

## 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 are 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

The screenshot shows a configuration form for a Snapshot analytic. It includes fields for Name (SS000), Group (0), Description, Connector (XLR\_DA), and Source (MIXER\_ZONE1\_TEMP). There are also three optional sections: Value Limits (0 to 0 EU), Value Deadband (0 EU), and 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.

## Multi-Snapshot

The screenshot shows a configuration form for a Multi-Snapshot analytic. It includes fields for Name (SM000), Group (0), Description, Connector (XLR\_DA), and a primary Source (Cycle\_ID). Below these are four additional Source fields: Cycle\_Lot, Cycle\_Product, and Cycle\_Operator.

Multi-Snapshot is not an analytic that provides data for a report, but instead is an analytic that can be used to frame an event like the timestamp when a cycle starts and stops.

Up to four values can be read from the process and stored along with the start and end times. These are set in the **Source** options available.

## 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

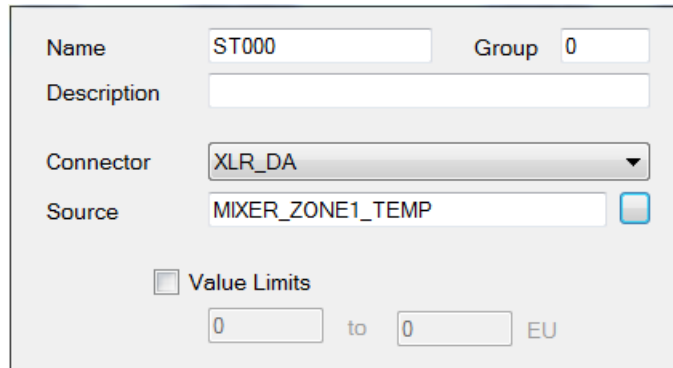
Name	DF000	Group	0
Description			
Connector	XLR_DA		
Source	PROD_TOTAL	<input type="checkbox"/>	
Type	increment		
Rollover	10000	<input type="checkbox"/>	
Scale	1	<input type="checkbox"/>	

Difference is an analytic designed for analog totalizers to determine things like the amount produced over a certain period of time. 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



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

- Total
- Maximum value
- Maximum time
- Minimum value
- 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.

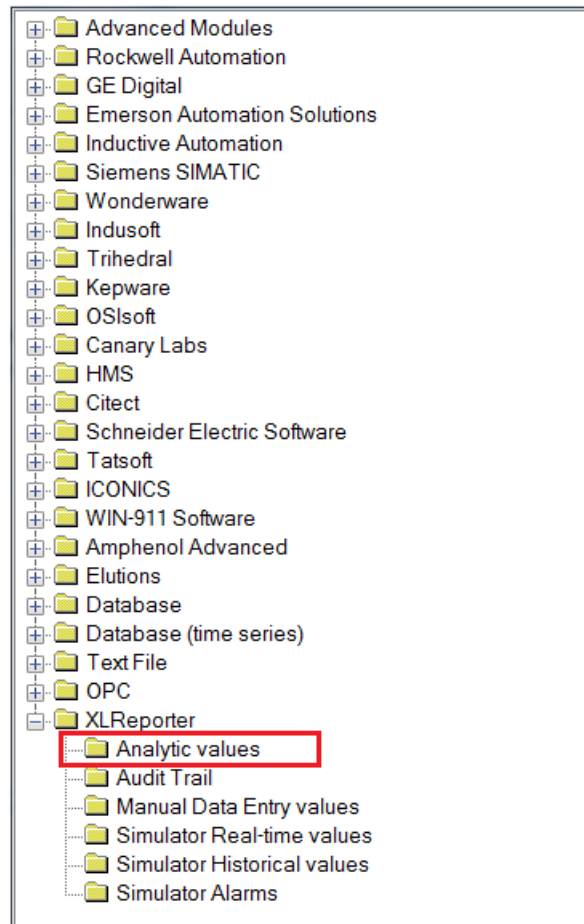
## Analytic Database

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.

