

Export to Database

Overview

In many reporting applications, the information produced must be stored not only in human-readable formats like workbooks or web pages, but to other data systems such as relational databases. **XLReporter** does this by mapping cells in the template workbook to columns in a relational table and allowing the user to design an SQL statement that defines how the data collected in the report is stored to the database.

This spotlight document highlights a few useful configurations of this feature.

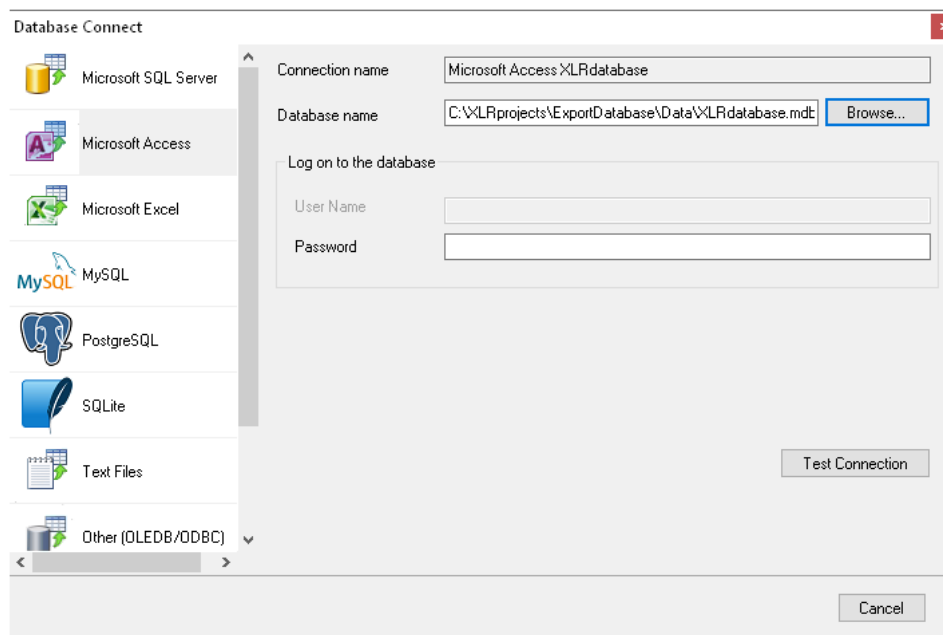
Set up

Configure a Connector

A **Data Connector** is required to export data to a database (and also to collect it *from* a database). This is defined in the **Project Explorer** under the **Data** tab by selecting **Connectors**. Click the Add button to add a new connector.

A **Database** connector is required and is configured by expanding **Database** and select one of the available formats.

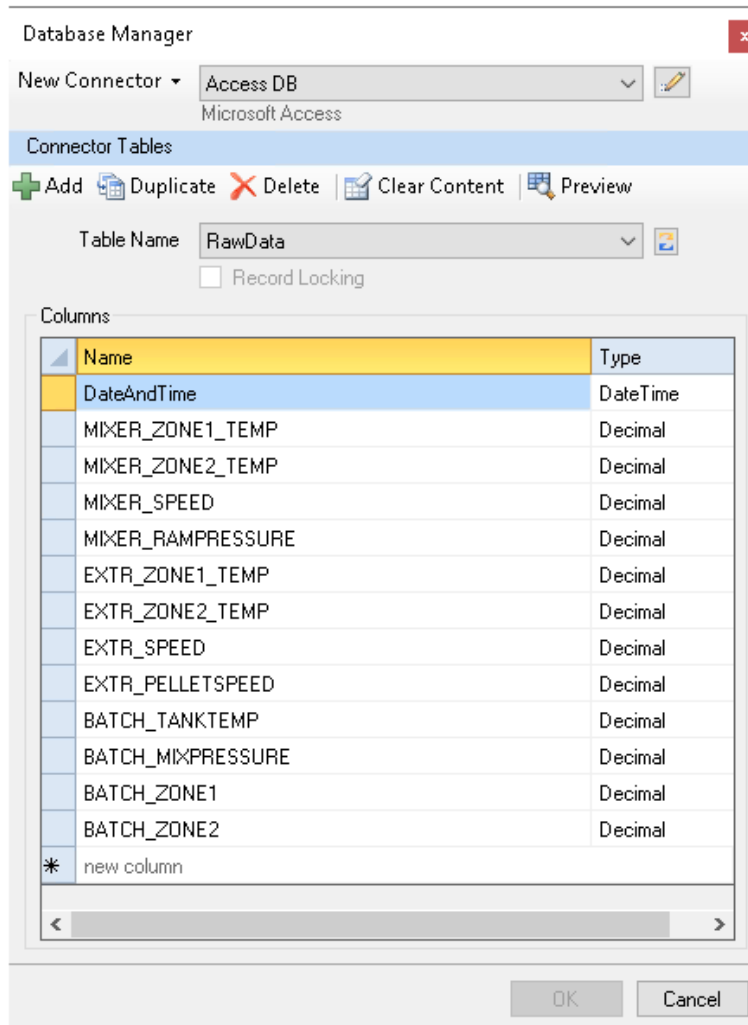
To define the database to export the data to, click the browse pushbutton [...] for **Primary Database** and specify the connection parameters



The connection parameters depend on the database format. In the above example, a **Microsoft Access** database included in your **XLReporter** project is connected from the `\Data` folder of the active project. Use this database if there isn't one already available on the SCADA network.

