formulas

Formulas behave the same way as chart series in that if the cell reference in the formula spans a range where rows or columns are inserted, the cell reference range is automatically resized with the number of rows or columns inserted.

For example, if a cell contains the initial formula *=SUM(A1:A2)* and 8 rows are inserted into *$A$2,* the formula becomes *=SUM(A1:A10)*.

Scenarios

To understand the flexibility available for data placements, let's consider a series of reporting scenarios and what settings work.

Scheduled Report: Daily Report with Real Time Values at the End of the Day

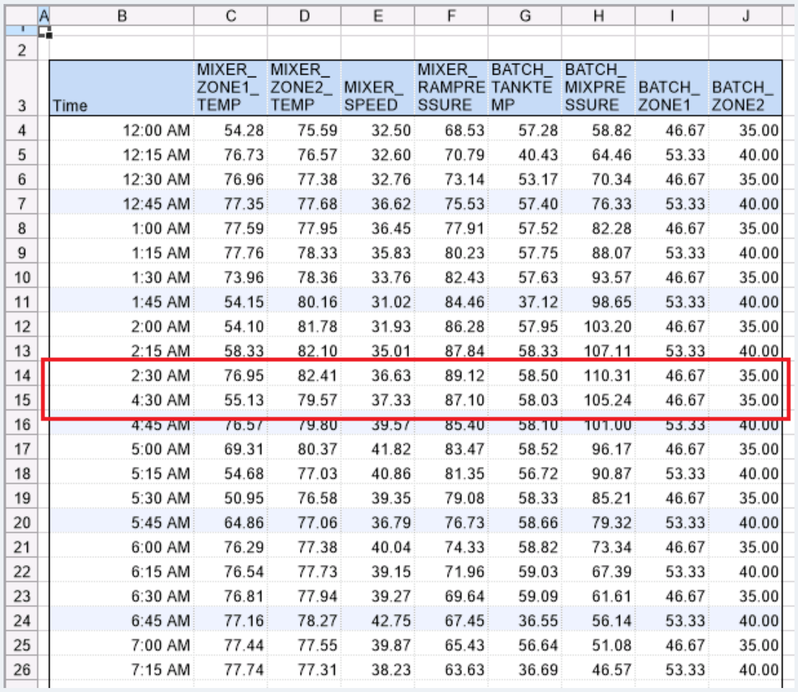
In this scenario, real time data is written to the daily report one time at the end of the day.

Therefore, the **Placement** of the connection should be *Direct*.

Scheduled Report: Daily Report with Real Time Values Every 15 Minutes over the Day

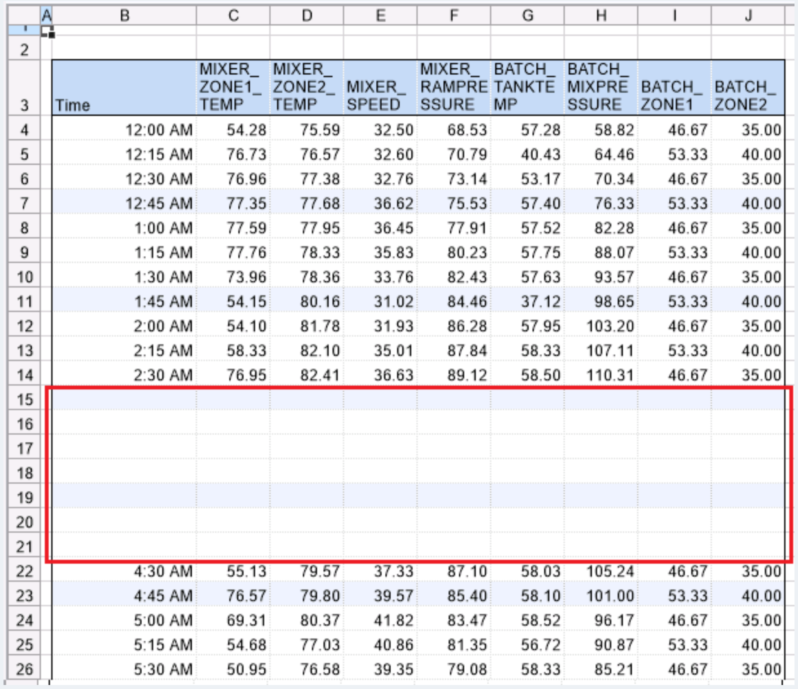
In this scenario, real time data is written to the daily report every 15 minutes throughout the course of the day, down the column starting at midnight. The net result is a report with 96 rows of data.

There are a few options for **Placement** in this scenario. The *Append* or *Insert At End* settings would work. However, if the machine is switched off during the middle of the day and then restarted later, there would be no gap in the report to account for these missing updates.



	A	B	C	D	E	F	G	H	I	J
1										
2										
3		Time	MIXER_ZONE1_TEMP	MIXER_ZONE2_TEMP	MIXER_SPEED	MIXER_RAMPRESSURE	BATCH_TANKTEMP	BATCH_MIXPRESSURE	BATCH_ZONE1	BATCH_ZONE2
4		12:00 AM	54.28	75.59	32.50	68.53	57.28	58.82	46.67	35.00
5		12:15 AM	76.73	76.57	32.60	70.79	40.43	64.46	53.33	40.00
6		12:30 AM	76.96	77.38	32.76	73.14	53.17	70.34	46.67	35.00
7		12:45 AM	77.35	77.68	36.62	75.53	57.40	76.33	53.33	40.00
8		1:00 AM	77.59	77.95	36.45	77.91	57.52	82.28	46.67	35.00
9		1:15 AM	77.76	78.33	35.83	80.23	57.75	88.07	53.33	40.00
10		1:30 AM	73.96	78.36	33.76	82.43	57.63	93.57	46.67	35.00
11		1:45 AM	54.15	80.16	31.02	84.46	37.12	98.65	53.33	40.00
12		2:00 AM	54.10	81.78	31.93	86.28	57.95	103.20	46.67	35.00
13		2:15 AM	58.33	82.10	35.01	87.84	58.33	107.11	53.33	40.00
14		2:30 AM	76.95	82.41	36.63	89.12	58.50	110.31	46.67	35.00
15		4:30 AM	55.13	79.57	37.33	87.10	58.03	105.24	46.67	35.00
16		4:45 AM	76.57	79.80	39.57	85.40	58.10	101.00	53.33	40.00
17		5:00 AM	69.31	80.37	41.82	83.47	58.52	96.17	46.67	35.00
18		5:15 AM	54.68	77.03	40.86	81.35	56.72	90.87	53.33	40.00
19		5:30 AM	50.95	76.58	39.35	79.08	58.33	85.21	46.67	35.00
20		5:45 AM	64.86	77.06	36.79	76.73	58.66	79.32	53.33	40.00
21		6:00 AM	76.29	77.38	40.04	74.33	58.82	73.34	46.67	35.00
22		6:15 AM	76.54	77.73	39.15	71.96	59.03	67.39	53.33	40.00
23		6:30 AM	76.81	77.94	39.27	69.64	59.09	61.61	46.67	35.00
24		6:45 AM	77.16	78.27	42.75	67.45	36.55	56.14	53.33	40.00
25		7:00 AM	77.44	77.55	39.87	65.43	56.64	51.08	46.67	35.00
26		7:15 AM	77.74	77.31	38.23	63.63	36.69	46.57	53.33	40.00

To account for this, the **Placement** should be set to *Offset* with the **Offset** set to *mD/15*. Now, if the machine is switched off and restarted later, there is a clear gap in the report, so the missing time is obvious.