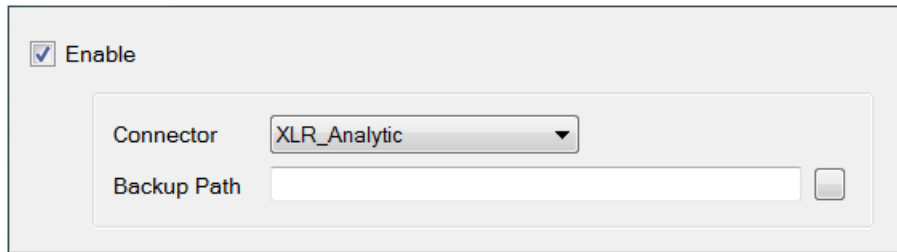
The image shows a configuration dialog box for a connector. The 'Connector Name' field contains 'XLR\_Analytic'. The 'Description' field is empty. The 'Primary Database' section is expanded, showing the 'Type' as 'Microsoft Access' and the 'Data Source' as 'C:\XLRprojects\XLR\_Demo\Data\XLRanalytic.mdb'. A 'Settings' button is located at the bottom right of the dialog box.

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

The other step required to store analytics to a database is to enable database storage in the **Analytic Designer** by clicking the **Storage** menu at the top.



The screenshot shows a dialog box with a checked checkbox labeled "Enable". Below it is a section with two fields: "Connector" is a dropdown menu set to "XLR\_Analytic", and "Backup Path" is a text input field with a browse button to its right.

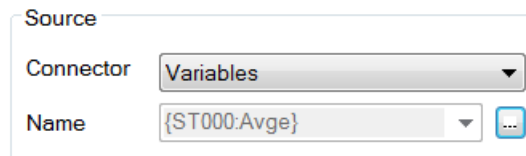
Check **Enable** and specify the **Connector**.

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.

## Reporting on Analytics

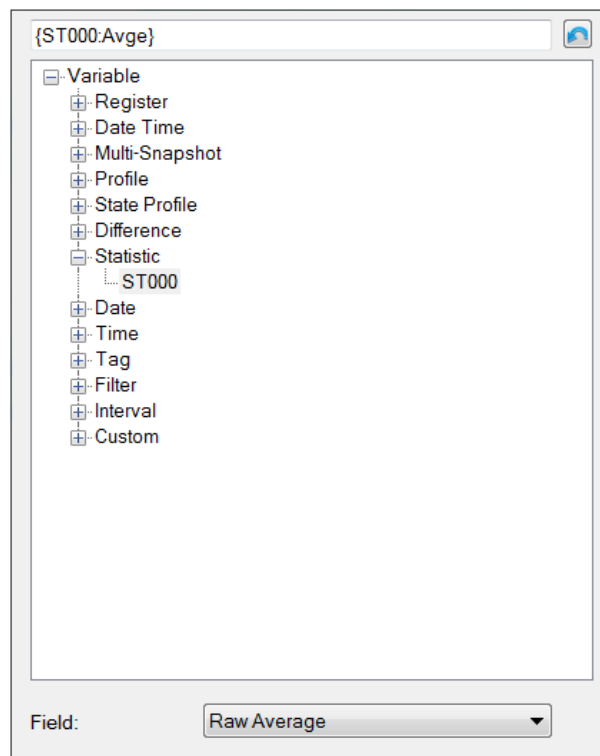
When designing a template there are two ways to configure data connections to analytics.

### Real Time Analytics



The screenshot shows a dialog box with a "Source" section. It contains three fields: "Connector" is a dropdown menu set to "Variables", and "Name" is a dropdown menu set to "{ST000:Avge}" with a browse button to its right.

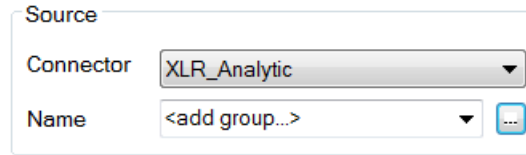
For real time analytics, under the **Source** section of the connection set **Connector** to *Variables*.




The screenshot shows a dialog box with a title bar containing "{ST000:Avge}". The main area is a tree view under the heading "Variable". The tree structure is as follows: Variable (expanded) -> Register, Date Time, Multi-Snapshot, Profile, State Profile, Difference, Statistic (expanded) -> ST000 (expanded) -> Date, Time, Tag, Filter, Interval, Custom. At the bottom, there is a "Field:" label and a dropdown menu set to "Raw Average".

Click the browse pushbutton [...] to access all the available analytics. Select a specific analytic to see a list of fields available for it.