Create Table and Columns

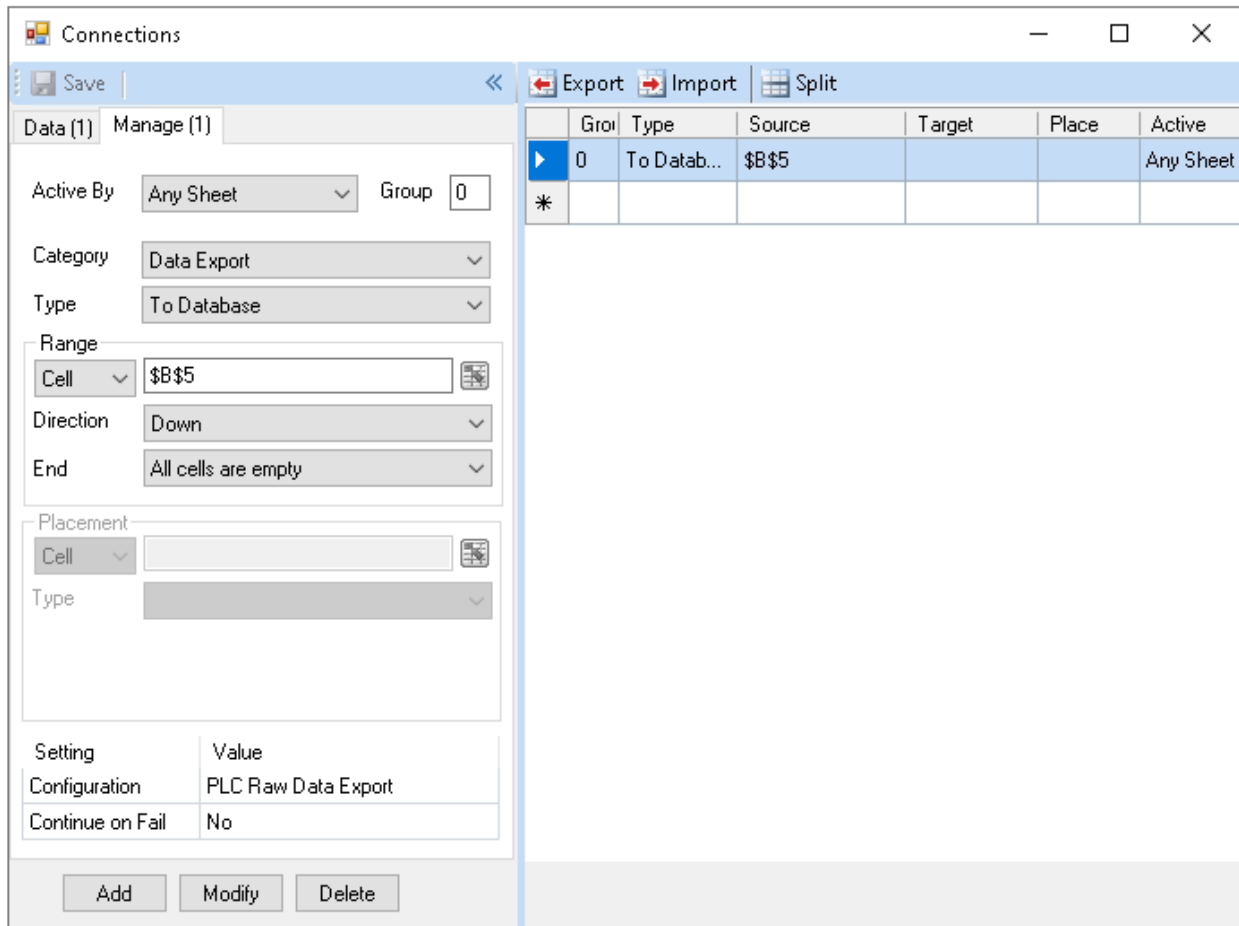
XLReporter provides tools to create a table and the necessary fields to contain the exported data. To configure these, in the **Project Explorer**, under the **Tools** menu, open the **Database Manager**.



In **Database Manager**, select the **Database Connector** and then click **Add** to create a new table. In the **Columns** area, add a column for each value to be exported, and assign the appropriate type.

Data Export Connection

The **Data Export To Database** management connection is assigned from within a **Report Template**. So, first, identify the template which collects values that need to be exported. Or, create a new one. From the **Template Studio**, open the **Connect** menu and select the **Manage** tab.



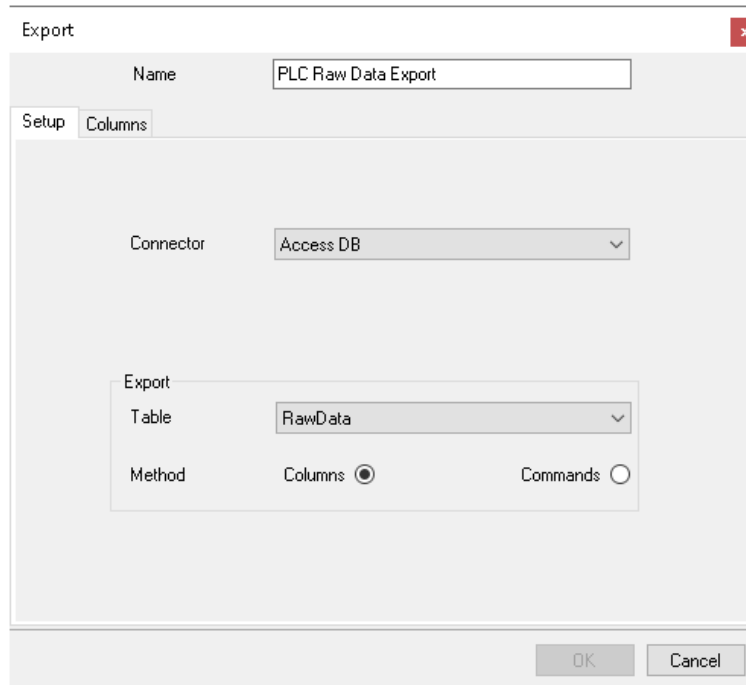
Here, set the **Category** to *Data Export*, and **Type** to *To Database*. This connection takes the following parameters:

- **Active By** determines the sheet in the template from which the values are exported. For templates with multiple sheets, set this to a specific sheet.
- **Group** is used to add update logic which processes connections in a specific order, or under specific **Conditions** in the project schedule. If all **Data** and **Management** connections are set to **Group 0**, then the data connections are updated in top-down order, followed by the management connections in top-down order.
- **Range** determines how many rows are processed by the export connection. In this example image, the connection evaluates the cell *\$B\$5*, and if that cell has content, processes the export connection once. Then, the connection examines each row below the **Range** cell (**Direction Down**) and processes the connection again for each row until a blank cell is found in column *B*.
- **Configuration** uses a browser pushbutton [...] to launch the **Export Configurator**.
- **Continue on Fail** determines what happens if the export SQL statement fails when multiple rows or columns of data are processed for the export.

For example, if the **Range** is set to export three rows of data and the export of the first row fails, if this is set to *No*, the export stops here. If set to *Yes*, the next two rows are exported.