	A	B	C	D	E	F	G	H	I	J
1										
2										
3		Time	MIXER_ ZONE1_ TEMP	MIXER_ ZONE2_ TEMP	MIXER_ SPEED	MIXER_ RAMPRE SSURE	BATCH_ TANKTE MP	BATCH_ MIXPRE SSURE	BATCH_ ZONE1	BATCH_ ZONE2
4		12:00 AM	54.28	75.59	32.50	68.53	57.28	58.82	46.67	35.00
5		12:15 AM	76.73	76.57	32.60	70.79	40.43	64.46	53.33	40.00
6		12:30 AM	76.96	77.38	32.76	73.14	53.17	70.34	46.67	35.00
7		12:45 AM	77.35	77.68	36.62	75.53	57.40	76.33	53.33	40.00
8		1:00 AM	77.59	77.95	36.45	77.91	57.52	82.28	46.67	35.00
9		1:15 AM	77.76	78.33	35.83	80.23	57.75	88.07	53.33	40.00
10		1:30 AM	73.96	78.36	33.76	82.43	57.63	93.57	46.67	35.00
11		1:45 AM	54.15	80.16	31.02	84.46	37.12	98.65	53.33	40.00
12		2:00 AM	54.10	81.78	31.93	86.28	57.95	103.20	46.67	35.00
13		2:15 AM	58.33	82.10	35.01	87.84	58.33	107.11	53.33	40.00
14		2:30 AM	76.95	82.41	36.63	89.12	58.50	110.31	46.67	35.00
15										
16										
17										
18										
19										
20										
21										
22		4:30 AM	55.13	79.57	37.33	87.10	58.03	105.24	46.67	35.00
23		4:45 AM	76.57	79.80	39.57	85.40	58.10	101.00	53.33	40.00
24		5:00 AM	69.31	80.37	41.82	83.47	58.52	96.17	46.67	35.00
25		5:15 AM	54.68	77.03	40.86	81.35	56.72	90.87	53.33	40.00
26		5:30 AM	50.95	76.58	39.35	79.08	58.33	85.21	46.67	35.00

Since the amount of data is known, any formatting, formulas or charts can be configured for the 96 rows of data in the template.

Scheduled Report: Daily Report with Real Time Values Every Time an Event Occurs

In this scenario, real time data is written to the daily report any time an event occurs throughout the course of the day, down the column.

There are a few options for **Placement** in this scenario. The *Append* setting would work. However, any formatting, formulas or charts in the report would not be automatically adjusted every time data is written to the report. While there are management functions that can make these adjustments, it would be easier if this were done automatically.

The *Insert At Start* or *Insert At End* **Placement** works best here because it will carry any formatting with it as well as adjust any chart and formula ranges.

If the data for the report should be listed in ascending order (oldest to newest) use *Insert At End*, whereas for descending order (newest to oldest), use *Insert At Start*.

Scheduled Report: Daily Report with History Values for Every Hour of the Day

In this scenario, history data is written to the daily report at the end of the day to bring in values for every hour of the day.

Since every hour of the day can be retrieved at one time, the **Placement** of the connection should be *Direct*.

Scheduled Report: Monthly Report with History Values for Every Four Hours of the Month

In this scenario, history data is written to the monthly report to show every four hours over the day.

Just like in the previous scenario, all the history data for the month can be retrieved at one time at the end of the month to produce the report.

However, there are some customers who would like to see the report being built up during the month, up to the previous day.

To accomplish this, the **Placement** can be set to *Offset* and the **Offset** set to *dM*6*. *dM* is the day of the month, zero based (0-30). The *6 multiplier compensates for the six values brought in for the day (every four hours).

So when updated on the first of the month, the **Offset** evaluates to *0*. On the second of the month it evaluates to *6*, the third to *12* and so on. If an update is missed, the six rows for that day are left blank in the report.

On Demand Report: History Values Over Time Period and Interval Specified by the User

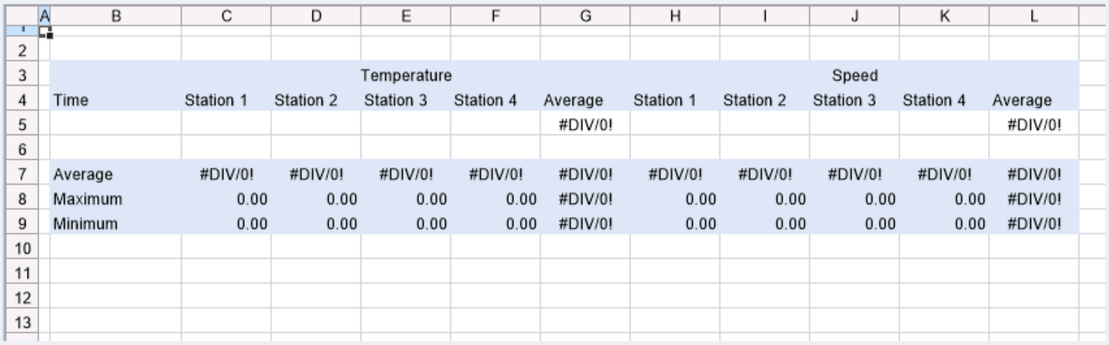
In this scenario, history data is written to the report based on the start time, end time and interval specified by the user and generated on demand based on these settings.

The **Placement** for the connection could be set *Direct* since all the data is returned in a single update. However, since the amount of data in the report is completely unpredictable, any charts, formulas or even formatting cannot be applied in the template, it must be done at runtime.

For formatting, the argument could be made to just format the entire column for the formatting desired. While that does work, every cell that is formatted makes the workbook file larger and one column is more than 1 million cells. The larger the file, the longer it takes to open, save and close, so keeping a tight rein on the file size is a good thing to do.

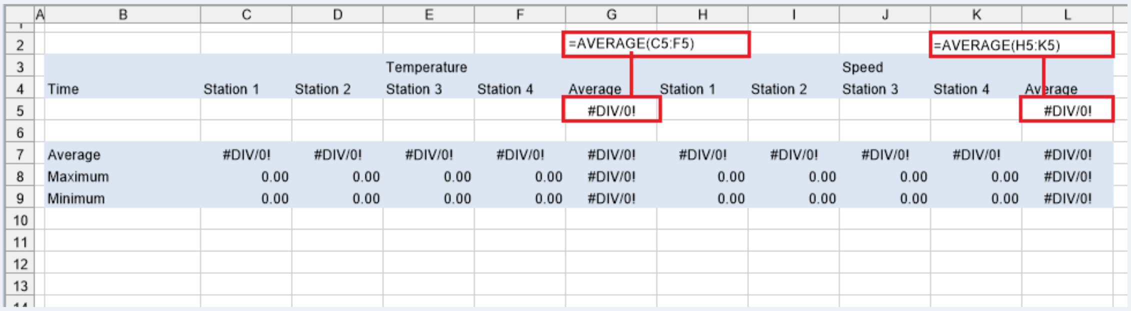
The *Insert At End* **Placement** works best for this scenario. Not only does it apply the formatting down to all the rows of data, any summary formulas and charts are updated automatically.

In addition, if there are row calculations required as part of the report, the formulas are filled downward with the data. For example, let's take a report that shows hourly temperatures and speeds for four different stations in a plant. For every hour, the report should also show the average temperature and average speed over all four stations.



