Exchange Recipe Data with a PLC

Overview

One of **XLReporter's** key features is that it can collect live data from a PLC directly into a report, with no database or storage medium. But, did you know that **XLReporter** can be used to store and load recipes of process parameters to and from the PLC?

This guide walks through the setup for recipe loading with **XLReporter** using a **Database Connector** to retrieve the recipe data. In our example, an Excel workbook is used as the database – this is an efficient solution because recipes can be easily accessed for editing by anyone with Excel. However, the configuration will also work against any relational database such as Microsoft Access or SQL Server.
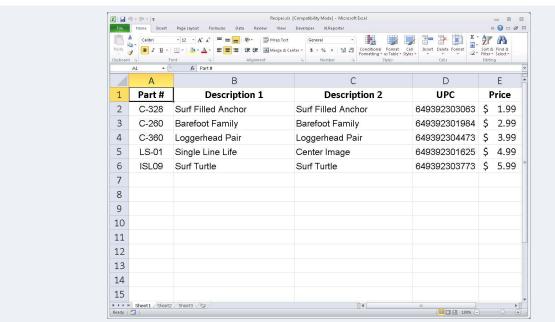
Working Example

To illustrate how **XLReporter** can exchange recipe data with a PLC let's consider the following example: A recipe needs to be loaded to the PLC based on the recipe ID set in the PLC. The recipe should be loaded when a bit in the PLC is set high.

To solve this, **XLReporter's** Scheduler monitors the PLC for the recipe download bit. When triggered, the recipe ID is read from the PLC, the recipe data is retrieved from the database and finally the recipe data back to the PLC to process.
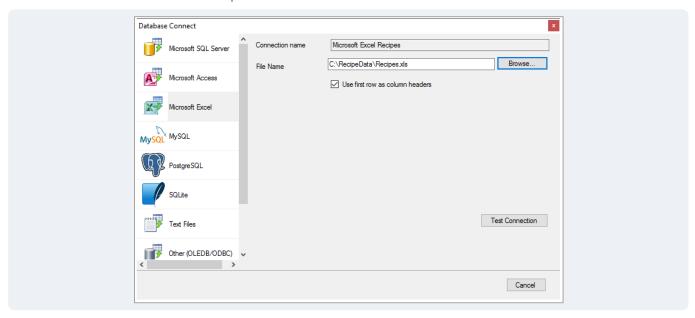
Recipe Database

The recipe database for this solution is stored in *C:\RecipeData\Recipes.xls*.

Data Connectors

There are two data connectors required for this: a **Real Time** connector that defines the connection to the processor and a **Database** connector to the recipe list.
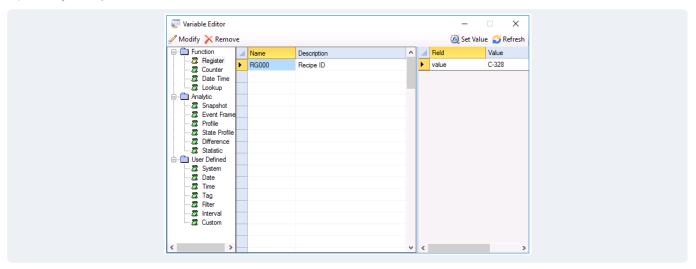

The **Database** connector is set to the **Microsoft Excel** workbook *Recipes.xls*.

Define a Function Variable

A **Function Register Variable** is used to hold the ID code of a recipe to look up in the workbook (or table for relational databases). To define,

Open **Project Explorer**, **Data** Tab, **Variables**.

- On the left, under **Function** select **Register**.
- Highlight a blank row and double-click to define a new **Variable**.
- Leave **Name** default, and set the **Description** to *Recipe ID*.
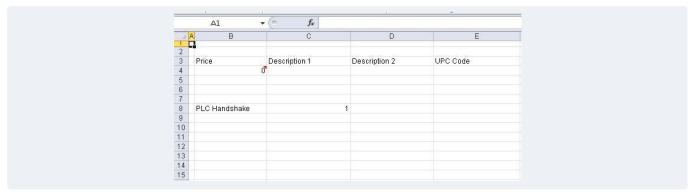
Design Report Template

Typically, a **Report Template** is designed to generate a report. In this case, the template processes a **Data Connection** to lookup the recipe, and then a **Management Connection** is used to output the data to the PLC.

From the **Home** tab of the **Project Explorer** select **Template, Studio**.

- Select **Template**, **New** to create a new template.
- Set **Name** to *Download*.
- Click **OK** to create a blank template.

Layout

Add some column headings to indicate which values will be collected, and where. This has no effect on the download but helps indicate the data being processed.

