Multiple Display Instances | New Popup in .NET and HTML5

How to Use

This feature allows you to have multiple instances of the same display opened simultaneously.

The specs for the computer used for testing are listed below:

- Product version *1.9*.
- *TWebServer* or any other WebServer (like IIS)

Besides allowing you to have the same popup display opened multiple times, they can also be customized using a list of user-defined parameters that are inputted on the method.

Creating a Popup

Let's create an example where we have multiple lines in our system. Each of these lines will be accessed and displayed using just one display.

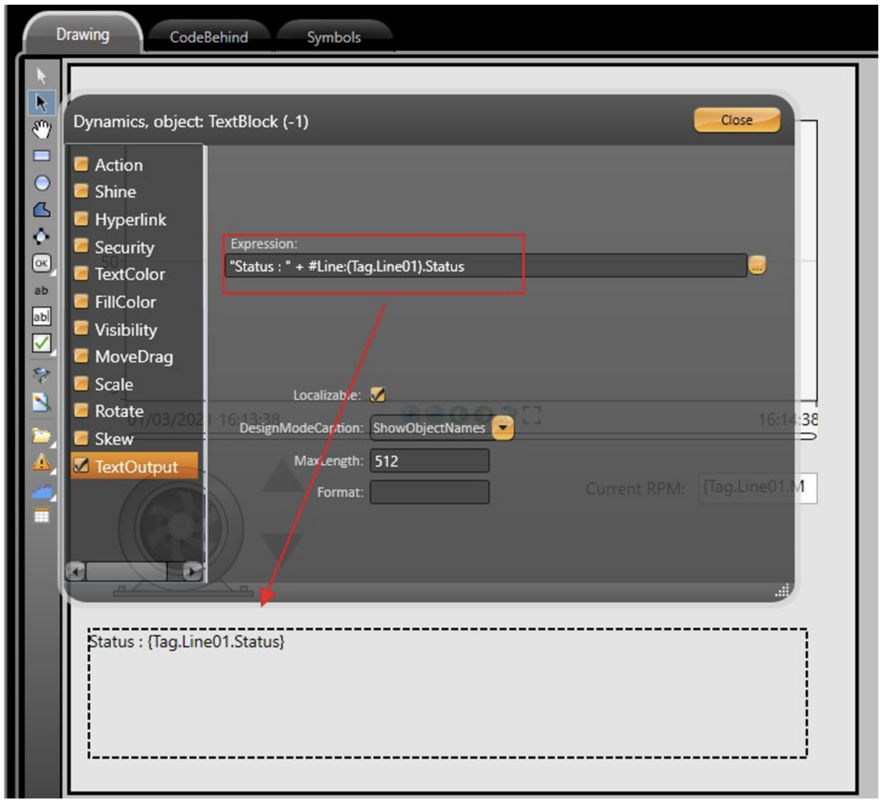
For each line, there is an associated Tag (of a custom Template datatype) which contains all of its details.

Line01	Line02	Line03
Status	Status	Status
MotorRPM	MotorRPM	MotorRPM
isEnable	isEnable	isEnable
lastError	lastError	lastError

To create a customizable popup, you need to follow a specific syntax when defining the elements/components expressions.

```
#<PropertyName>:<TagName>
```