	A	B	C	D	E	F	G	H	I	J	K	L
1												
2												
3					Temperature					Speed		
4		Time	Station 1	Station 2	Station 3	Station 4	Average	Station 1	Station 2	Station 3	Station 4	Average
5							#DIV/0!					#DIV/0!
6												
7		Average	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!
8		Maximum	0.00	0.00	0.00	0.00	#DIV/0!	0.00	0.00	0.00	0.00	#DIV/0!
9		Minimum	0.00	0.00	0.00	0.00	#DIV/0!	0.00	0.00	0.00	0.00	#DIV/0!
10												
11												
12												
13												

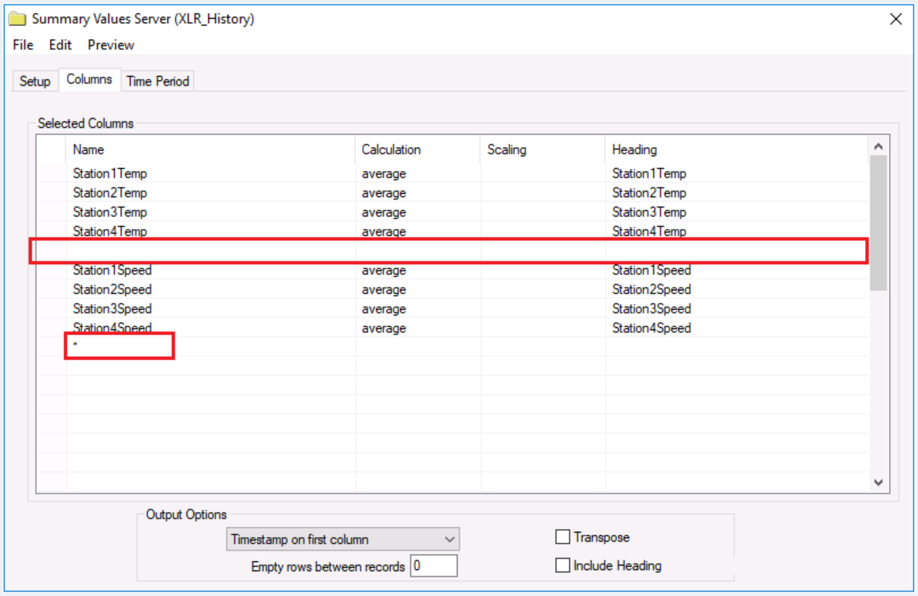
The average of the temperatures and speeds for every hour can be done using simple formulas.



	A	B	C	D	E	F	G	H	I	J	K	L
2							=AVERAGE(C5:F5)				=AVERAGE(H5:K5)	
3					Temperature					Speed		
4		Time	Station 1	Station 2	Station 3	Station 4	Average	Station 1	Station 2	Station 3	Station 4	Average
5							#DIV/0!					#DIV/0!
6												
7		Average	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!	#DIV/0!
8		Maximum	0.00	0.00	0.00	0.00	#DIV/0!	0.00	0.00	0.00	0.00	#DIV/0!
9		Minimum	0.00	0.00	0.00	0.00	#DIV/0!	0.00	0.00	0.00	0.00	#DIV/0!
10												
11												
12												
13												

When the *Insert At End* placement is applied, any formulas within range being inserted are automatically filled down for the number of rows inserted.

In the **History Data Group**, under the **Columns** tab,



Name	Calculation	Scaling	Heading
Station1Temp	average		Station1Temp
Station2Temp	average		Station2Temp
Station3Temp	average		Station3Temp
Station4Temp	average		Station4Temp
Station1Speed	average		Station1Speed
Station2Speed	average		Station2Speed
Station3Speed	average		Station3Speed
Station4Speed	average		Station4Speed
*			

Timestamp on first column

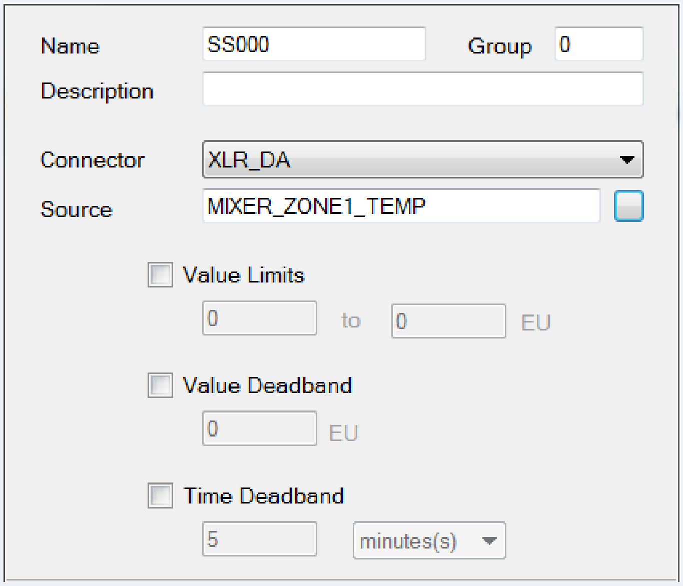
Empty rows between records 0

□ Transpose

□ Include Heading

The column for the average temperatures is accounted for by leaving an empty row. The average speed, however, is outside the range of the group. To account for this, in the row beneath the speed for the forth station, the **Name** is set as *. This acts like a placeholder to extend the group out by one column to account for the average speed formula.

Snapshot



Name SS000 Group 0

Description

Connector XLR_DA

Source MIXER_ZONE1_TEMP

□ Value Limits

0 to 0 EU

□ Value Deadband

0 EU

□ Time Deadband

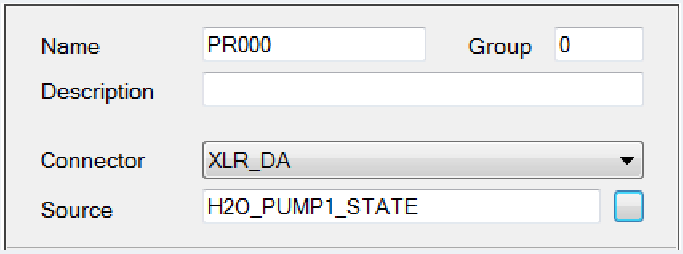
5 minutes(s)

Snapshot is the simplest analytic available. When triggered, the process value specified as **Source** is collected and stored so it is available for a report.

As part of the configuration, the following options can be configured:

- **Value Limits**
  If enabled, the value is only stored if it is within the limits specified.
- **Value Deadband**
  If enabled, the value is only stored if the difference between it and the previous value is at least the deadband value specified.
- **Time Deadband**
  If enabled, the value is only stored if the amount of time since the last value was stored is at least the deadband time specified.

Profile



Name PR000 Group 0

Description

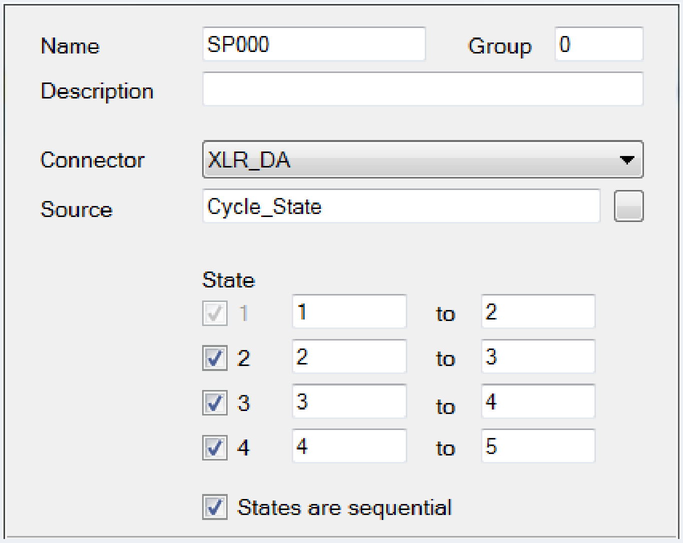
Connector XLR_DA

Source H2O_PUMP1_STATE

Profile is an analytic designed for digital contacts in the process. It is used to determine KPI's like the number of starts and stops as well as the amount of uptime and downtime.

The **Source** setting should be set to a digital contact in the process.

State Profile



Name	SP000	Group	0
Description			
Connector	XLR_DA		
Source	Cycle_State		

State

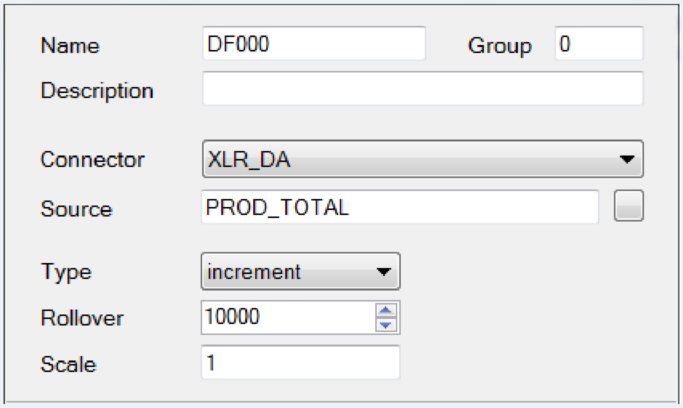
☑ 1	1	to	2
☑ 2	2	to	3
☑ 3	3	to	4
☑ 4	4	to	5

☑ States are sequential

State profile is an analytic designed for analog values in the process where each value represents a specific **State** of the process. It can be used to determine KPI's like the number of times the state was entered and how long it was in each **State**. Up to four unique **State** values can be defined.

