

Analytics

Overview

Analytics is one of the most powerful features of **XLReporter**. It can produce a set of analytical calculations on values in the process. There is a suite of analytic types available to do everything from statistics like average, minimum, and maximum, to analytic types that can derive things like uptime and downtime of different machines in the plant.

The following document highlights what analytics are available, how to configure them and some real-world examples of their usage.

Understanding Analytics

Analytics is not a replacement to a historian. Raw values are not collected and stored so they can be retrieved later on. Rather, analytics are configured to read values from the process periodically which contribute to a set of calculations. At the end of a time period (or on a specific event) the calculations are completed and made available for reporting.

As a simple example, consider a temperature value available in the process. An analytic can be configured to read this value every minute to calculate the average temperature and then, at the end of the day, make this average value available for reporting. Once this is made available, internally a new average is started which is completed at the end of the next day.

There are two ways to bring analytic data into a report.

Real Time Analytics

Real time analytics are the last set of calculations made available.

Analytic Database

Analytic calculations can be stored to a database so that multiple records can be retrieved for reporting in a single request. Note, the collected process values are not stored, only the analytic calculations.

Analytic Setup

Analytics are configured in the **Project Explorer** under the **Data** tab by selecting **Analytics**.

Each analytic type has its own branch on the left side of the **Analytic Designer** under the **Data Items** tab. The following analytic types are available:

Snapshot

Name	SS000	Group	0
Description			
Connector	XLR_DA		
Source	MIXER_ZONE1_TEMP	<input type="checkbox"/>	
<input type="checkbox"/> Value Limits	0	to	0 EU
<input type="checkbox"/> Value Deadband	0	EU	
<input type="checkbox"/> 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	<input type="checkbox"/>	

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	<input type="checkbox"/>	
State			
<input checked="" type="checkbox"/> 1	1	to	2
<input checked="" type="checkbox"/> 2	2	to	3
<input checked="" type="checkbox"/> 3	3	to	4
<input checked="" type="checkbox"/> 4	4	to	5
<input checked="" type="checkbox"/> 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

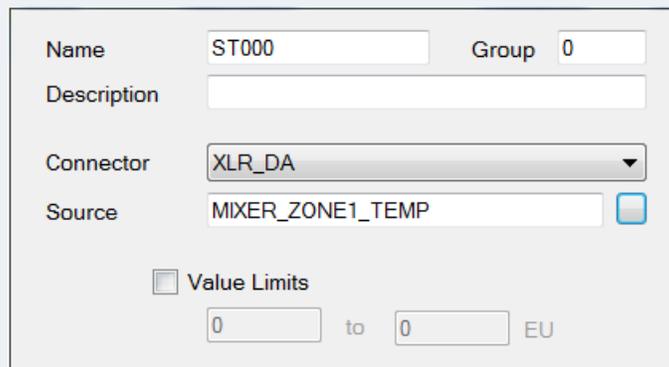
Name	DF000	Group	0
Description			
Connector	XLR_DA		
Source	PROD_TOTAL	<input type="checkbox"/>	
Type	increment		
Rollover	10000		
Scale	1		

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



The screenshot shows a configuration form for a statistic. It includes the following fields and options:

- Name:** ST000
- Group:** 0
- Description:** (empty text box)
- Connector:** XLR_DA (dropdown menu)
- Source:** MIXER_ZONE1_TEMP (text box with a refresh icon)
- Value Limits:** A checkbox that is currently unchecked, followed by two text boxes containing the number 0, a 'to' label, and the unit '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 **Snapshot** 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.

