FlowPanel Dynamic Symbols Example

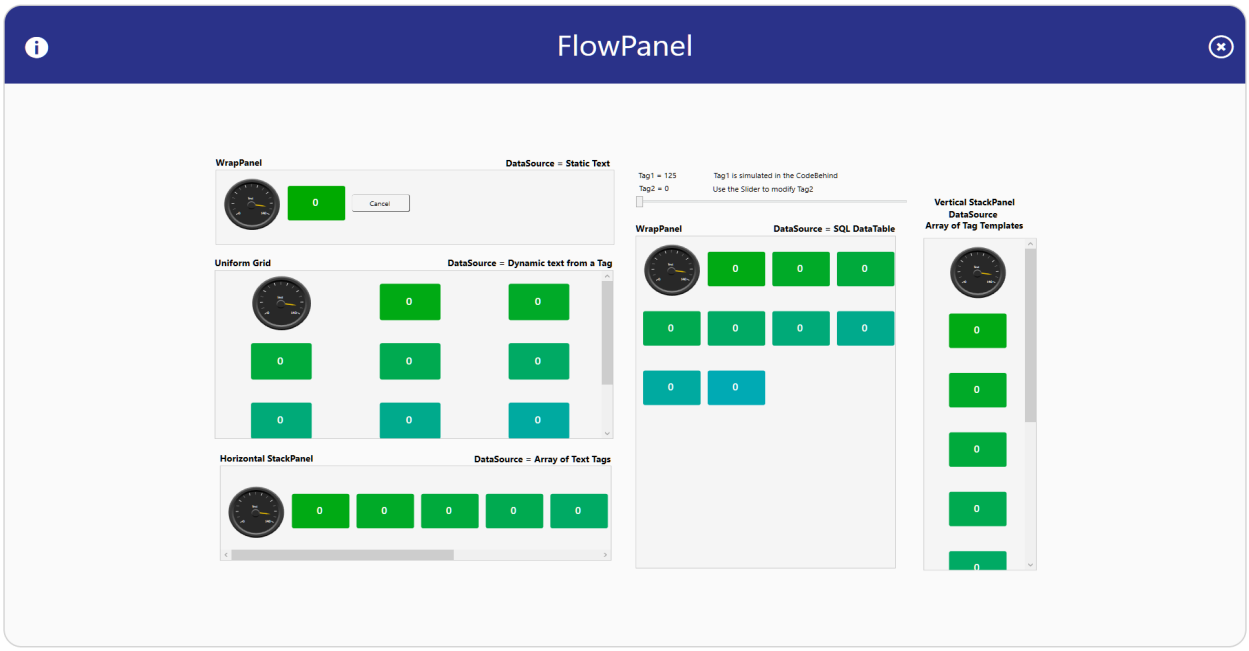
Dynamically load symbols in panels.

Download the solution FlowPanel Dynamic Symbols Example.dbsln.

This example demonstrates how to use the **TFlowPanel** component.

Summary

This example demonstrates how to use the **TFlowPanel** component, which allows for the dynamic creation of Symbols at runtime by querying the quantity and definitions of the symbols from databases or real-time tags.



Technical Information

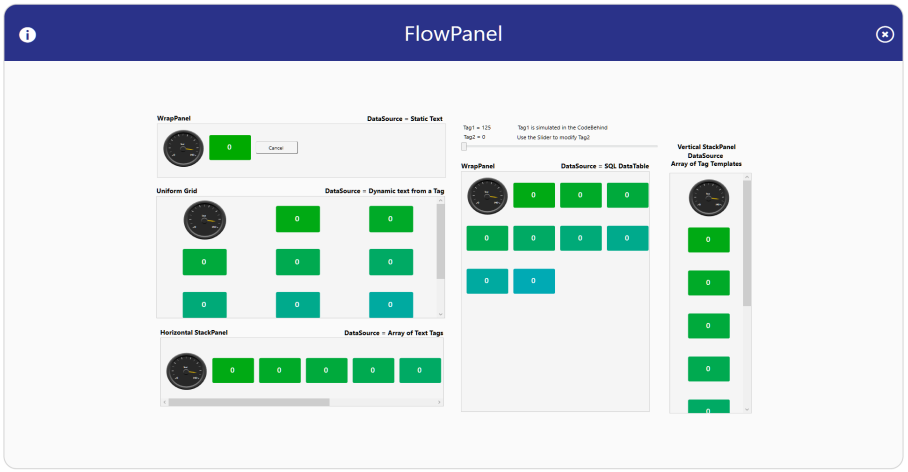
The example demonstrates the **TFlowPanel** control, dynamically creating **WrapPanels** as well as horizontal and vertical **StackPanels**.