If the **Source** value cycles through each **State** sequentially, check the **States are sequential** option, otherwise this should be left unchecked.

Difference



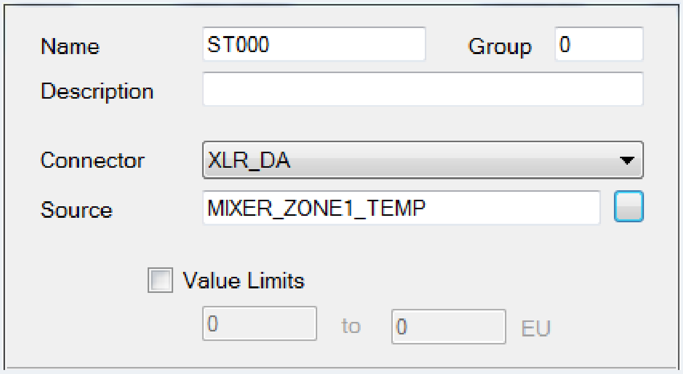
Name	DF000	Group	0
Description			
Connector	XLR_DA		
Source	PROD_TOTAL		
Type	increment		
Rollover	10000		
Scale	1		

Difference is an analytic designed for analog totalizers to determine things like the amount produced over a certain time period. For example, a totalizer may be configured to start from *0* and reset when it reaches *10,000*. However, a report may need to know how much was produced over the day. To determine this, the totalizer value at the beginning of the day should be subtracted from the value at the end of the day.

The **Type** setting defines the behavior of the totalizer, e.g., does it *increment* or *decrement*.

The **Rollover** setting defines at what value the totalizer rolls over, e.g., resets. This is factored into the difference calculation when needed. If the value does not roll over, set this to *0*.

Statistic



Name ST000 Group 0

Description

Connector XLR_DA

Source MIXER_ZONE1_TEMP

□ Value Limits

0 to 0 EU

Statistic analytics are designed to produce a suite of statistical calculations including:

- Total
- Maximum value
- Maximum time
- Minimum time
- Range
- Average
- Standard Deviation

Value Limits can be used to restrict the process values that contribute to the statistic calculations to those that fall between the values specified.

Analytic Schedule

Every configured analytic can be scheduled under the **Schedule** tab. There are two actions typically required for **Update** and **Store**.

Update Actions

The **Update** actions define when the data is read from the process to contribute to the analytic calculations.

Store Actions

The **Store** actions define when the analytic calculations are completed, stored, and made available for reporting.

Groups

Each analytic can be configured with a **Group** number. By default, this is set to *0*. This can be set to a specific number so that when scheduled, the analytics in a specific group can be updated or stored without affecting other configured analytics of the same type.

Group *0* is considered the global group. This means that any analytic configured as *0* is updated or stored regardless of the group number specified with the **Update Group** or **Store Group action**.

Specifics

Snapshot

The **Snaphot** analytics do not require an **Update** action to be configured, only a **Store** action.

Difference

The **Update** and **Store** actions for **Difference** analytics should be configured at the same time or on the same event with the **Update** listed above the **Store** in the list.

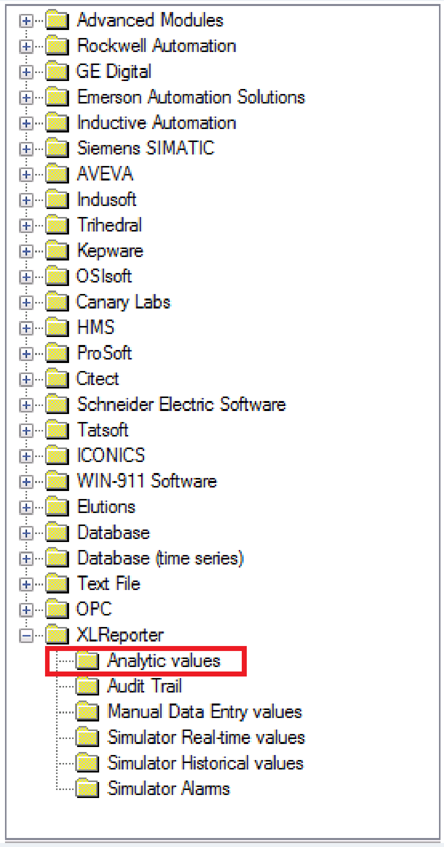
Analytic Database

As previously mentioned, analytics can be stored to a relational database.

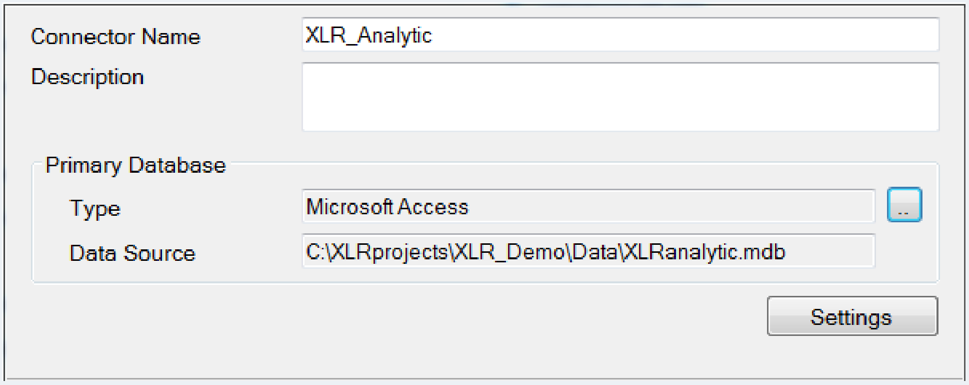
Connector

The first step in configuring analytics to store to a database is to create a connector in the project to define the database to store to.

Connectors can be added to the project in the **Project Explorer** under the **Data** tab by clicking **Connectors**. The **Add** button is provided to add a new connector.



Under the **XLReporter** folder select **Analytic values** for the connector type.



Connector Name: XLR_Analytic

Description:

Primary Database

Type: Microsoft Access

Data Source: C:\XLRprojects\XLR_Demo\Data\XLRanalytic.mdb

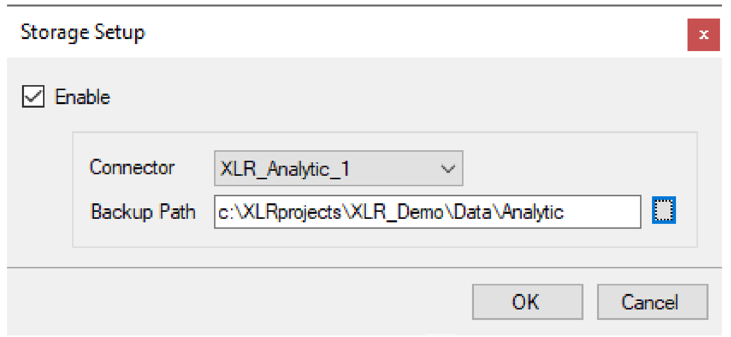
The database is defined by clicking the browse pushbutton [...] for **Primary Database**. There are options for Microsoft SQL Server and MySQL. If these are not available, a blank Microsoft Access database is always installed with the project. Browse to the *Data* folder of the project and select *XLRanalytic.mdb*.

Once configuration is complete, on **OK**, a set of tables is created in the database to hold the analytic data.

Analytic Designer

Once the **Analytic values** connector is configured, analytics are automatically set up to store to the database specified.