## Analytic Data Group



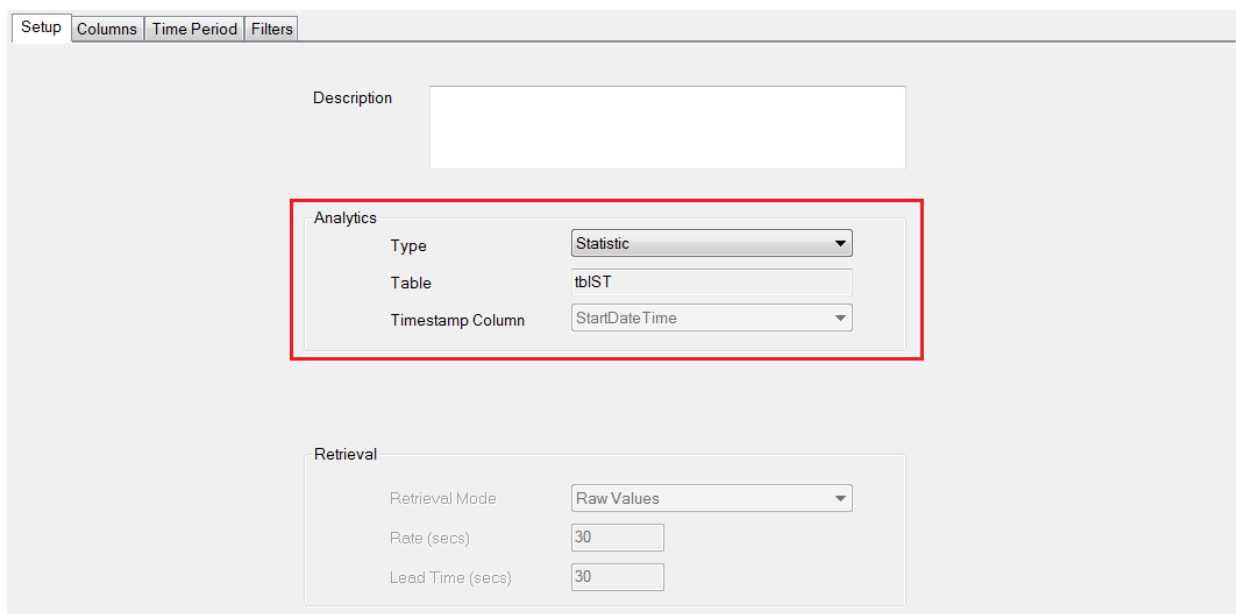
Analytic Data Groups are designed to retrieve the analytic data stored in the database. To configure a group, under the **Source** section set **Connector** to *XLR\_Analytic* (the name of the analytic connector configured). For **Name**, with *<add group>* set, click the browse pushbutton [...].



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.  
This is provided to query out event frames from a Multi-Snapshot analytic.

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.

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:

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:

Duration:

Start At

Day:

Time:

Interval

Count:

All

Every:

Bounds to include:

Endpoints to include:

Time Ordering:

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

## Scenarios

The following presents a few real life scenarios where analytics are used for the reporting solution. In all of these scenarios, analytics are configured to store to a database.