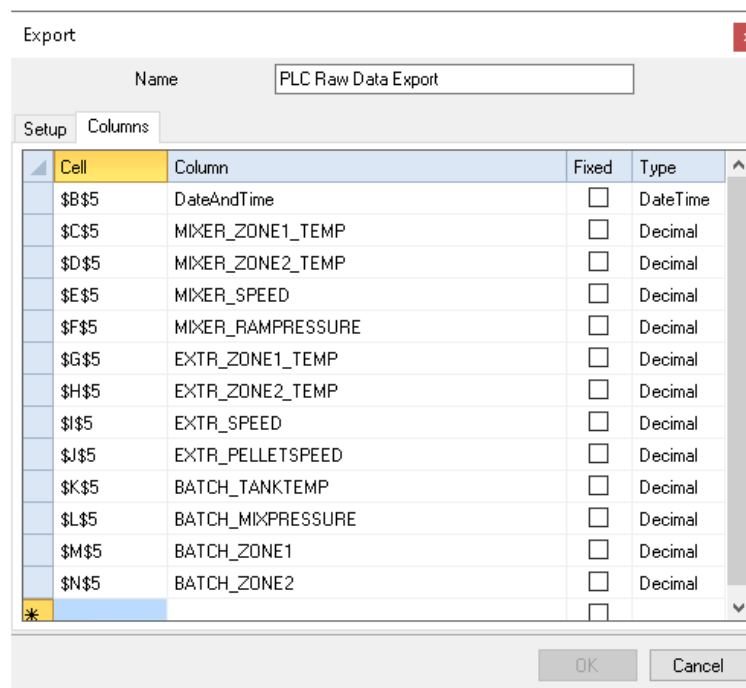
Export Configuration

The **Configuration** parameter of the connection maps cell values in the worksheet to the columns in the database table.



The screenshot shows the 'Export' dialog box with the 'Setup' tab selected. The 'Name' field contains 'PLC Raw Data Export'. The 'Connector' dropdown is set to 'Access DB'. The 'Table' dropdown is set to 'RawData'. The 'Method' section has 'Columns' selected with a radio button, and 'Commands' is unselected. The 'OK' and 'Cancel' buttons are at the bottom right.

The **Setup** tab specifies the **Connector** and **Table** used in the export operation. The export can be designed by *Columns*, or by *Commands*.

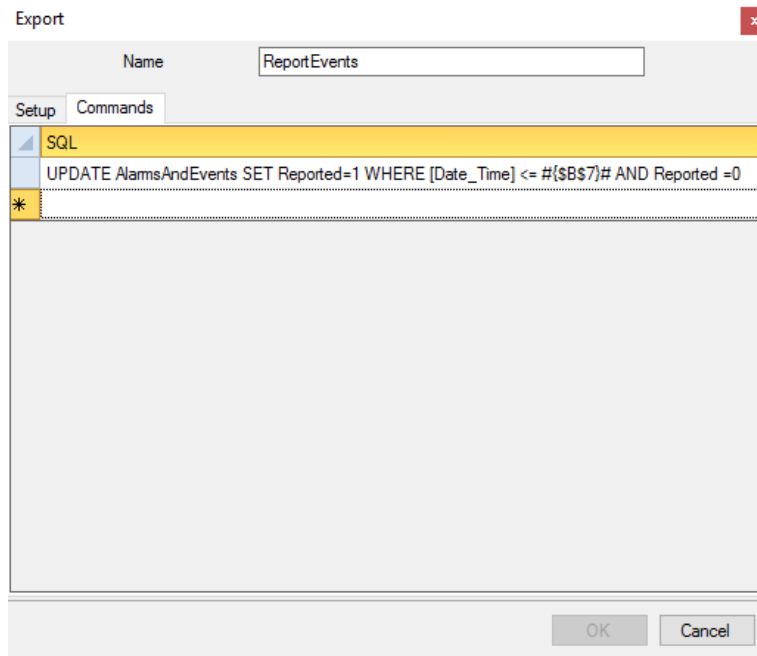


The screenshot shows the 'Export' dialog box with the 'Columns' tab selected. A table maps worksheet cells to database columns. The table has four columns: 'Cell', 'Column', 'Fixed', and 'Type'. The 'Fixed' column contains checkboxes, and the 'Type' column contains data types. The 'Cell' column contains cell addresses like '\$B\$5', '\$C\$5', etc. The 'Column' column contains database column names like 'DateAndTime', 'MIXER_ZONE1_TEMP', etc. The 'Fixed' column has checkboxes, and the 'Type' column has data types like 'DateTime', 'Decimal', etc. The 'OK' and 'Cancel' buttons are at the bottom right.

Cell	Column	Fixed	Type
\$B\$5	DateAndTime	<input type="checkbox"/>	DateTime
\$C\$5	MIXER_ZONE1_TEMP	<input type="checkbox"/>	Decimal
\$D\$5	MIXER_ZONE2_TEMP	<input type="checkbox"/>	Decimal
\$E\$5	MIXER_SPEED	<input type="checkbox"/>	Decimal
\$F\$5	MIXER_RAMPRESSURE	<input type="checkbox"/>	Decimal
\$G\$5	EXTR_ZONE1_TEMP	<input type="checkbox"/>	Decimal
\$H\$5	EXTR_ZONE2_TEMP	<input type="checkbox"/>	Decimal
\$I\$5	EXTR_SPEED	<input type="checkbox"/>	Decimal
\$J\$5	EXTR_PELLETSPEED	<input type="checkbox"/>	Decimal
\$K\$5	BATCH_TANKTEMP	<input type="checkbox"/>	Decimal
\$L\$5	BATCH_MIXPRESSURE	<input type="checkbox"/>	Decimal
\$M\$5	BATCH_ZONE1	<input type="checkbox"/>	Decimal
\$N\$5	BATCH_ZONE2	<input type="checkbox"/>	Decimal
*		<input type="checkbox"/>	

With *Columns*, the **Columns** settings result in an INSERT statement. This is made by mapping the cells in the worksheet to fields in the table.

The **Fixed** parameter is applicable when multiple rows processed based on the **Direction** parameter of the connection. If this is checked, the same cell is used for each row to process whereas if this is unchecked the cell changes based on the row.