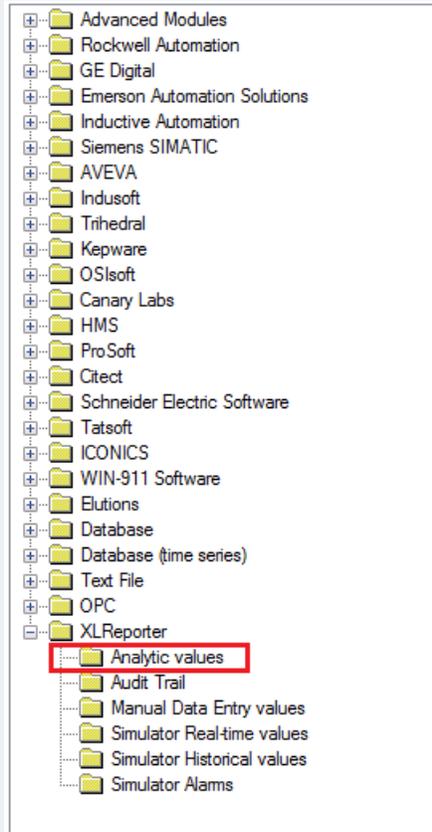
Analytic Database

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.

A screenshot of a configuration dialog box for a connector. It has the following fields: 'Connector Name' with the value 'XLR_Analytic'; 'Description' which is empty; 'Primary Database' section containing 'Type' set to 'Microsoft Access' and 'Data Source' set to 'C:\XLRprojects\XLR_Demo\Data\XLRanalytic.mdb'. There is a browse button [...] next to the Type field and another browse button [...] next to the Data Source field. A 'Settings' button is located at the bottom right of the dialog.

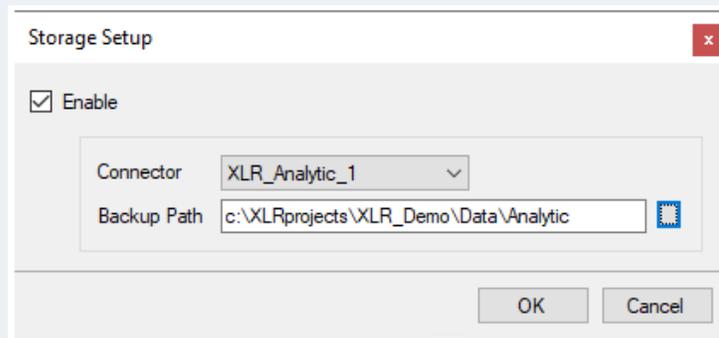
The database is defined by clicking the browse pushbutton [...] for **Primary Database**. There are options for Microsoft SQL Server and MySQL. If these are not available, a blank Microsoft Access database is always installed with the project. Browse to the *Data* folder of the project and select *XLRanalytic.mdb*.

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.



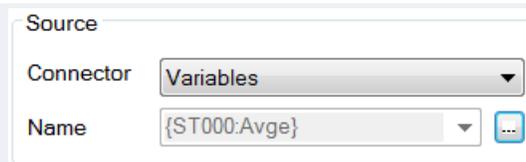
Notice that **Enable** is checked and the **Connector** set up is selected.

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.

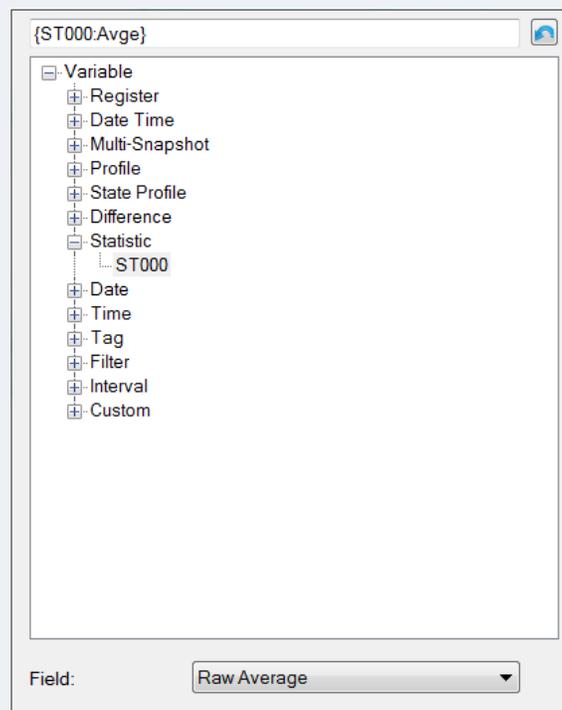
Reporting on Analytics

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

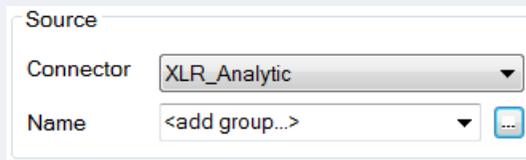
Real Time Analytics



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



Analytic Data 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:

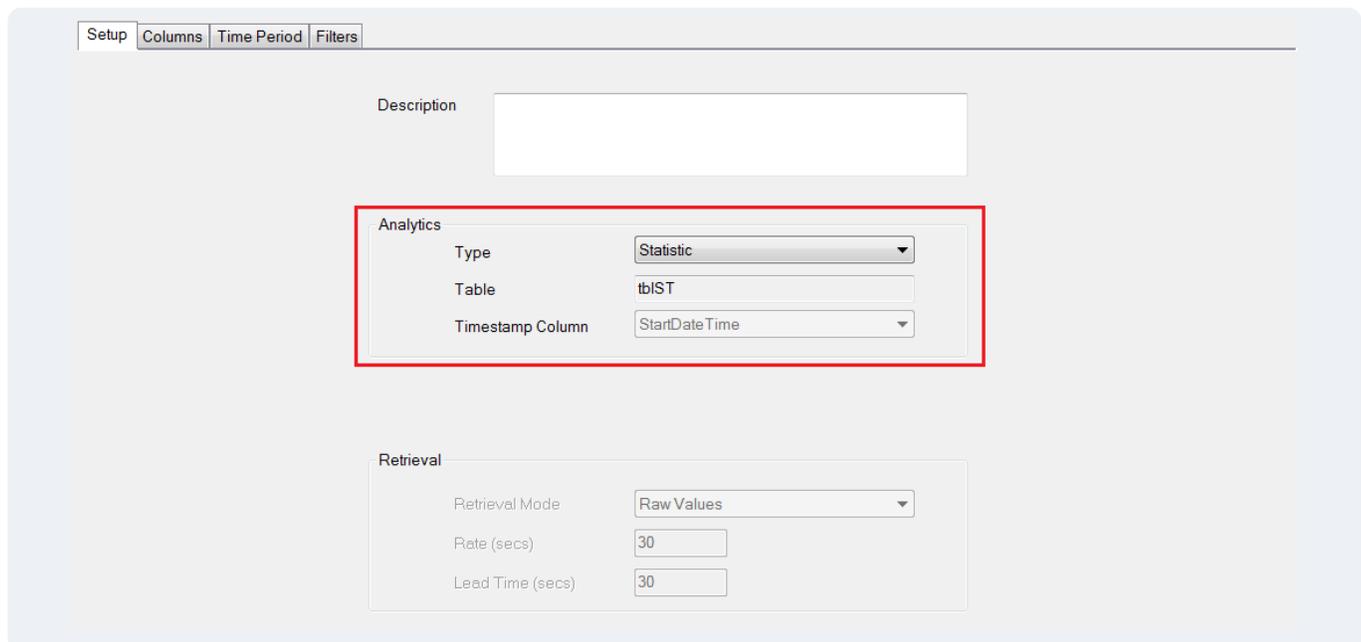
- **Summary Values**

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.



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

Analytic Designer

The screenshot shows the 'Data Items' tab in the Analytic Designer. On the left, there is a tree view under 'Analytic' with sub-items: Snapshot, Multi-Snapshot, Profile, State Profile, Difference, and Statistic. The main area is a table with the following data:

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.

The screenshot shows the 'Schedule' tab in the Analytic Designer. It features a toolbar with 'Add', 'Modify', 'Delete', 'Outline', and 'Test' buttons. Below the toolbar is a table with the following data:

Condition	Action
<input checked="" type="checkbox"/> Continuous	Recur 15 minutes(s); <every day>; 00:00:00 StoreAll Snapshot

Below the table, there is a text input field containing '* add schedule'.

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

Setup Columns **Time Period** Filters

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.

Setup Columns **Time Period** Filters

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*.