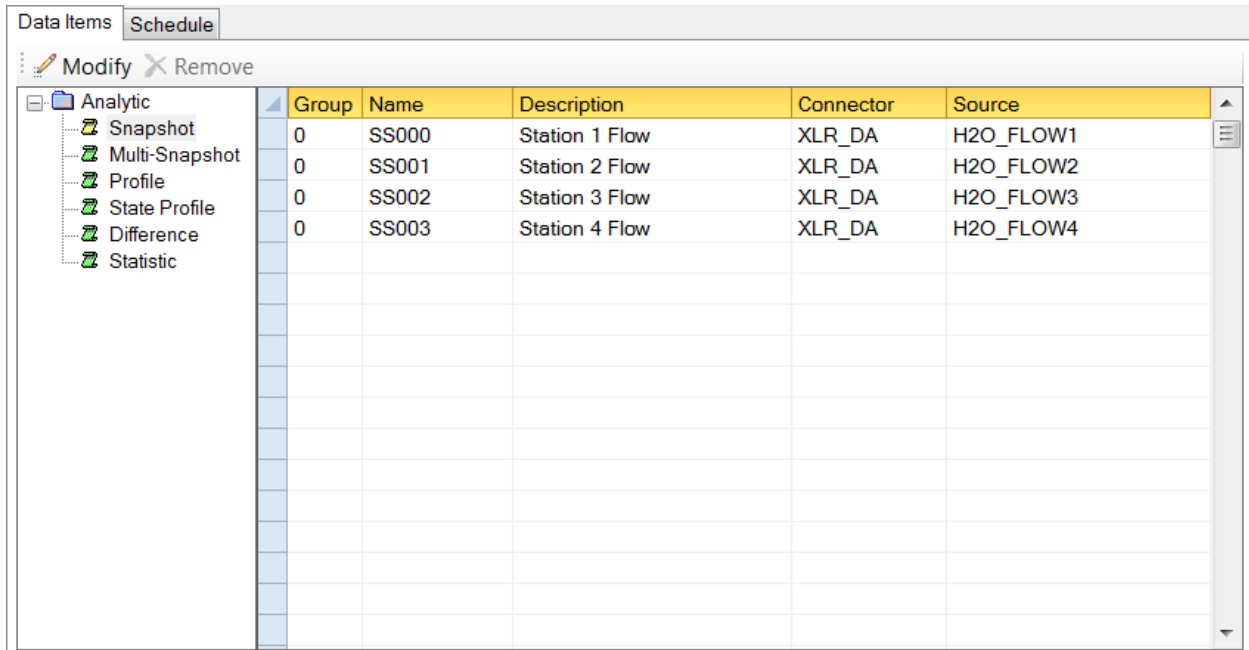
### Multiple Reports with Values

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 of these 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.



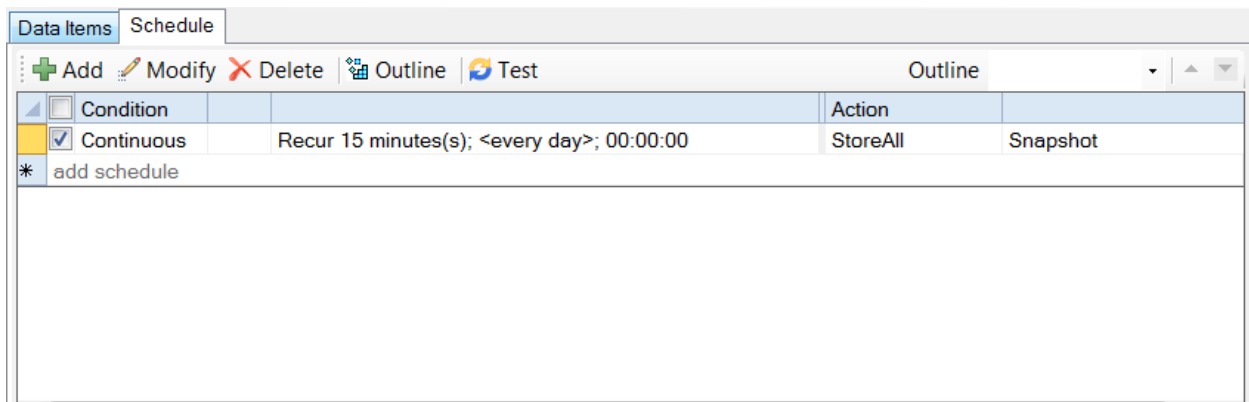
## Analytic Designer



The screenshot shows the 'Data Items' tab with a sub-tab 'Schedule'. On the left, there is a tree view under 'Analytic' with 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 configuration. It includes a toolbar with 'Add', 'Modify', 'Delete', 'Outline', and 'Test' buttons. The main area shows a table with the following data:

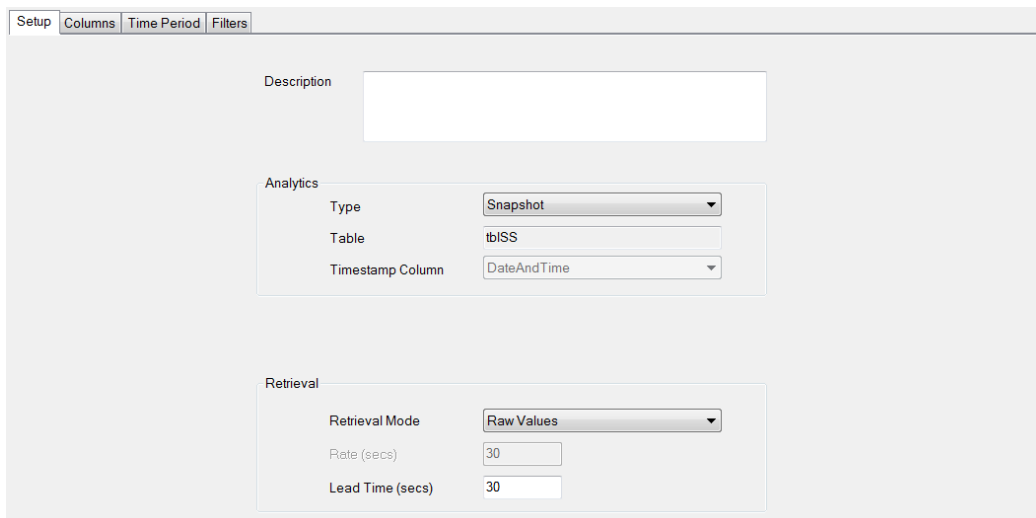
Condition	Action
<input checked="" type="checkbox"/> Continuous	StoreAll

Below the table, there is a text field containing: "Recur 15 minutes(s); <every day>; 00:00:00".

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.



The screenshot shows the 'Setup' tab configuration for a data group. It includes the following fields:

- Description: [Empty text box]
- Analytics:
  - Type: Snapshot (dropdown)
  - Table: tbISS (text box)
  - Timestamp Column: DateAndTime (dropdown)
- Retrieval:
  - Retrieval Mode: Raw Values (dropdown)
  - Rate (secs): 30 (text box)
  - Lead Time (secs): 30 (text box)

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

The screenshot shows the 'Setup' tab with the 'Analytics Type' set to 'Snapshot'. The 'Selected Columns' table is visible, showing four columns: Name, Calculation, Scaling, and Heading. The 'Output Options' section includes a dropdown for 'Timestamp on first column', a checkbox for 'Transpose', and a checkbox for 'Include Heading'.

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:

Transpose

Include Heading

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

The screenshot shows the 'Columns' tab with the 'Time Period' and 'Interval' settings. The 'Time Period' is set to 'Relative' with a duration of 'Current' day. The 'Interval' is set to 'Every' 15 minutes. The 'Bounds to include' is set to 'None' and the 'Endpoints to include' is set to 'Start Time'.

Period

Type:

Duration:

Start At

Day:

Time:

Interval

Count

All

Every

Limit:

minutes

Bounds to include:

Endpoints to include:

Time Ordering:

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.

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

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

## On Demand Batch Report

In this scenario, batch reports are required but rather than generate automatically at the end of the batch, these reports must be available to generate On Demand whenever someone wants one. The data for these reports are stored in a continuous historian.

However, the user will not know anything about the batch except for the name of it. Each batch should be presented to the user to select and once selected, a report is generated for the batch.

## Analytic Editor

To capture the information for each batch a **Multi-Snapshot** analytic is configured.

Not only will this store the start and end time of the batch, but also the Lot and Product ID for each batch. This will help the user identify the batch.

Condition	Action	Multi Snapshot
<input checked="" type="checkbox"/> XLR_DA Cycle_Start = 1	UpdateAll	Multi Snapshot
<input checked="" type="checkbox"/> XLR_DA Cycle_Start = 0	StoreAll	Multi Snapshot

Under the **Schedule** tab there are two actions configured. The first updates the **Multi-Snapshot** when the cycle starts. This captures the start date and time of the batch along with the Lot and Product IDs. The second captures the end date and time and then stores all this information as a record to the database.

## Template Design

As part of the template design, a Data Connection is configured to a History Data Group to retrieve the data required for the batch.

The screenshot shows the 'Time Period' configuration tab. It is divided into three main sections: 'Period', 'Interval', and 'Bounds to include'.  
- **Period:** The 'Type' is set to 'Variable'. Under 'Start', the 'Date' is '{Start Date}' with a 'Date includes Time' checkbox checked. The 'Time' is '{Start Time}'. Under 'End', the 'Type' is 'Time', 'Date' is '{End Date}' with 'Date includes Time' checked, and 'Time' is '{End Time}'.  
- **Interval:** The 'Count' radio button is selected with a value of '60'. The 'Every' radio button is also selected with a value of '5' and a unit of 'minute'.  
- **Bounds to include:** Set to 'None'.  
- **Endpoints to include:** Set to 'Start Time'.  
- **Time Ordering:** Set to 'Ascending'.