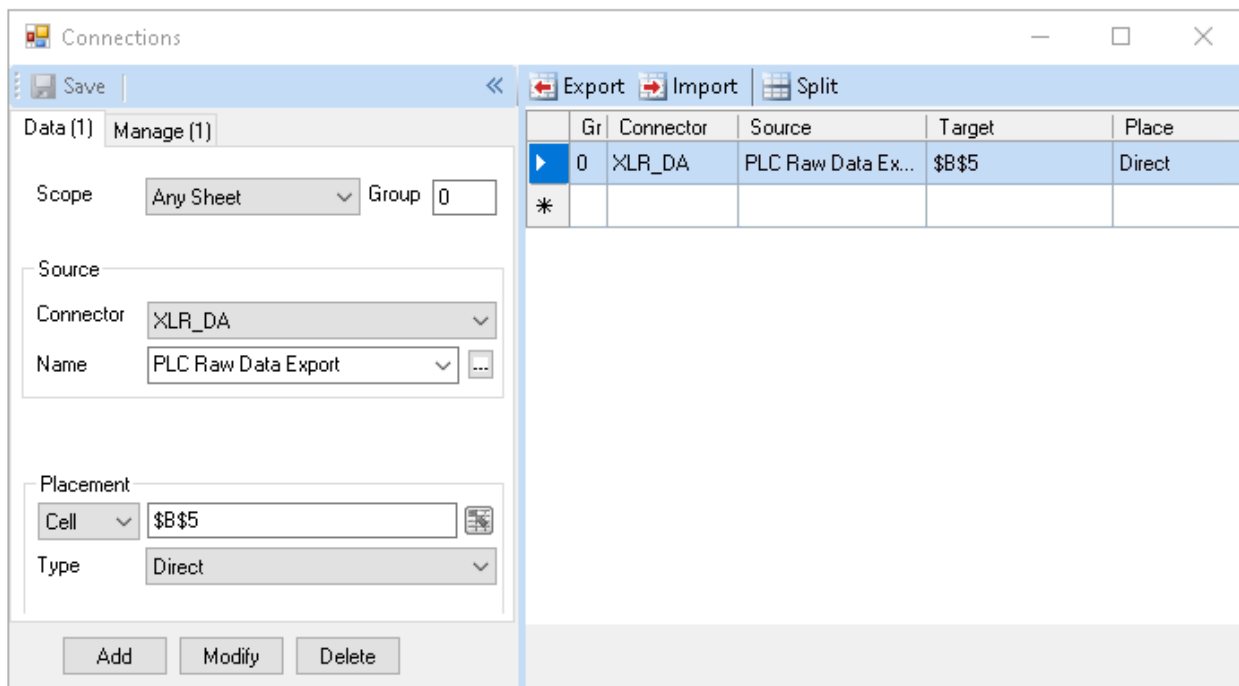
With *Commands*, the export is designed as a written SQL statement that can use any SQL functionality supported by the underlying database engine. Cell values are referenced using {} syntax, e.g. {B\$7}.

With *Commands*, **Fixed** cells are denoted with absolute references, e.g. {B\$7} vs {B7} for a cell reference that changes based on the row.

Scenario: Export Summary Data

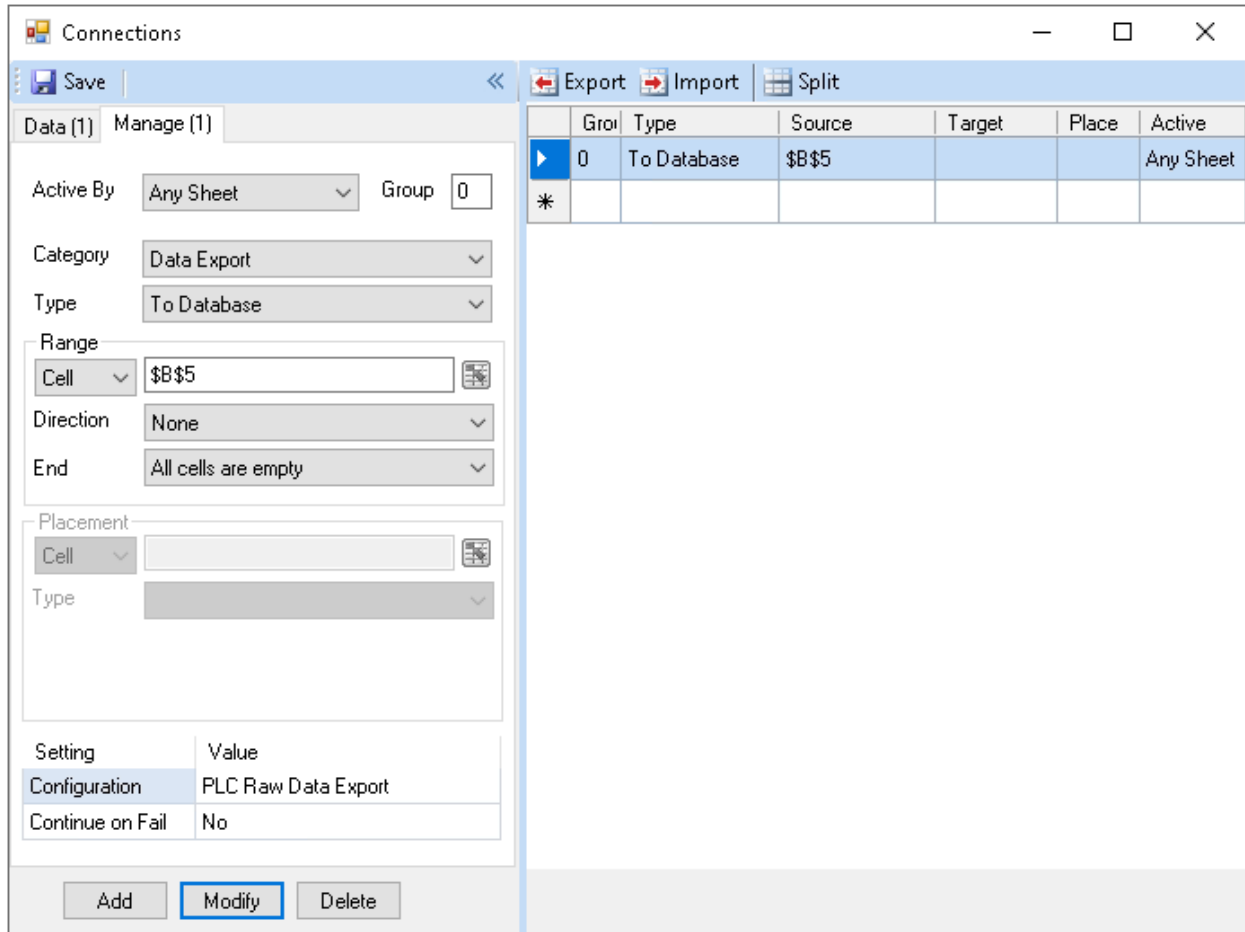
One example where the export feature is useful is when a report is logging live values from a PLC to produce a historical record set in a database. With this approach, the exported data could be queried later in another report.

