Use Third Party Excel Add-Ins with XLReporter

Overview

Many historians are packaged with an Excel add-in to display process data such as tag values and alarms in an Excel workbook. The add-in provides Historian functions that require input from cell locations to produce results in workbook "arrays". The operation is manually intense, difficult to automate, and the final workbooks cannot be easily distributed because of the specialized Historian functions.

XLReporter provides a layer of capability that can automate a workbook managed by an Excel add-in. It drives the input parameters to produce reports automatically, removes element that are specific to the Historian and then distribute the results in various formats to a number of devices such as server, printer and mobile.

Prerequisites

It is assumed that a workbook has already been developed using the Excel add-in. This will be used in the next section.

In addition, determine the add-in **Name** as displayed in Excel. To access the **Add-In Manager,** in Excel, open **File, Options, Add-Ins**.
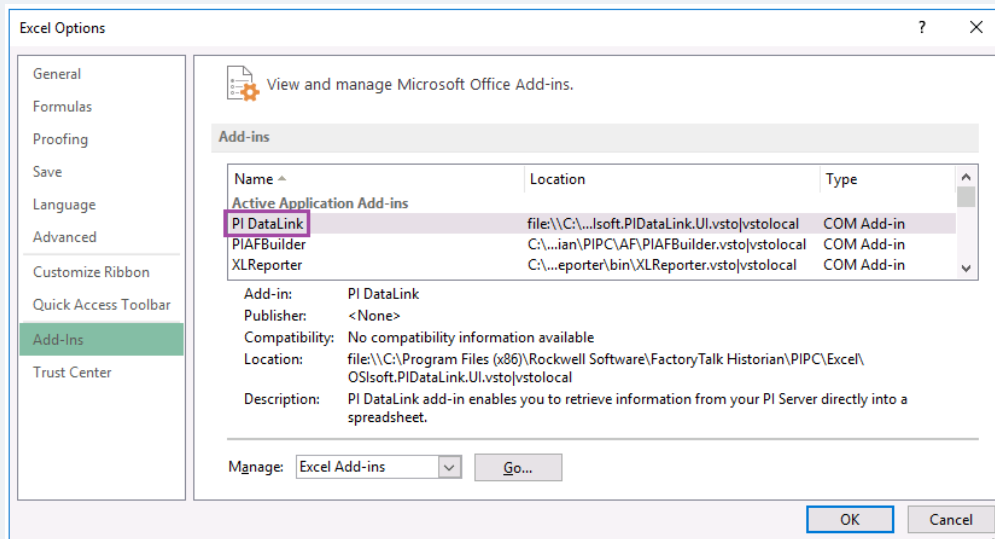
Template Design

