2012

Performed Tasks

UI for Rockwell.Logging.Tag Type

General status of performed and completed tasks in the project

Entegreat
Magaly V. Vargas
2/15/2012

Contents

Performed and completed tasks. 2

Make the forms pops up when create and edit a Rockwell.Logging.Tag Item. 2

Add functionality to the UI in order to create or edit the item. 2

Create the UI to gather the Fault Logging configuration data. 3

Create and assign Tag Reference Properties. 4

Summary of completed task 6

Pending tasks 6

Main found issues 6

Installing the UI 6

Performed and completed tasks.

I received a base application with two forms to create and edit Items of the type Rockwell.Logging.Tag .

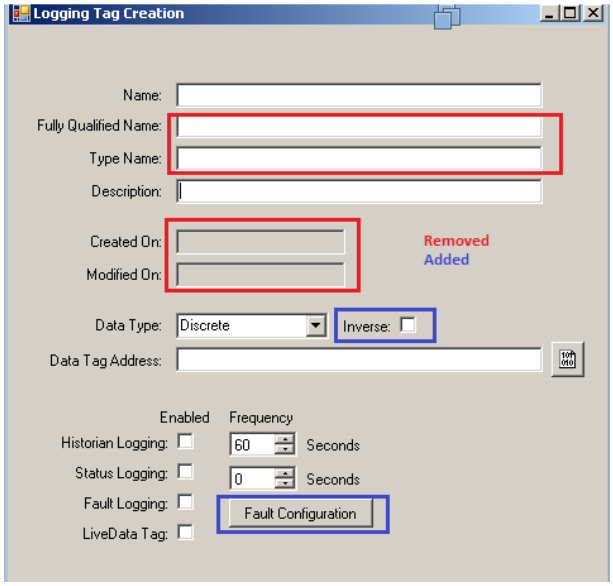
The tasks performed were:

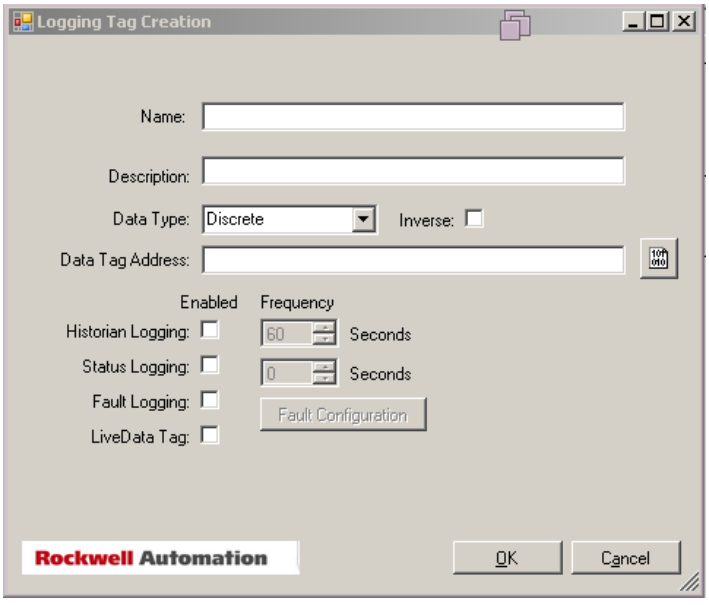
Make the forms pops up when create and edit a Rockwell.Logging.Tag Item.

For this was necessary to modify the assembly to associate the UI with the type. Then, the resulting dll must be copied to the wwwroot/Incuity/download folder. Restart the IIS

Add functionality to the UI in order to create or edit the item.