Data Connection



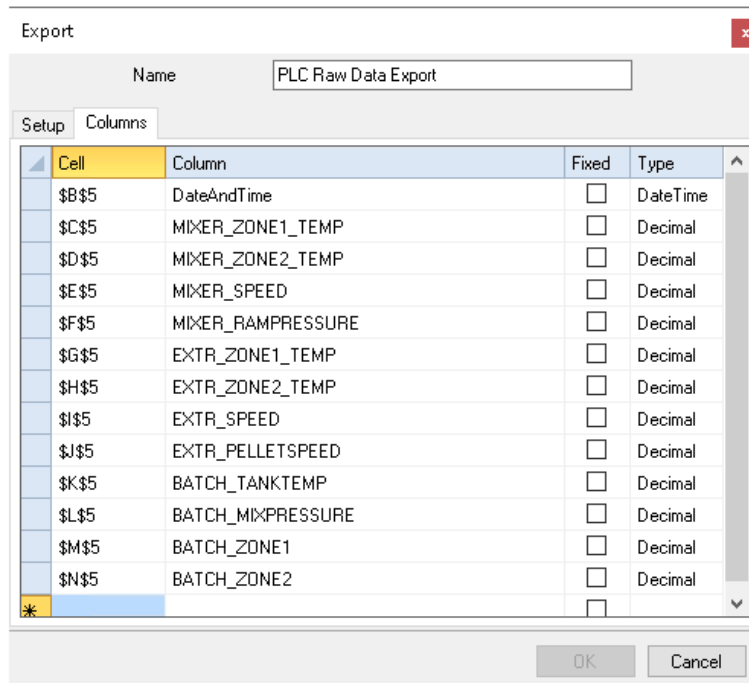
The **Data Connection** is a **Real Time Values Data Group** which brings back a single row of data and overwrites the data onto row 5 using the **Placement Type** *Direct*.

Management Connection



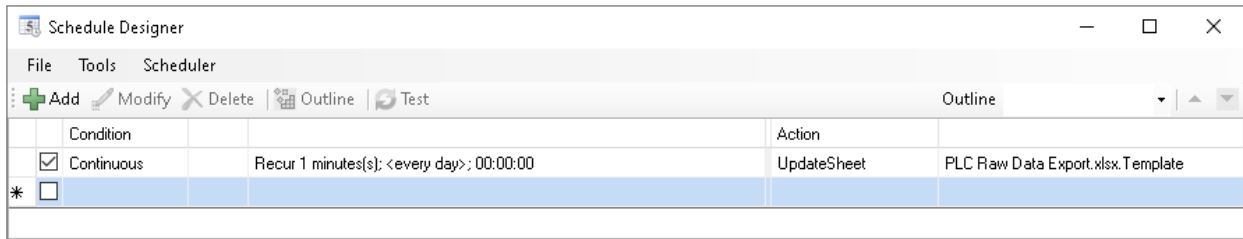
The connection in this scenario runs with **Direction None**, so it is processed only once.

Export Configuration



The **Export Configuration** maps the cells across row 5 where the **Data Connection** is placed to the fields in the table.

Schedule



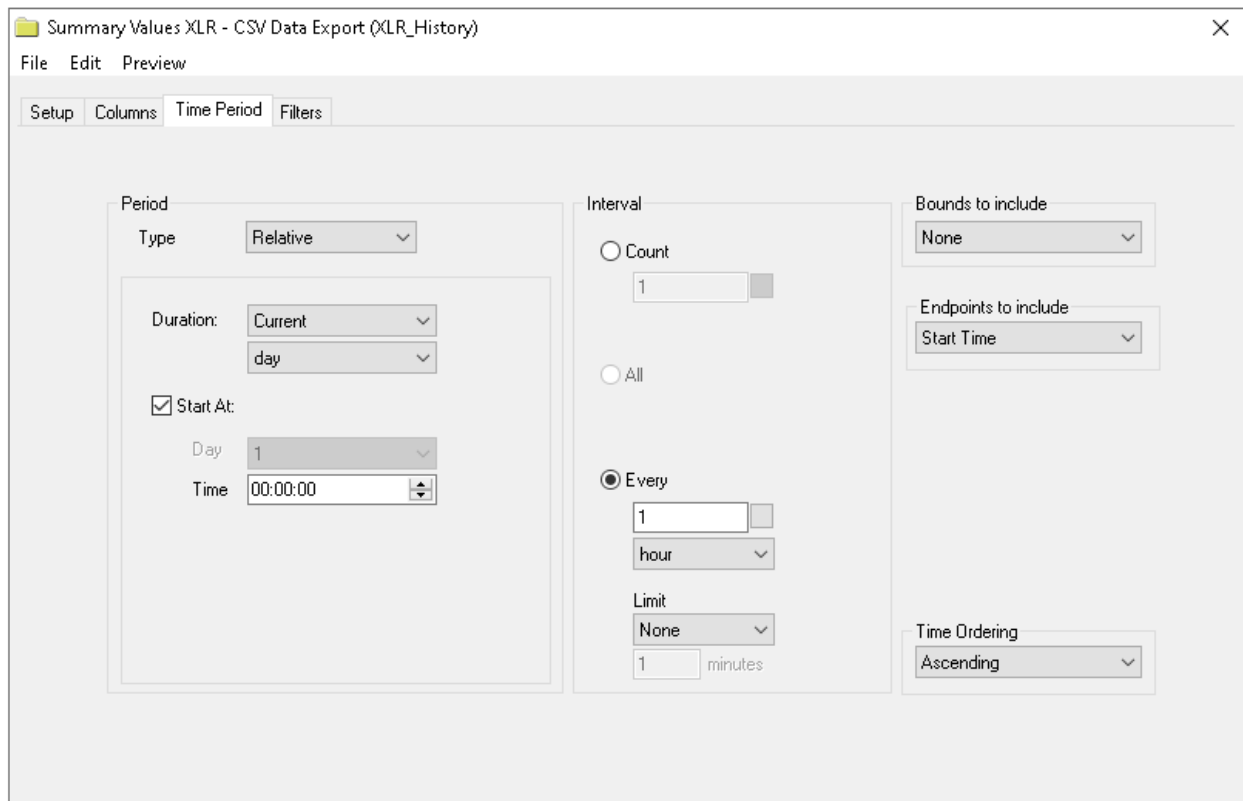
	Condition		Action	Outline
<input checked="" type="checkbox"/>	Continuous	Recur 1 minutes(s); <every day>; 00:00:00	UpdateSheet	PLC Raw Data Export.xlsx.Template
* <input type="checkbox"/>				

The report is updated every minute on the schedule. In this configuration, the update rate of the report effectively sets the resolution on the historical data. So in this case there are 1-minute samples recorded continuously.

Scenario: Export Summary Data

In this scenario, the Report Template produces hourly summary values on a daily report sheet from CSV files connected via a **Historical Data Connector**. The Entire table of 24 rows is exported to the database.