The first step is to import the workbook that was developed using the add-in function into the **XLReporter** template studio in Excel.
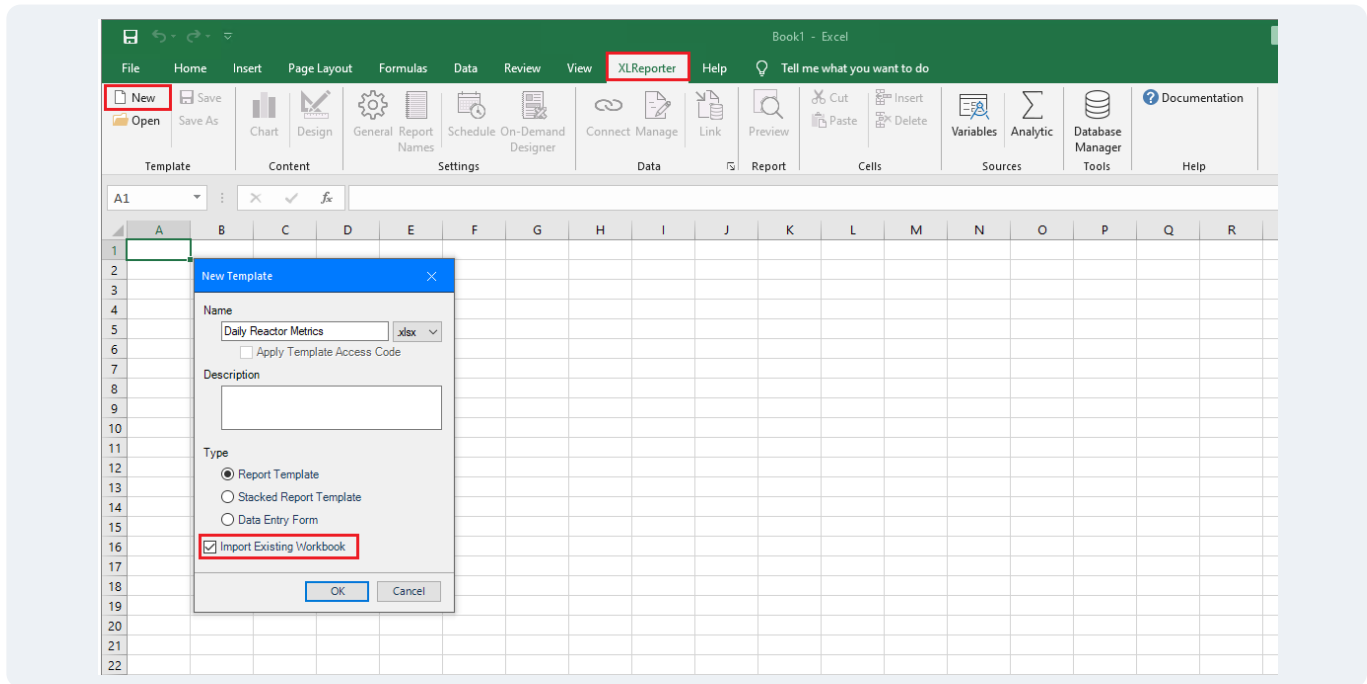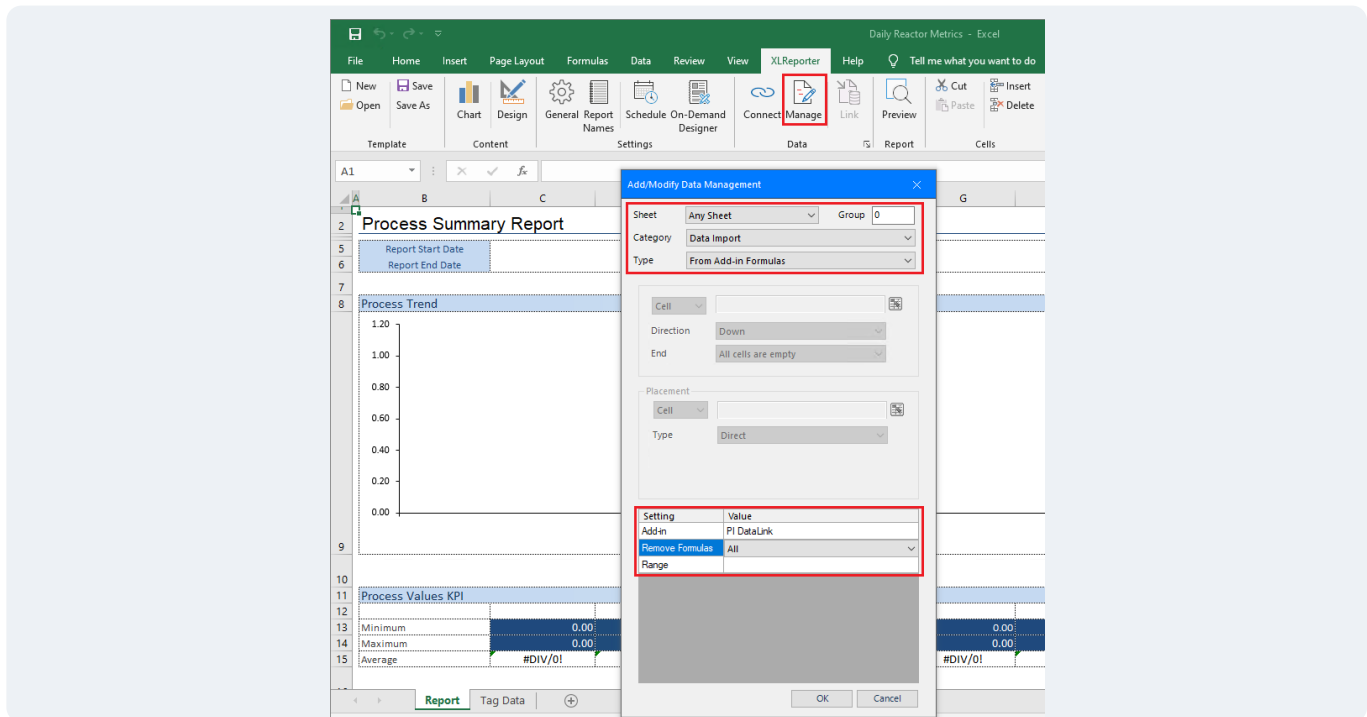

On the **XLReporter** tab, select **New Template**.

- Set the **Name**
- Check **Import Existing Workbook**
- Click **OK** select the workbook file to import.

Management Connection

This connection will remove the add-in formula from the workbook after the report has been updated.

From the **XLReporter** tab, in the **Data** section, select **Manage**.

- Click **Add** to add a new management connection
- Set **Category** to *Data Import*
- **Type** to *From Add-In Formulas*.
- Set **Add-In** to the **Name** of the add-in noted previously.
- Set **Remove Formulas** to *All* and leave **Range** as default (entire sheet).
- Click **OK** to add the connection to the list.
- Click **OK** again to save and return to the template

Manage Input Values

Usually, add-ins get their input from cell locations e.g., start and end time of the report. **XLReporter** can be configured to update these input cells automatically using an **Expression**.