Under the **Time Period** tab, the **Start** and **End** are set to **Variables** that will be set when the report is run On Demand.

## On Demand Design

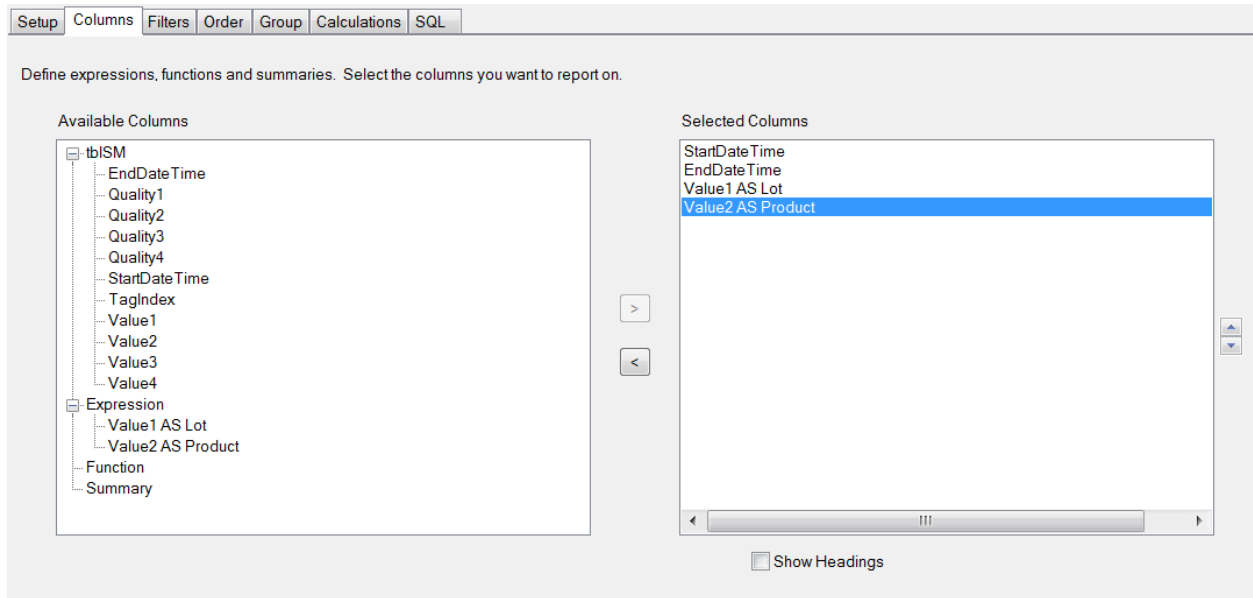
In order to present the user with a list of batches, a Database Data Group is configured to query the batches.

This is done in the **Project Explorer** under the **Tools** tab by selecting **Connector Groups**. For the **XLR\_Analytic**, a new **Custom Values** group is created.

The screenshot shows the 'Database Tables/Views' configuration tab. It includes a 'Connection' section with 'Reconnect' and 'View...' buttons. The 'Database Tables/Views' list shows 'tblDF', 'tblName', 'tblPR', 'tblSM', 'tblSP', 'tblSS', and 'tblST'. The 'Selected Tables/Views' list shows 'tblSM'. A 'Filter' dropdown is set to '\*' and an 'Apply' button is present. The 'Records to Fetch' dropdown is set to 'ALL'. At the bottom, there is a 'Joins' table with columns for 'Column', 'Condition', and 'Column'.