Data Connection



Summary Values: XLR - CSV Data Export (XLR_History)

File Edit Preview

Setup Columns Time Period Filters

Period

Type: Relative

Duration: Current day

Start At:

Day: 1

Time: 00:00:00

Interval

Count: 1

All

Every: 1 hour

Limit: None

1 minutes

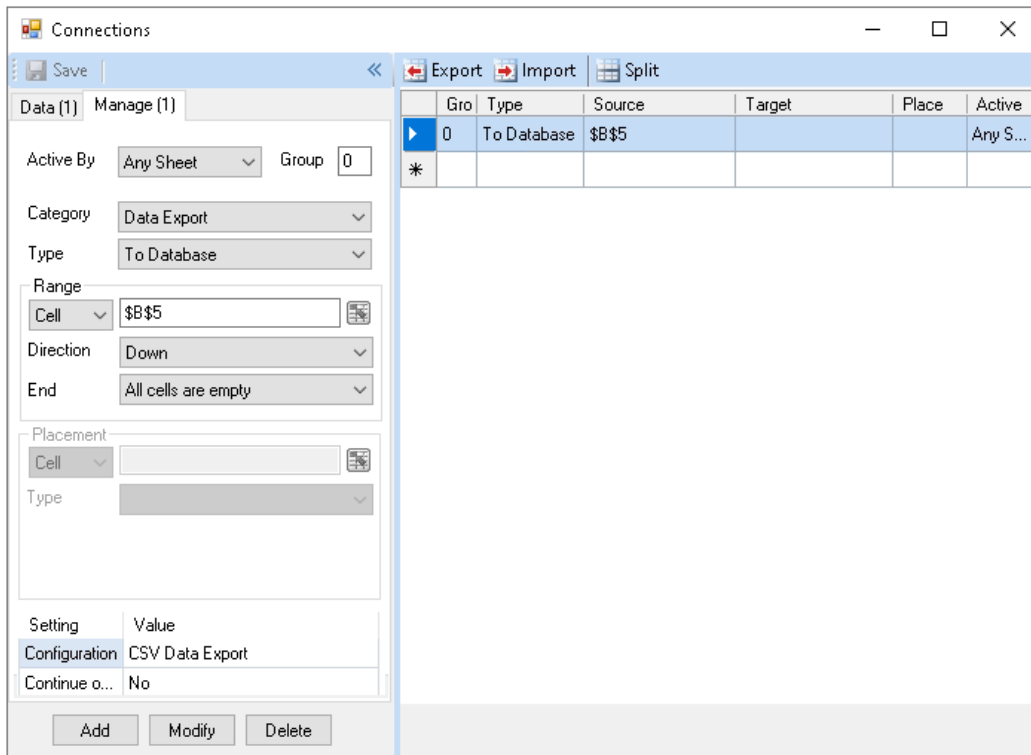
Bounds to include: None

Endpoints to include: Start Time

Time Ordering: Ascending

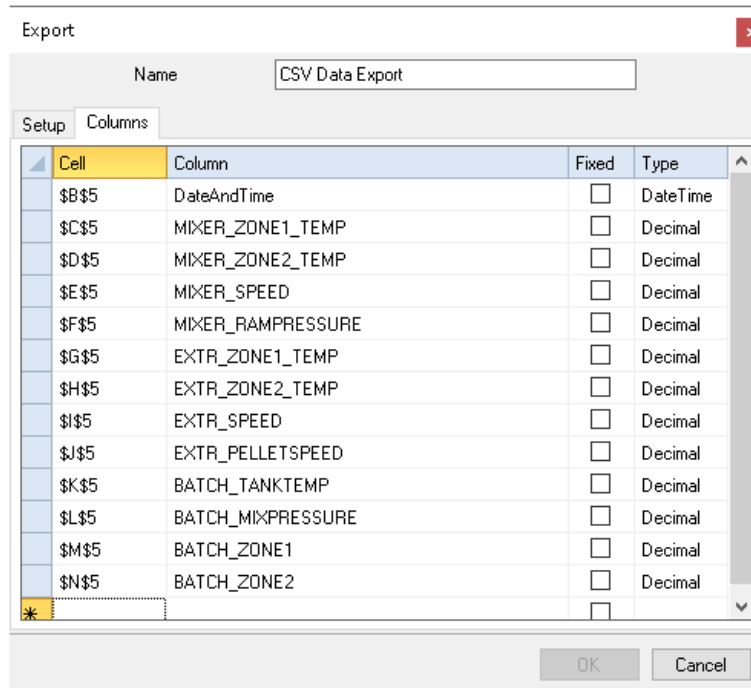
Here, the **Data Connection** is a **Summary Values History Data Group** which collects hourly averages over a day from the raw samples recorded to the text files.

Management Connection



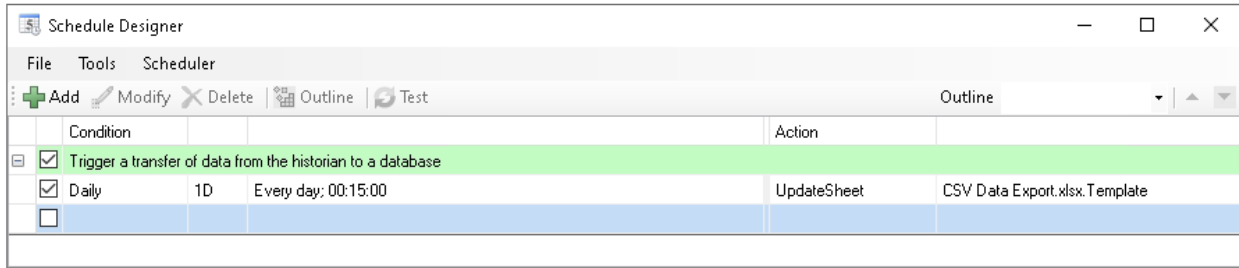
The **Data Export** connection is set with **Direction** *Down* starting in cell *\$B\$5*.

Export Configuration



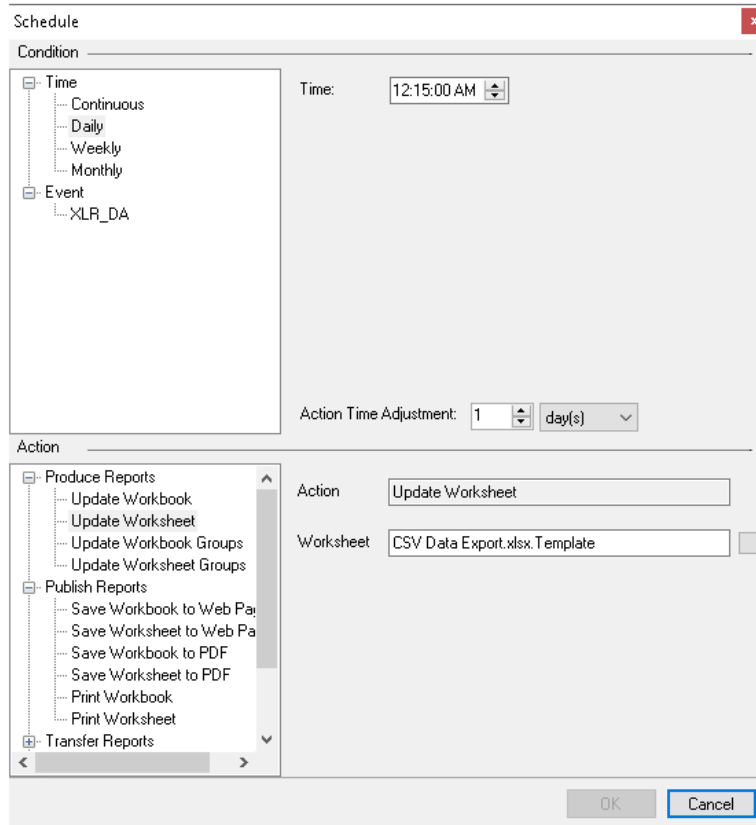
Like the previous example, this **Export Configuration** maps the cells in row 5 to the tag columns in the table. Combined with the **Direction** this means that each time the report is updated, 24 rows of hourly averages for each tag are exported to the database.