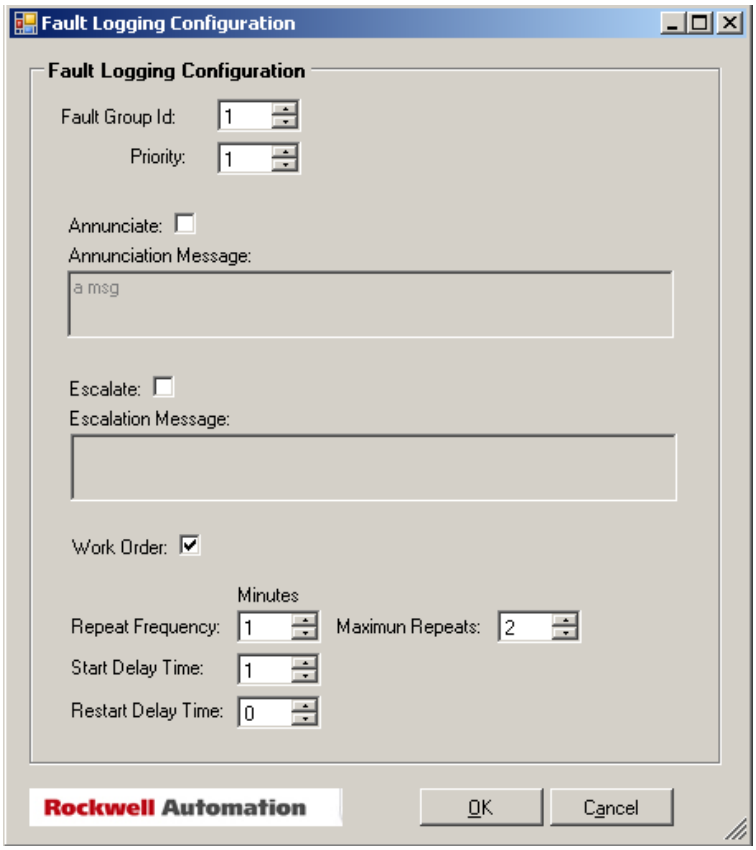
- Creates the code behind for the buttons of the control, load the types in the combobox, enable or disable controls according with the user selections.
- Add the logic to create or edit an Item.
- Add controls for missing properties and removed those that were not necessaries in the creation form (after consulting Robert Malthaner)





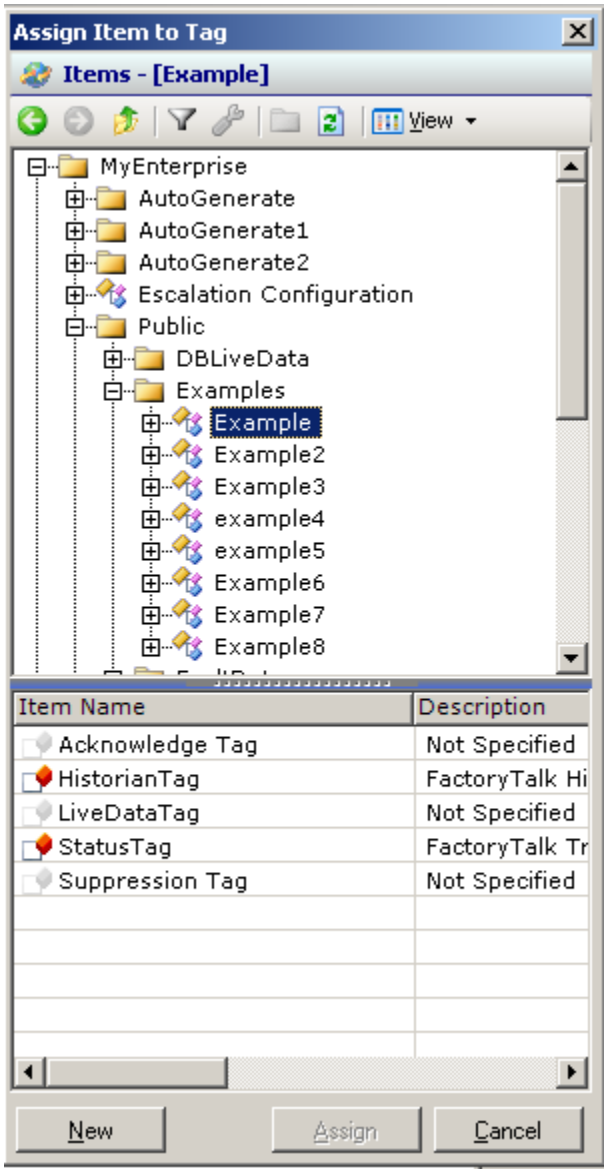
Create the UI to gather the Fault Logging configuration data.