To verify this, in the **Analytic Designer,** click the **Storage** menu at the top.



Storage Setup

☑ Enable

Connector: XLR_Analytic_1

Backup Path: c:\XLRprojects\XLR_Demo\Data\Analytic

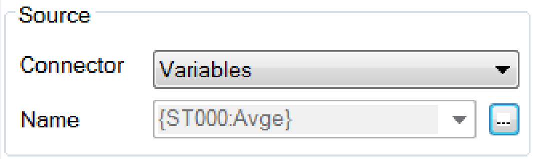
Notice that **Enable** is checked and the **Connector** set up is selected.

The **Backup Path** is provided so that if for any reason the database is unavailable, data is temporarily stored in this path. Once the database is available again the data is written to the database and the path is cleared out. This is automatically set up to the *Data\Analytic* folder in the current project but can be changed if needed.

Reporting on Analytics

When designing a template there are two ways to configure data connections to analytics.

Real Time Analytics

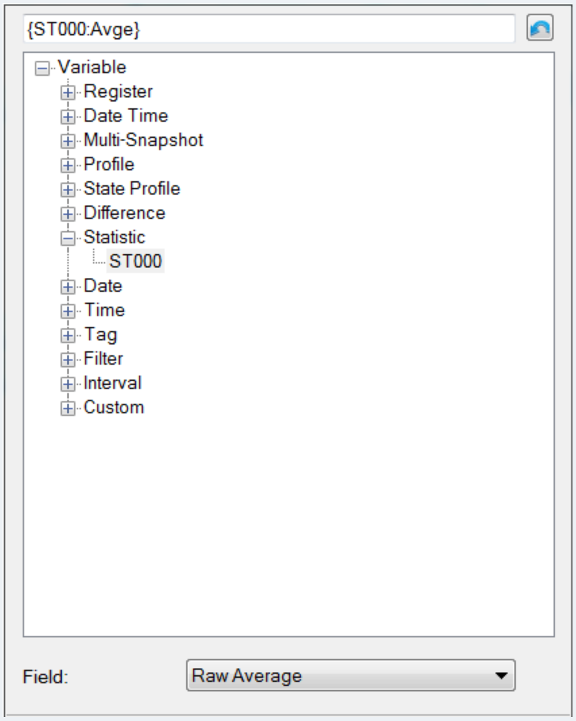


Source

Connector: Variables

Name: {ST000:Avge}

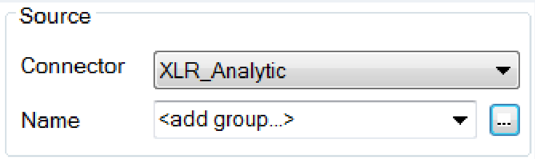
For real time analytics, under the **Source** section of the connection set **Connector** to *Variables*.



{ST000:Avge}

Field: Raw Average

Analytic Data Group

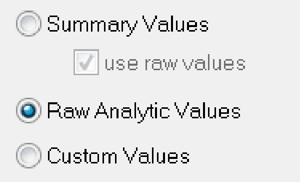


Source

Connector: XLR_Analytic

Name: <add group...>

Analytic Data Groups are designed to retrieve the analytic data stored in the database. To configure a group, under the **Source** section set **Connector** to *XLR_Analytic* (the name of the analytic connector configured). For **Name**, with <*add group*> set, click the browse pushbutton […].



○ Summary Values

☑ use raw values

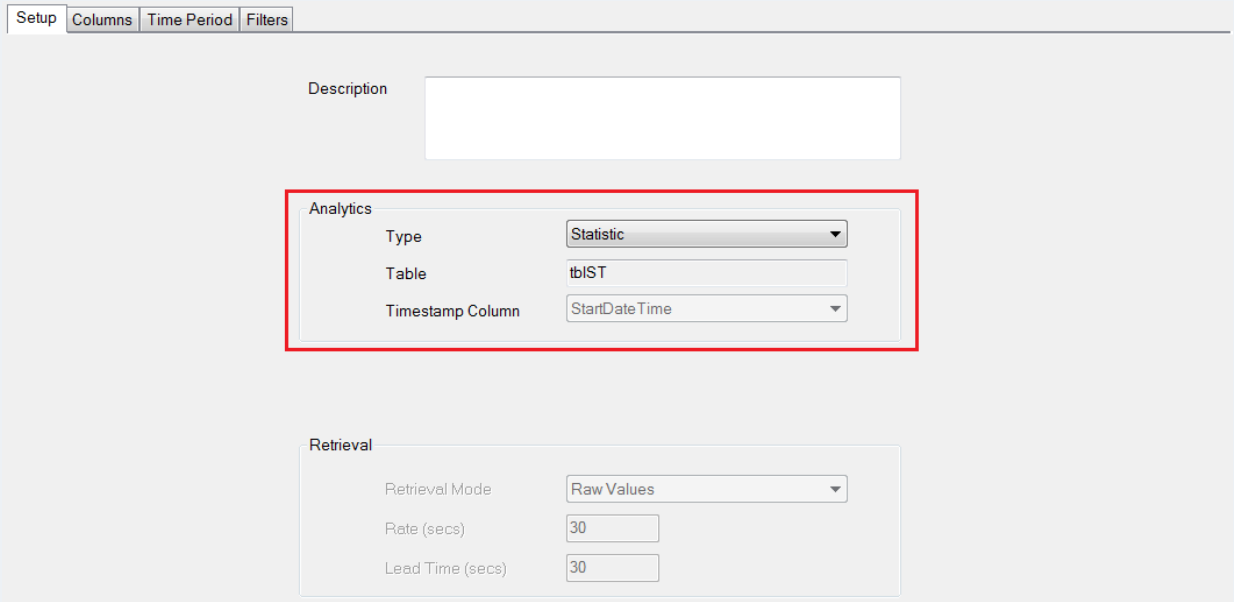
● Raw Analytic Values

○ Custom Values

The following group types are available:

- **Summary Values**
  Summary values calculate summaries like averages, minimums and maximums from analytic data logged to the database.
  For example, summary calculations can be performed on Snapshot values. However, these calculations can be done on any field of any analytic stored. For example, the total on count of a Profile analytic over the day or the average of the daily Statistic minimum value. The possibilities are endless.
- **Raw Analytic Values**
  Raw analytic values return the values recorded in the analytic database over the time period specified.
- **Custom Values**
  Custom values opens the **Database Group Builder** to construct a custom query against the analytic database.

For the purposes of this, we will focus on the **Raw Analytic Values** group type.