The panel contents are dynamically generated from various data sources. The example showcases 5 scenarios:

1. Panel configured with a fixed text;
2. Panel configured with text from a tag;
3. Panel configured from an array of text tags;
4. Panel configured from a DataTable tag;
5. Panel configured from an array of UserType tags.

1. Data Source = Text / Panel Type = WrapPanel.

Items for 'Text' Source can be filed by writing in this field or by clicking in the icons in the bottom left on the settings page.



2. Data Source = Text Tag / Panel Type = UniformGrid.

The code below sets the value of the text-type tag (@Tag.tagText) that is used as a source for the FlowPanel.

```
// Task.ServerStartup
Tag.tagText.Value = "#Element=HMI/Circular_Gauge/Tacho3 #Value=tag.tag1 #Max=100 ;";
string str = "";
for (int i = 1; i < 10; i++) {
        string add = i.ToString();
        str += "#Element=HMI/Miscellaneous/ColorLabel #Value=tag.tag2 #Color=\"#FF00AA" + (i * 20).ToString
("X2") + "\" ; ";
}
@Tag.tagText.Value += str;
```

3. Data Source = Array / Panel Type = StackPanelHorizontal

The code below sets the values of an array of text-type tags (@Tag.tagTextArray[ ]) that are used as the source for the FlowPanel.

```
//Task.ServerStartup
string[] strs = @Tag.tagText.Value.Split(new char[] { ';' }, StringSplitOptions.RemoveEmptyEntries);
for (int j = 0; j < Math.Min(@Tag.tagTextArray.Length, strs.Length); j++)
        @Tag.tagTextArray[j] = strs[j].Trim();
```

4. Data Source = Object / Panel Type = WrapPanel

The code below sets the values of a DataTable tag (@Tag.tagTable) that is used as the source for the FlowPanel.

```
//Task.ServerStartup
DataTable table1 = new DataTable();
table1.Columns.Add("Element", typeof (string));
table1.Columns.Add("Value", typeof (string));
table1.Columns.Add("Max", typeof (int));
table1.Columns.Add("Color", typeof (string));
DataRow row1 = table1.NewRow();
row1["Element"] = "HMI/Circular_Gauge/Tacho3";
row1["Value"] = "Tag.Tag1";
row1["Max"] = 100;
table1.Rows.Add(row1);

for (int j = 1; j < 10; j++) {
        DataRow row2 = table1.NewRow();
        row2["Element"] = "HMI/Miscellaneous/ColorLabel";
        row2["Value"] = "Tag.Tag2";
        row2["Max"] = 100;
        row2["Color"] = "\"#FF00AA" + (j * 20).ToString("X2") + "\"";
        table1.Rows.Add(row2);
}
@Tag.tagTable.Value = table1;
```

5. Data Source = Object / Panel Type = WrapPanel

The code below sets the values of a UserType tag (@Tag.dt) that is used as the source for the FlowPanel.

```
//Task.ServerStartup
@Tag.dt[0].Element = "HMI/Circular_Gauge/Tacho3";
@Tag.dt[0].Value = "tag.tag1";
@Tag.dt[0].Max = 140;
for (int j = 1; j < 10; j++) {
        @Tag.dt[j].Element = "HMI/Miscellaneous/ColorLabel";
        @Tag.dt[j].Value = "tag.tag2";
        @Tag.dt[j].Color = "\"#FF00AA" + (j * 20).ToString("X2") + "\"";
}
```

In this section...