Schedule



Condition		Action
<input checked="" type="checkbox"/>	Trigger a transfer of data from the historian to a database	
<input checked="" type="checkbox"/>	Daily 1D Every day: 00:15:00	UpdateSheet CSV Data Export.xlsx.Template
<input type="checkbox"/>		

The report is processed daily on the schedule.



Schedule

Condition

- Time
 - Continuous
 - Daily
 - Weekly
 - Monthly
- Event
 - XLR_DA

Time: 12:15:00 AM

Action Time Adjustment: 1 day(s)

Action

- Produce Reports
 - Update Workbook
 - Update Worksheet
 - Update Workbook Groups
 - Update Worksheet Groups
- Publish Reports
 - Save Workbook to Web Page
 - Save Worksheet to Web Page
 - Save Workbook to PDF
 - Save Worksheet to PDF
 - Print Workbook
 - Print Worksheet
- Transfer Reports

Action: Update Worksheet

Worksheet: CSV Data Export.xlsx.Template

OK Cancel

The report is scheduled at **12:15 AM** with an **Action Time Adjustment** of **1 day**. This means that at the beginning of the day, the data is collected for the previous day and exported to the database. This 15-minute buffer time is designed so that there is no possibility that an HMI or other program is writing data to the source CSV files at the time the report is updated. This could actually apply to any historical data source.

Export History Summary to Database		Table	xlrHistory									
	Date and Time	Mixer zone	Mixer zone	Mixer spee	Mixer ram	Extruder zi	Extruder zi	Extruder s	Extruder p	Batch tank	Batch m	
Hourly avg, to database	3/24/2020	76.29378	77.7899	50.11347	62.7249	57.38239	78.33914	81.31617	78.24917	57.01231	44.3122	
	3/24/2020 1:00	78.50815	77.83454	35.72091	70.45105	56.37327	78.09217	76.05265	68.01163	57.71216	63.6276	
	3/24/2020 2:00	63.42646	79.14045	39.03443	79.76628	56.30211	78.20365	73.48987	83.38987	57.33865	86.9156	
	3/24/2020 3:00	62.54867	78.29209	39.14787	87.41651	58.13102	80.96239	74.32904	84.32904	54.31377	106.041	
	3/24/2020 4:00	77.57018	79.96127	43.05945	90.7293	56.51297	84.04023	78.24059	79.30902	50.5458	114.323	
	3/24/2020 5:00	78.77362	89.80178	54.05834	88.54739	55.28711	72.92641	72.49602	72.77997	49.58016	108.866	
	3/24/2020 6:00	54.45997	88.95308	54.86505	81.63299	53.55337	77.96192	76.5917	81.35332	54.82082	91.5824	
	3/24/2020 7:00	77.40626	84.17875	53.20966	72.4015	58.35033	78.29891	75.06755	74.97374	54.48899	68.5037	
	3/24/2020 8:00	78.54325	81.42709	43.89915	64.07773	57.21693	78.02009	74.33768	78.59237	57.42621	47.6943	
	3/24/2020 9:00	71.92887	79.65299	44.98925	59.56943	57.7532	78.05295	80.86599	67.55386	58.403	36.4235	
	3/24/2020 10:00	65.99177	82.95128	44.10781	60.45147	54.04886	81.35124	79.98458	77.72213	46.66602	38.6286	
	3/24/2020 11:00	78.28986	84.4767	35.24337	66.41573	56.47893	82.87666	73.87808	77.53149	56.96371	53.5393	
	3/24/2020 12:00	72.53671	82.42147	37.97318	75.37872	54.50236	80.82143	77.37164	80.78209	51.12887	75.946	
	3/24/2020 13:00	65.96844	85.79614	38.61922	84.20941	54.6002	84.1961	78.01771	78.27592	50.96369	98.0235	
	3/24/2020 14:00	65.21491	92.69109	50.67328	89.82299	54.34421	91.09105	74.22708	73.63772	52.88678	112.057	
	3/24/2020 15:00	77.2387	63.44101	53.35346	90.25847	55.68132	77.98359	78.42518	82.98842	52.3913	113.146	
	3/24/2020 16:00	73.1711	75.95744	52.92967	85.36373	57.5052	78.84777	78.0014	78.27551	58.35276	100.905	
	3/24/2020 17:00	65.20303	77.91797	58.37722	76.84864	55.93528	81.21839	75.64199	67.2408	46.53022	79.621	
	3/24/2020 18:00	78.08464	79.92671	58.41835	67.68776	55.69255	83.98433	73.83933	83.83933	53.12749	56.7194	
	3/24/2020 19:00	67.65434	85.71215	59.45145	61.08126	57.96111	89.76977	74.87247	78.25288	55.66096	40.2031	
	3/24/2020 20:00	53.5077	71.64337	68.91148	59.33698	56.79237	80.17739	76.35739	68.49337	52.18061	35.8424	
	3/24/2020 21:00	65.25359	68.88589	75.69535	63.06424	56.67643	78.82349	76.42249	83.59579	57.68363	45.1605	
	3/24/2020 22:00	77.15601	77.37102	76.03482	70.961	56.58416	78.5494	76.76195	84.67602	52.30849	64.902	