Description

Analytics

Type: Statistic

Table: tblST

Timestamp Column: StartDateTime

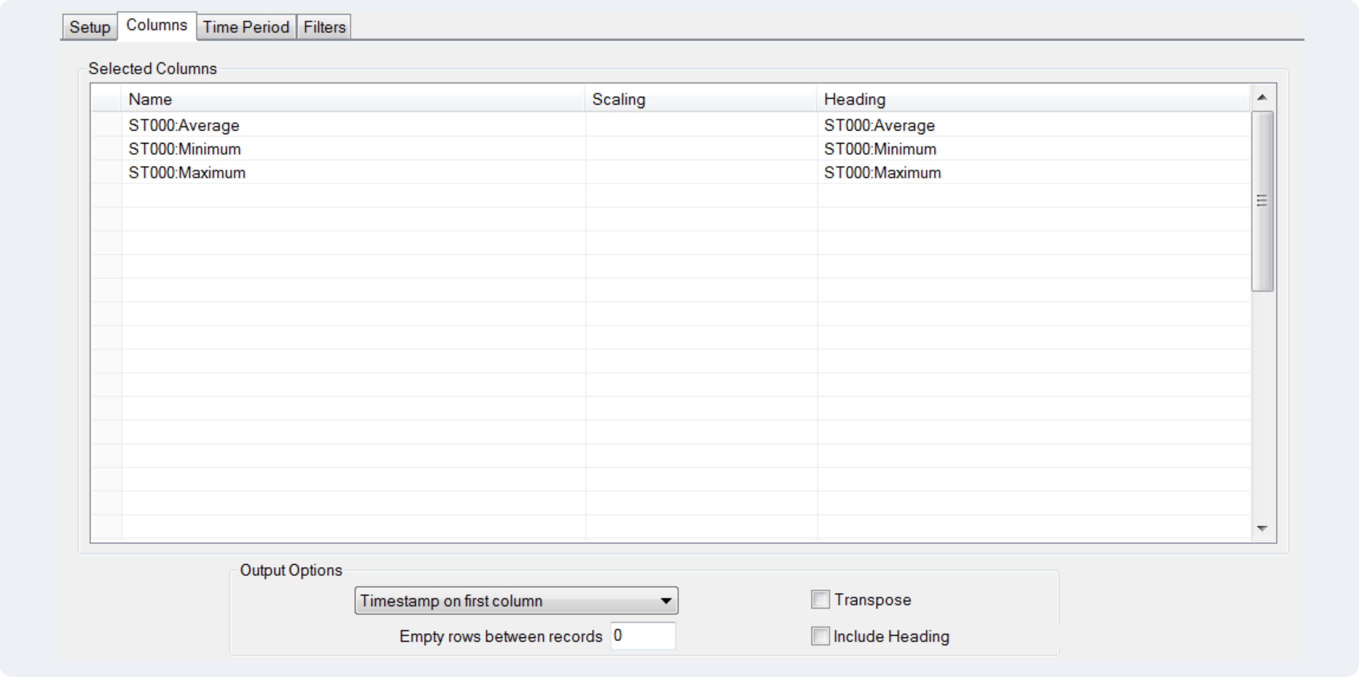
Retrieval

Retrieval Mode: Raw Values

Rate (secs): 30

Lead Time (secs): 30

Under the **Setup** tab, in the **Analytics** section, set the **Type** to the type of Analytic to retrieve data from.



Selected Columns

Name	Scaling	Heading
ST000:Average		ST000:Average
ST000:Minimum		ST000:Minimum
ST000:Maximum		ST000:Maximum

Output Options

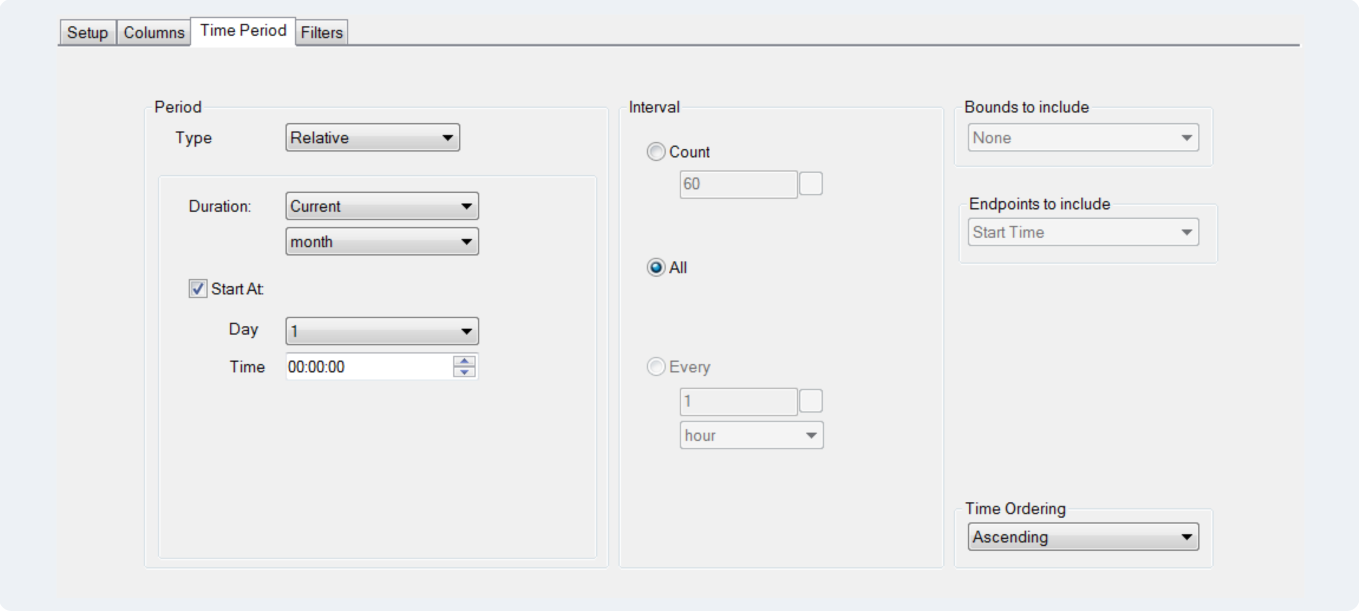
Timestamp on first column

Empty rows between records 0

□ Transpose

□ Include Heading

Under the **Columns** tab, each specific analytic (based on the **Type** specified under the **Setup** tab) can be specified to return.



Period

Type Relative

Duration: Current

month

☑ Start At

Day 1

Time 00:00:00

Interval

○ Count

60

● All

○ Every

1

hour

Bounds to include

None

Endpoints to include

Start Time

Time Ordering

Ascending

Under the **Time Period** tab, the overall time period for the group is defined.

Scenarios

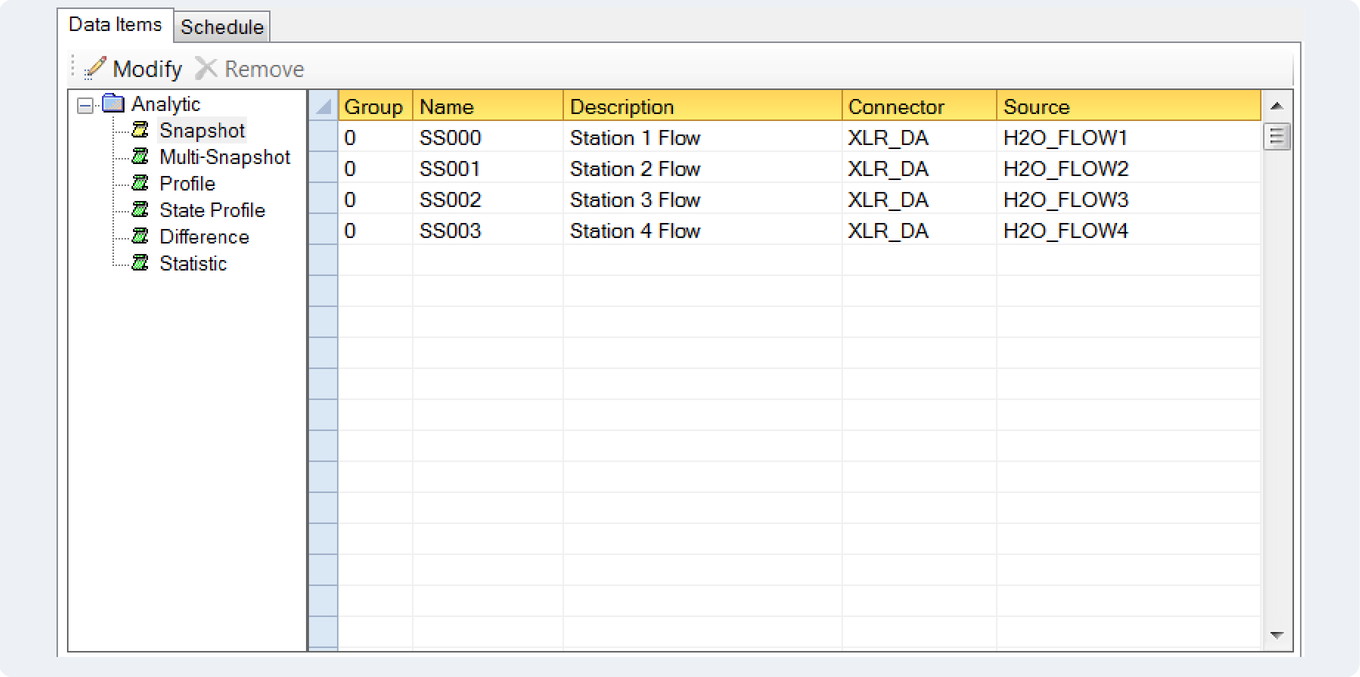
The following presents a real-life scenario where analytics are used for the reporting solution. In this scenario, analytics are configured to store to a database.