- It was created a new form called FaultSettings.cs, with the fields to complete the fault logging information:



Create and assign Tag Reference Properties.

For this task it was necessary to create a new form called **TagModelBrowserForm.cs** with a customized control called **BrowserCtrl**.

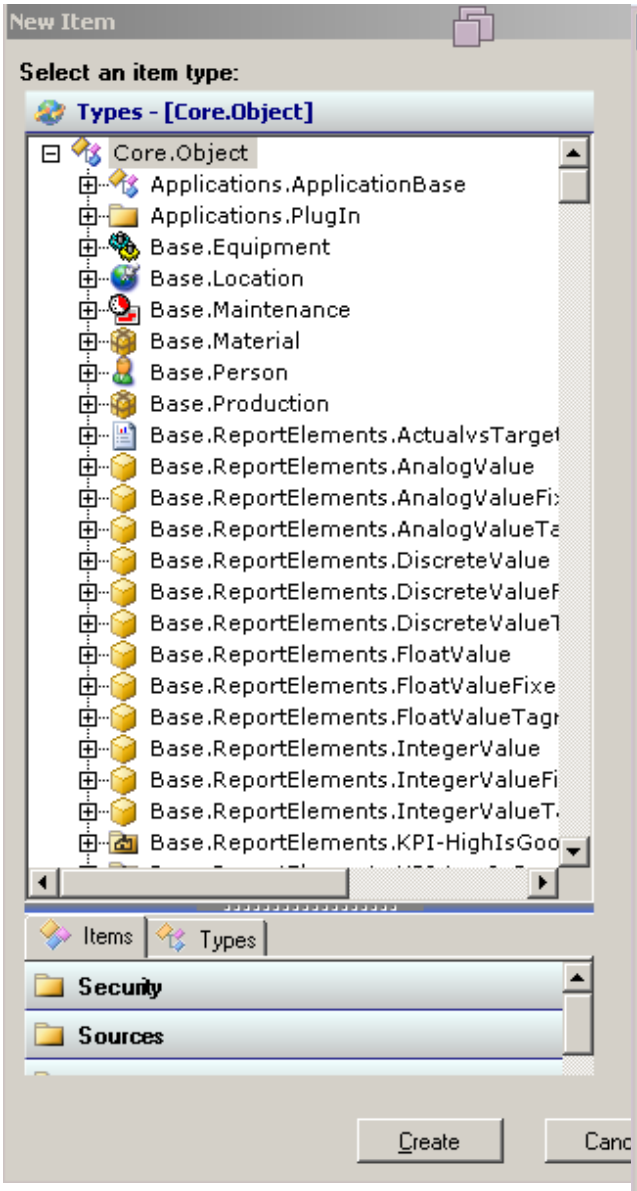


It took a long, find out where was this control because we don't even its name.