Scenario: Run a Custom SQL Statement

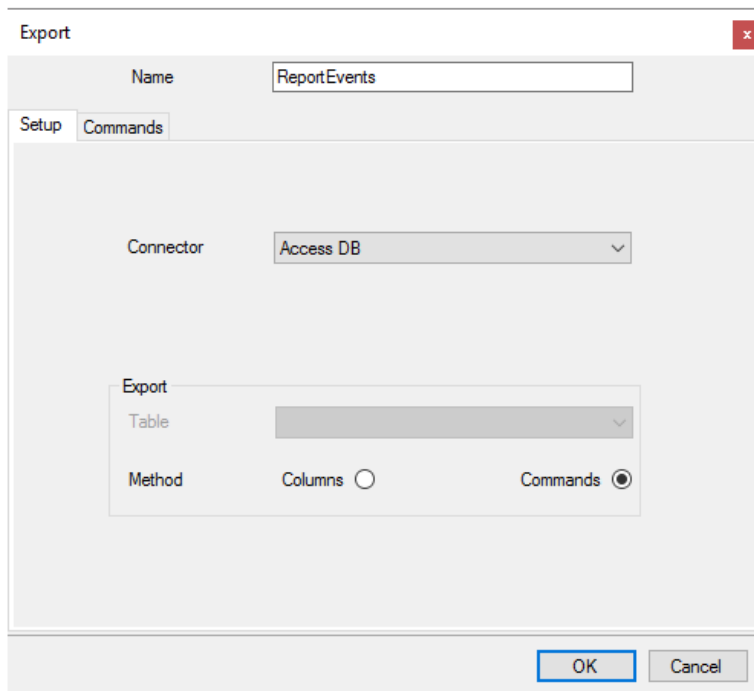
By default, the **Data Export to Database** connection uses an INSERT statement, but it can be reconfigured to process any SQL statement accepted by the back-end database.

Daily Event Report
 XLReporter at Franklin MA Facility
 Sent: Tue 3/17/2020 4:03 PM
 To: DN

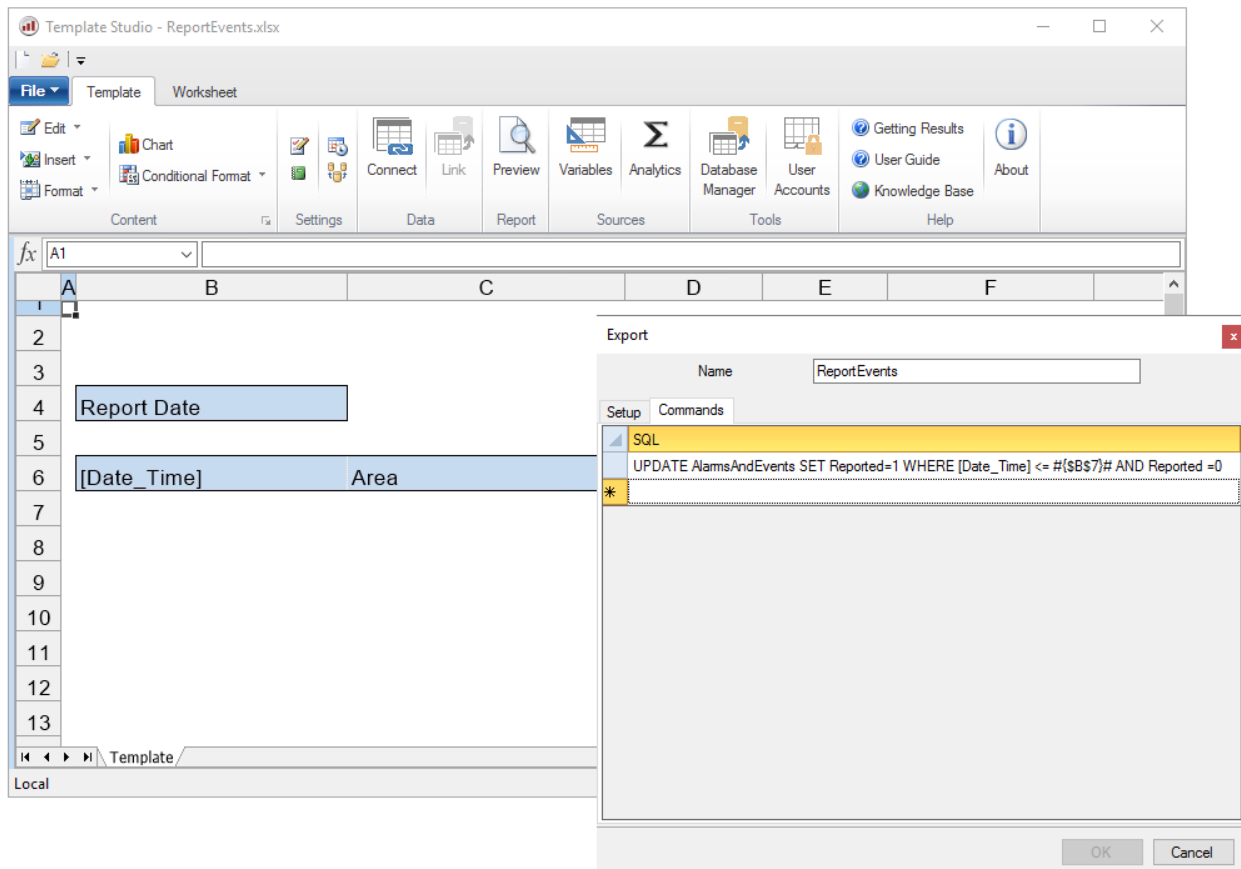
Report Date: 3/17/2020 0:00

[Date_Time]	Area	Attribute	Category	Desc1	Desc2	[Event_Level]	[Event_SubType]Type
03/17/2020 05:43:41	12-63-SVAT	LO_ALM	PROCESS	LOW	Low Alarm Value 2.12611 Limit 6	07-ADVISORY	ALARM
03/17/2020 05:43:28	31_STILLAGE	LO_LO_ALM	PROCESS	LOLO	Low Low Alarm Value 0.490524 Limit 5	15-CRITICAL	ALARM
03/17/2020 05:43:28	31_STILLAGE	LO_LO_ALM	PROCESS	LOLO	Low Low Alarm Value 3.94349 Limit 5	15-CRITICAL	ALARM
03/17/2020 05:43:04	12-32-TRANSPORTER	DISC_ALM	PROCESS	CFN	Change From Normal Value 1	07-ADVISORY	ALARM
03/17/2020 05:24:21	12-31-REFINING	FAIL_ALM	PROCESS	FAILED	Clear	07-ADVISORY	ALARM
03/17/2020 04:43:07	31_STILLAGE	FAIL_ALM	PROCESS	FAILED	Clear	11-WARNING	ALARM

In this scenario, a report template is configured to collect event data from an alarm and event database. The data is filtered based on a field called *Reported=* which returns only data that has not yet been reported by XLReporter. In the Export configuration, there is a custom **Command** option to specify an SQL statement (versus the default *INSERT* statement used with the **Columns** option).



Once the records are collected to the report, an UPDATE statement is processed to change the *Reported* field from 0 to 1.



Cell values can be referenced in the command, such as in this case where the *Reported* flag is only raised for records older than the newest record in the report.