Column	Condition	Column

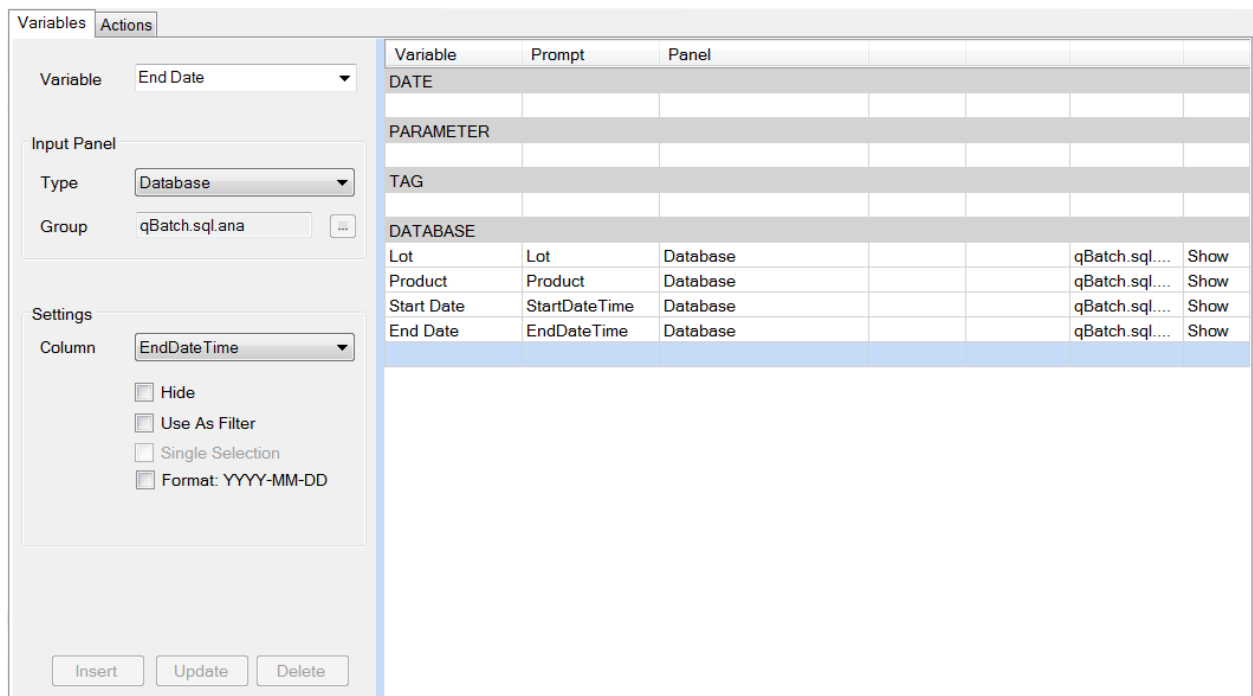
Under the **Columns** tab, the **tbISM** table is selected. This is the table with the Multi-Snapshot analytics.



Under the **Columns** tab, the start date, end date, lot, and product are selected.

This group is saved as *qBatch*.

In the template, the **On Demand Designer** has the settings configured to present the batches to the user.



This is done by configuring **Variables** configured to the **DATABASE** panel.

Database				
Filter <input type="text" value="None"/>				
	Product	Lot	StartDateTime	EndDateTime
<input checked="" type="checkbox"/>	P990-150	1200404	10/1/2015 1:32:00 AM	10/1/2015 4:07:00 AM
<input type="checkbox"/>	P02369-80	1200362	10/1/2015 7:08:15 AM	10/1/2015 1:01:21 PM
<input type="checkbox"/>	P02369-80	1200363	10/1/2015 4:07:45 PM	10/1/2015 9:15:32 PM
<input type="checkbox"/>	P72-0809	1210043	10/1/2015 10:01:32 PM	10/1/2015 11:56:11 PM
<input type="checkbox"/>	P72-0809	1210044	10/2/2015 6:32:00 AM	10/2/2015 10:08:05 AM
<input type="checkbox"/>	P50-30318	1200350	10/2/2015 1:17:11 PM	10/2/2015 4:21:08 PM
<input type="checkbox"/>	P50-30318	1200351	10/2/2015 7:44:10 PM	10/2/2015 11:12:08 PM
<input type="checkbox"/>	P50-30318	1200352	10/3/2015 1:12:44 AM	10/3/2015 4:45:00 AM
<input type="checkbox"/>	P990-150	1200401	10/3/2015 9:08:07 AM	10/3/2015 11:14:14 AM
<input type="checkbox"/>	P990-150	1200402	10/3/2015 1:26:10 PM	10/3/2015 6:32:15 PM
<input type="checkbox"/>	P990-150	1200403	10/3/2015 7:28:03 PM	10/3/2015 11:32:39 PM

When the user goes to generate a report, they are presented with a list of batches that have occurred.