Next, add a cell that indicates a "handshake" value that will be written to the PLC after the download is performed *(cell C8 in the image above)*.
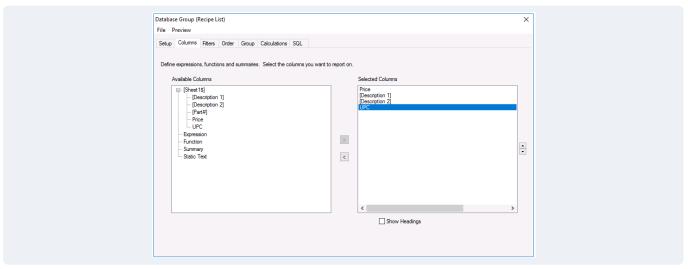
Data Connection

This section will vary based on the **Connector** used to collect recipe information. The purpose of this **Data Connection** is to filter on the value of the previously defined **Function Variable** to return the record that represents the recipe to be exported.
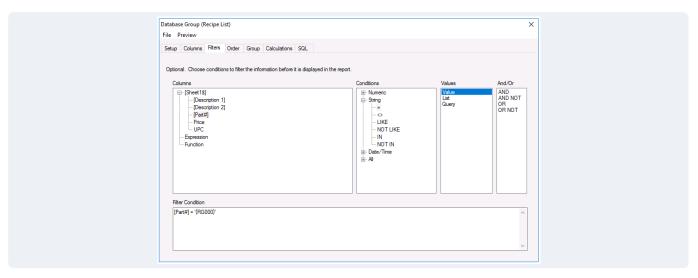
Open **Connect.** Under **Source,**

- For **Connector**, select the **Database Connector** defined.
- Set **Name** to *<add group …>*.
- Click the browse […] pushbutton to configure the database data group.
- For the **Type** select *Standard*.

Under the **Setup** tab, select *Sheet1$*.

Under the **Columns** tab, select the recipe parameter fields to be downloaded.

Under the **Filters** tab, select the **Column** that indicates the recipe ID.

For **Condition** select *Numeric, = Value*, and select the **Function Variable** that is set to the active recipe ID on the **Schedule** (*RG000* in this example).

Management Connections

Export Recipe

Open **Connect**, and select the **Manage** tab.

- Set the **Category** to *Data Export*.
- Set the **Type** to *To Tag List*.

Under **Apply To**,

- Set the **Start** to the top row where the recipe data appears e.g. *B4*.
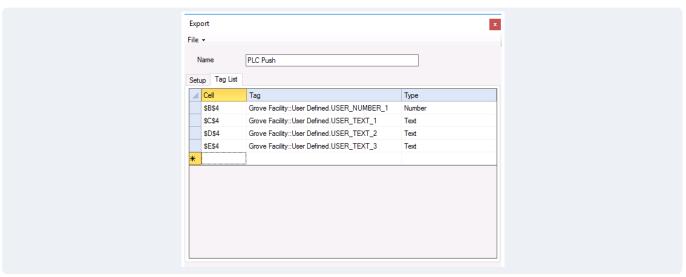- Set **Direction** to *None*.

In the **Setting** area,

- Click **Configuration** and open the browser.
- Select **New.**

In the **Export** dialog, under the **Setup** tab,

- Select the **Real Time Connector**.

Under the **Tag List** tab,

Map the cells to be exported with the parameter tags in the PLC.
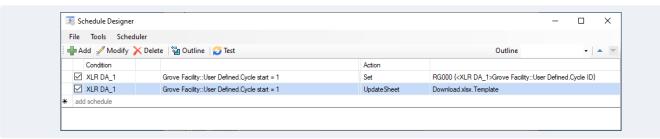
Select the **Type** based on the data type in the PLC.

Click **File**, **Save** to save changes.

Back in **Connections**, click **Add**.

When the recipe export is triggered from the **Schedule**, the data connection will first bring the recipe values into the report, then the **Management Connection** will run, exporting the cell values to the PLC.

Design a Schedule

A **Schedule** is required to monitor the PLC and trigger the export based on a tag value transition. Open the **Project Explorer,** and under the **Project** tab, select **Schedule Designer**.

There, highlight an empty row and select **Add.**

Under **Condition** expand **Event.**

- Select the **Real Time Values Connector.**
- Select the **Tag** and **Condition** that triggers the recipe downloading operation.

Under **Action** expand **Manage Variables.**

- Select *Set Value to a Variable.*
- Set **Variable** to the **Function Variable** defined for the recipe name (e.g. *RG000*).
- Open the browser **[…]** for **Value**.
- Collapse the **Calendar** category and select **Connector Item** category.
- Select the tag that contains the recipe ID.
- Click **OK**.

This **Action** will monitor the **Event Condition** and capture the recipe ID into a **Variable**. Next, **Add** an action that processes the recipe lookup and export.

- Set the same **Condition** used for the previous **Action**.
- Under **Action** expand **Produce Reports**.
- Select *Update Worksheet*.
- Set **Worksheet** to *Download.xlsx.Template*.