Multiple Reports with Values

In this scenario there are two reports required for each day. The first report needs to display 15-minute process values over the day. The second report needs to display those same process values but every 4 hours over the day.

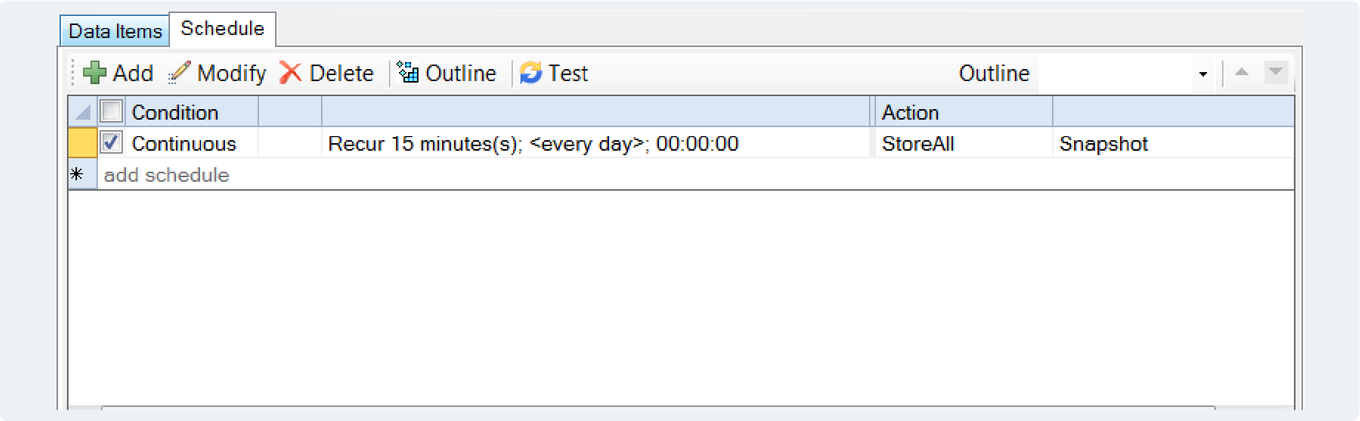
While both could be configured with real time connections to pull this data, since both reports require the same process values, this data could be stored using **Snapshot** analytics and then retrieved at the end of the day for both reports.

Analytic Designer



Group	Name	Description	Connector	Source
0	SS000	Station 1 Flow	XLR_DA	H2O_FLOW1
0	SS001	Station 2 Flow	XLR_DA	H2O_FLOW2
0	SS002	Station 3 Flow	XLR_DA	H2O_FLOW3
0	SS003	Station 4 Flow	XLR_DA	H2O_FLOW4

For **Snapshot**, four snapshots are configured to collect the four process values needed for both reports.

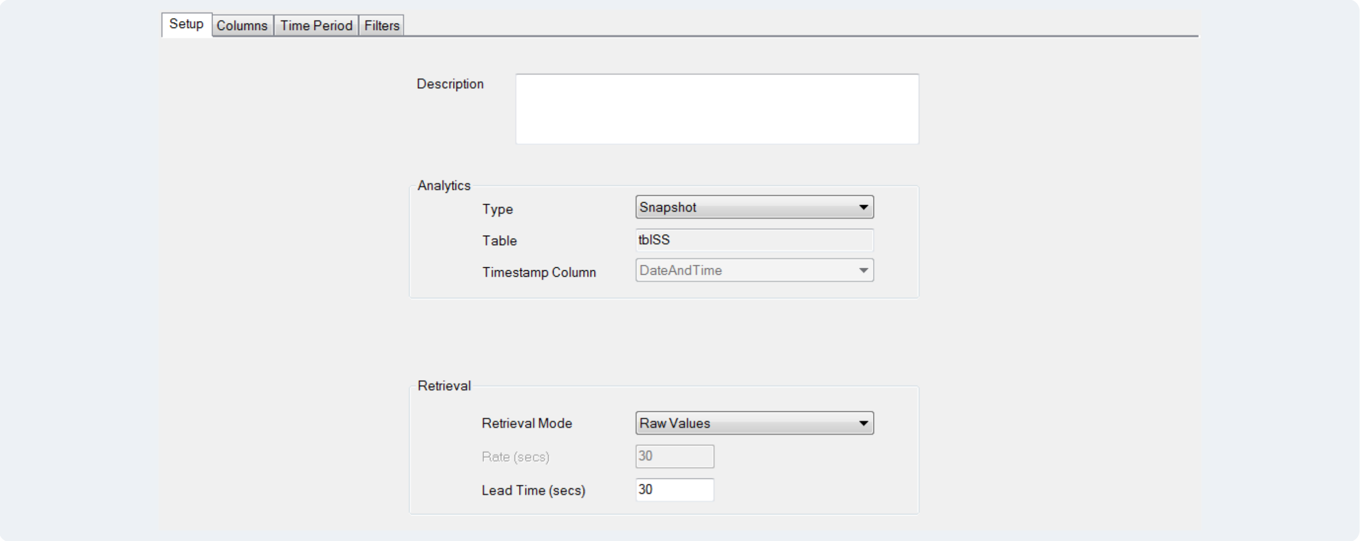


□ Condition			Action	
☑ Continuous		Recur 15 minutes(s); <every day>; 00:00:00	StoreAll	Snapshot
add schedule				

Under **Schedule**, the **Snapshot** analytics are scheduled to store every *15 minutes*, the faster of the data requirements.

Data Groups

For the report that requires 15-minute data over the day, a **Summary Values** data group is configured.