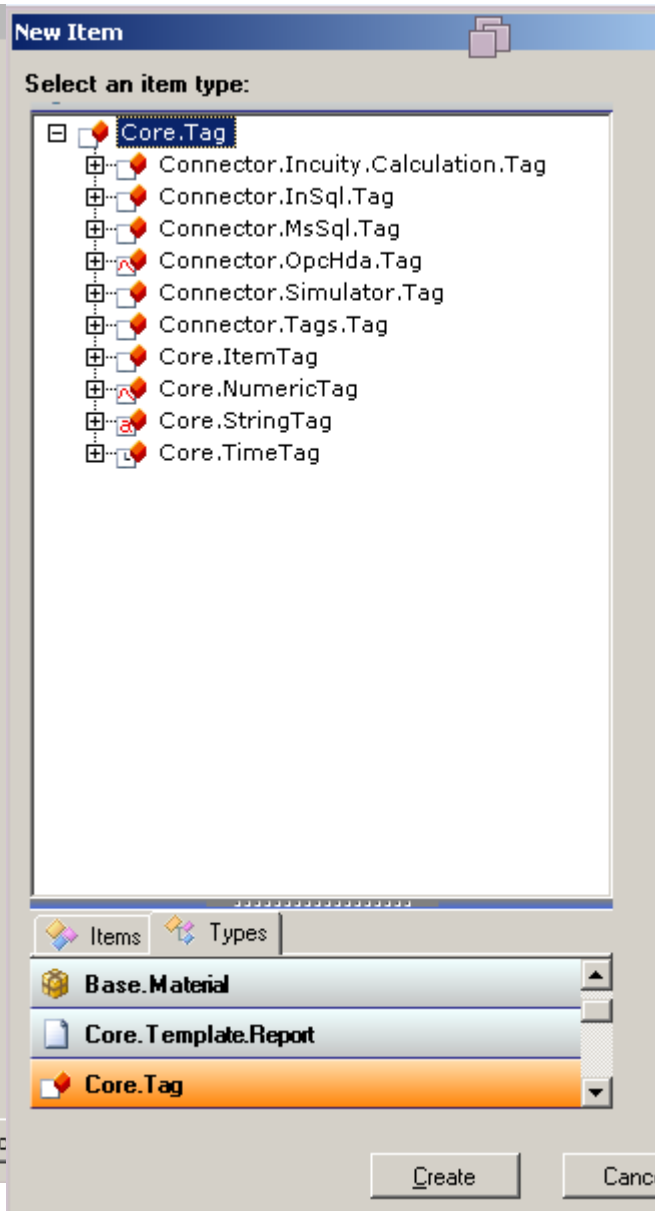
For this form called TagTreeView.cs it was created three new buttons ***New***, ***Assign*** **and** ***Cancel***.

- The **Assign** button assigns an existing reference to the tag property.
- The functionality for the **new** button is to show a new ***BrowserCtrl*** showing the elements in Mode.Types instead of Instances and filtered by Core.Tag.
- The form was created and the ***BrowserCtrl*** added. But I cannot configure it to show only the core.tag types.

This form looks like this:



Selecting the filter manually we can get this look:



I couldn't do it by code. I tried with different properties without result.

Summary of completed task

- Show the new UI when create or edit a Rockwell.Logging.Tag Item.
- Add functionality to the UI in order to create or edit the item.
- Create the UI to gather the Fault Logging configuration data.
- Create and assigns Tag Reference Properties.
- So, It is possible to create and edit Rockwell.Logging.Tag Item with all its properties

Pending tasks

- Set the node and property that must be selected and expanded in the **BrowserCtrl** when the TagBrowserWindows pops up.
- Filter the types that are shown in the TagTreeViewForm by Core.Tag.
- Enable or disable the Core.tags items shown in the TagTreeView according to the type that accept the tag property.

Main found issues

- There is a lack of documentation about the Controls used by the app.
- Took a long time to find info about BrowserCtrl.
- The labs barely used the BrowserCtrl.
- The designer of these customized controls only work for Visual Studio 2008, so I had to create the controls only by code.

Installing the UI

- Build the LoggingTagUI.WinUI.sln application in release mode
- picks the LoggingTag.WinUI.dll from the release output folder
- Copy the dll to C:\Inetpub\wwwroot\Incuity\Download.
- From the **Start** menu, select run and execute iisreset
- Now, go to VantagePoint and create or edit an Item of type Rockwell.Logging.Tag
- The new UI shall appear.