From the **Data** section, select **Connect**.

- Click **Add**
- Set **Connector** to *Expressions*.
- Open the browser **[…]** for **Name** and select the appropriate date/time expression.
- Set the **Target Cell** to the cell which requires the expression.
- Click **OK** to add the connection to the list.
- Click **OK** again to save and return to the template

Report Names

When a report is produced from the template, it will be named according to the **Report Names** setting. The name can be a combination of fixed text, date/time values and tag values.
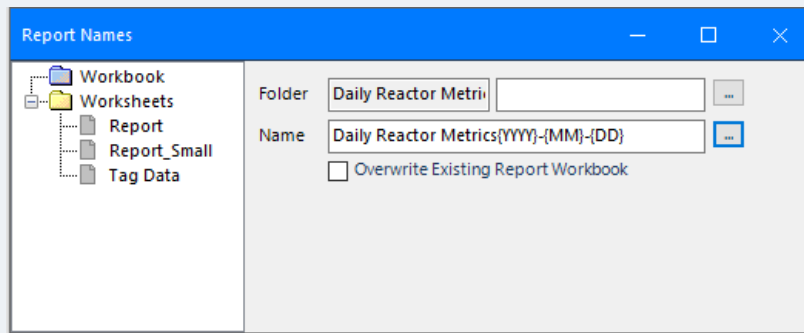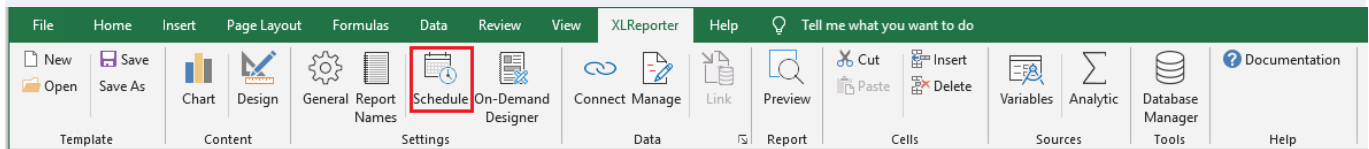

From the **XLReporter** tab, in the Settings **section**, open **Report Names**.

- Select the Workbook folder on the left
- Use the browser **[...]** for **Name** to set a date and time expression that will determine the name of the file when the report is updated.
  Select an expression that matches the time period of the report. If the report contains data relative to the current day, for example, select *Year-Month-Day*.

Schedule Design

The **Schedule Designer** is used to configure a schedule to process the template automatically and in the background. Combined with the **Report Names** settings, this allows the reports to be saved to unique files every month, day, hour, and so on.

From the **XLReporter** tab, in the **Settings** section, click **Schedule**.
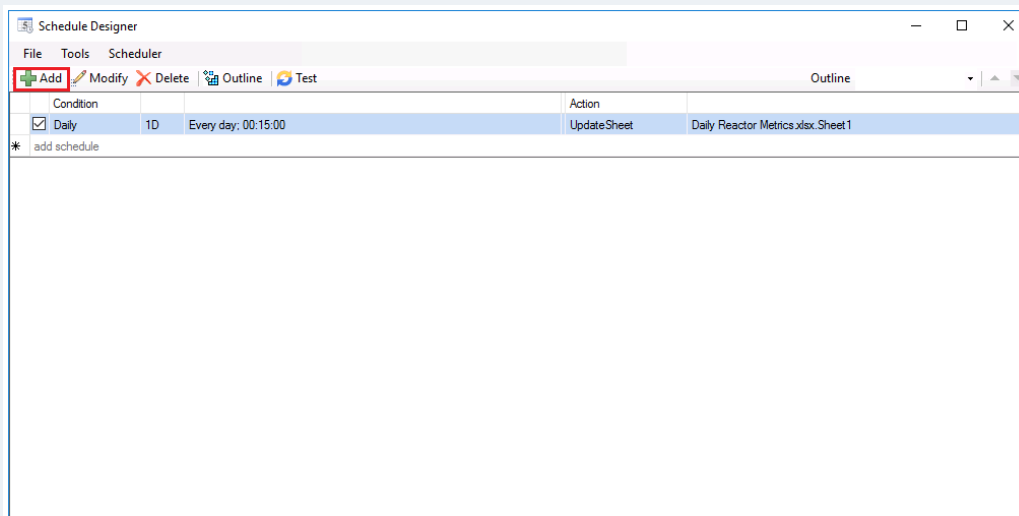
Distribution

A template based on an add-in template is treated exactly the same as any other template. This means it can be viewed and updated from **XLReporter Web Client**, the **Windows Client,** or published automatically via email, so that anyone in the organization can view it, even without excel, or the add-in installed!