Description

Analytics

Type: Snapshot

Table: tblSS

Timestamp Column: DateAndTime

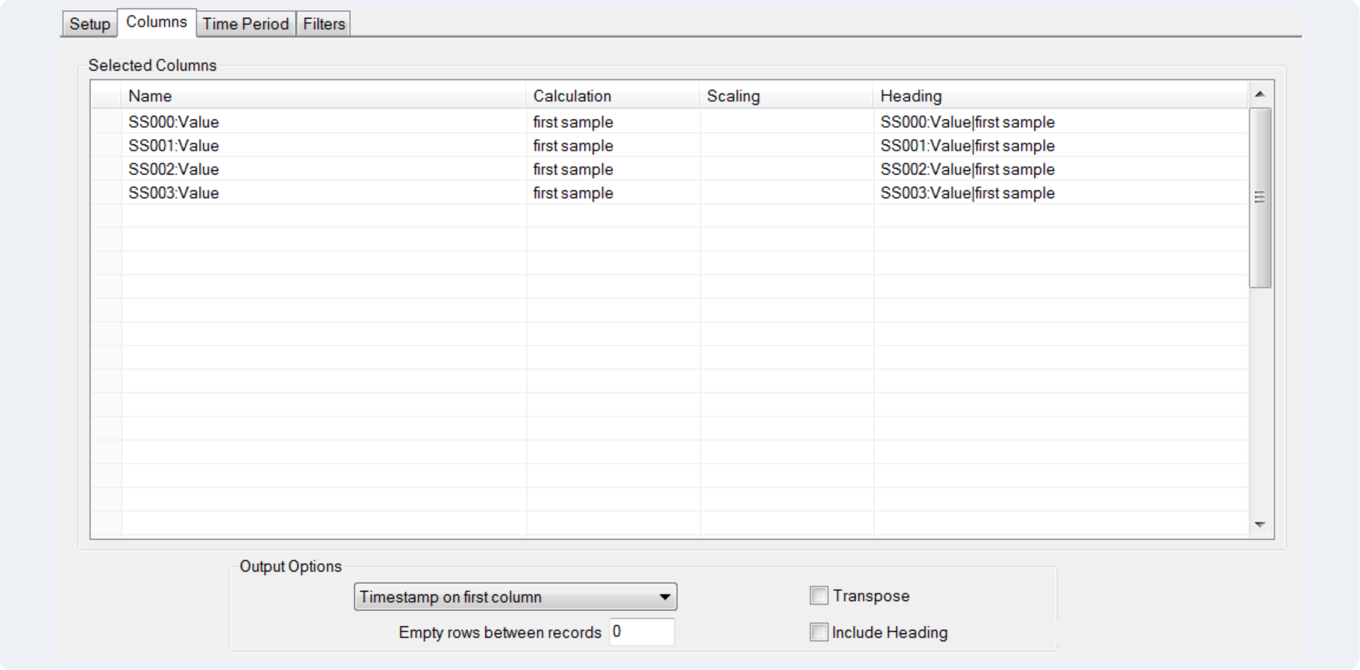
Retrieval

Retrieval Mode: Raw Values

Rate (secs): 30

Lead Time (secs): 30

Under the **Setup** tab, the **Analytics Type** is set to **Snapshot**.



Selected Columns

Name	Calculation	Scaling	Heading
SS000:Value	first sample		SS000:Value|first sample
SS001:Value	first sample		SS001:Value|first sample
SS002:Value	first sample		SS002:Value|first sample
SS003:Value	first sample		SS003:Value|first sample

Output Options

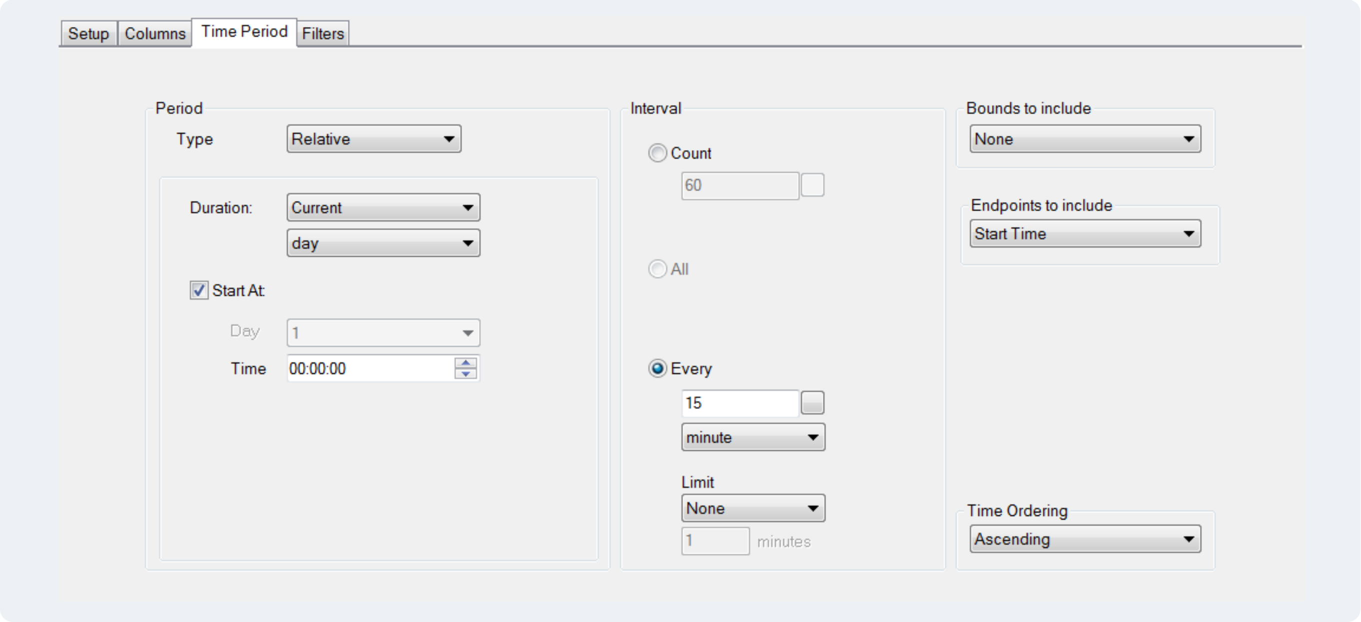
Timestamp on first column

Empty rows between records: 0

□ Transpose

□ Include Heading

Under the **Columns** tab, all four snapshots are selected. The **Calculation** is set to *first sample* to return the recorded sample.



Period

Type: Relative

Duration: Current

day

☑ Start At

Day: 1

Time: 00:00:00

Interval

○ Count

60

○ All

● Every

15

minute

Limit: None

1 minutes

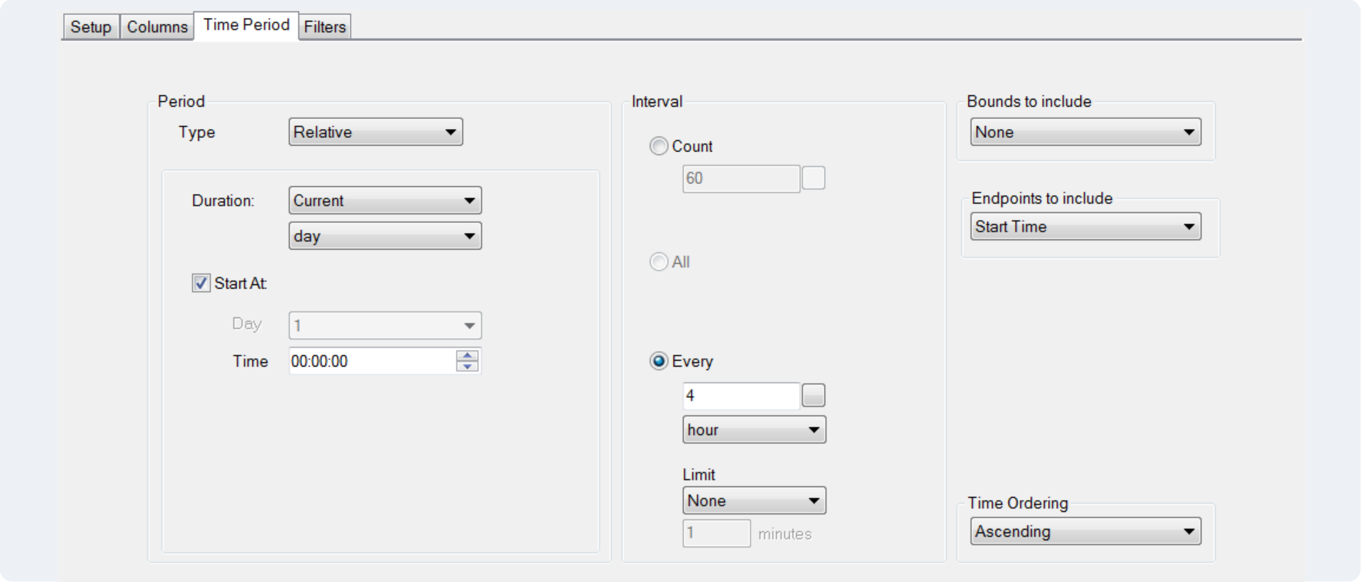
Bounds to include: None

Endpoints to include: Start Time

Time Ordering: Ascending

Under the **Time Period** tab, the **Period** is set up for the *Current day*. The **Interval** is set to *15 minutes* to retrieve 15-minute samples over the day.

The data group for the four hour report the data group is almost identical to the first one.



Period

Type: Relative

Duration: Current

day

☑ Start At

Day: 1

Time: 00:00:00

Interval

○ Count

60

○ All

● Every

4

hour

Limit: None

1 minutes

Bounds to include: None

Endpoints to include: Start Time

Time Ordering: Ascending

The only difference is under the **Time Period** tab the **Interval** is set to *4 hours* instead of *15 minutes*.