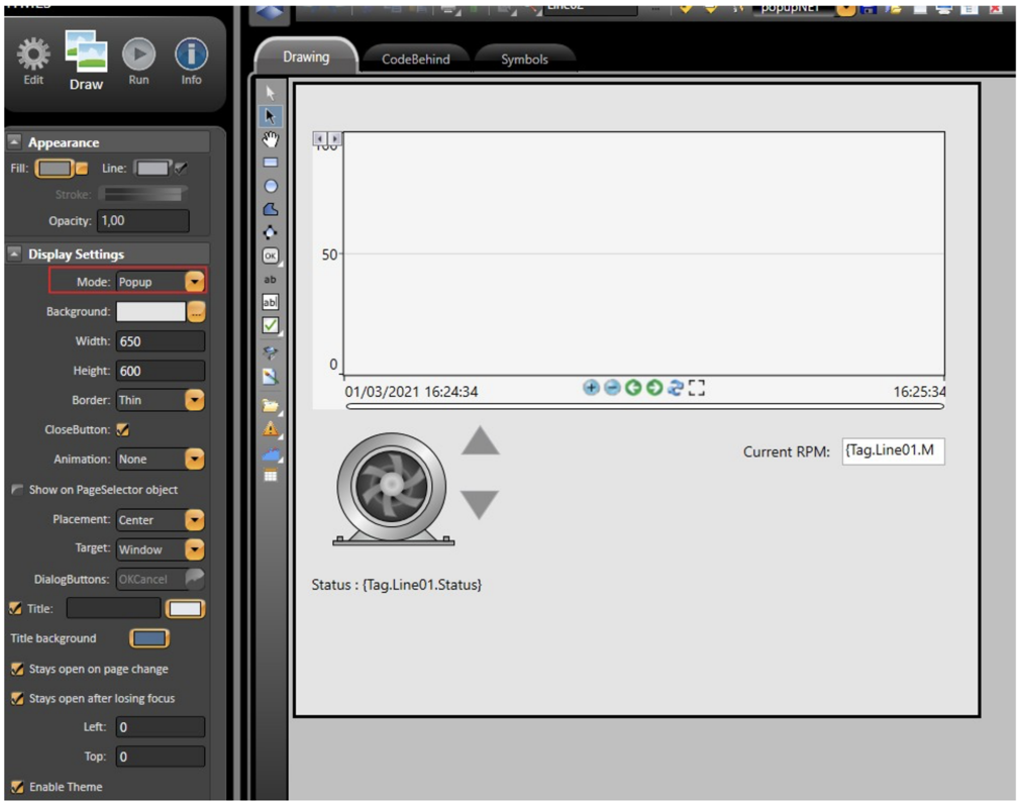
This syntax will create an exposed label for the element property. This makes it easier to map any linked tags to it.



This syntax can be used in all available components. It can also be retrieved in the CodeBehind Environment with the code below.

```
var label = this.CurrentDisplay.GetLabel("<LabelName>");
// <LabelName> is an user-defined input parameter
```

It is important to make sure that the display mode is configured as a **popup**.



Appearance

Fill:
Line:
Stroke:
Opacity: 1,00

Display Settings

Mode: Popup
Background:
Width: 650
Height: 600
Border: Thin
CloseButton: ✓
Animation: None
Show on PageSelector object
Placement: Center
Target: Window
DialogButtons: OKCancel
✓ Title:
Title background
✓ Stays open on page change
✓ Stays open after losing focus
Left: 0
Top: 0
✓ Enable Theme

Runtime Methods

The methods available for the NewPopup feature are listed below.

Display.<DsplayName>.NewPopup

This method is available for ***.NET*** displays **only**. Its syntax is described below:

```
/// <summary>
/// Begin Open new popup method
/// It should be called on ServerStartup task only
/// </summary>
/// <param name="items">LabelList[label1=tag1;label2=tag2],
/// Left, Top, Width, Height</param>
/// <returns>Always an empty string</returns> @Display.<DisplayName>.NewPopup(object[] items);
```

The items array will contain the LabelName and its assigned runtime object (Tag), which will replace the placeholder configuration on the display.

It is also possible to use fixed parameters in the LabelList. In this example, we will have our title be dynamic but unrelated to any Tags.

An example of this method is:

```
// Mapping for Tag.Line01 @Display.MyPopup.NewPopup("#Line=Line01;#Title='Title for Line01 Display'");
// Mapping for Tag.Line02 @Display.MyPopup.NewPopup("#Line=Line02;#Title='Title for Line02 Display'");

where [MyPopup] is the display name of this example.
```

To get the *Title* label from the LabelList and use it to modify the Display Title, the following code must be added to the *DisplayOpening* method:

```
public void DisplayOpening()
{
this.CurrentDisplay.SetTitle( this.CurrentDisplay.GetLabel("#Title") );
}
```

Client.NewPopup

This method is available for both ***.NET*** and ***HTML5*** displays. Its syntax is described below:

```
/// <summary>
/// Begin Open new popup method
/// It should be called on ServerStartup task only
/// </summary>
/// <param name="displayName">Display Name in Project</param>
/// <param name="items">LabelList[label1=tag1;label2=tag2],
/// Left, Top, Width, Height</param>
/// <returns>Always an empty string</returns> @Client.NewPopup(string displayName, object[] items);
```

The items array will contain the LabelName and its assigned runtime object (Tag), which will replace the placeholder configuration on the display.

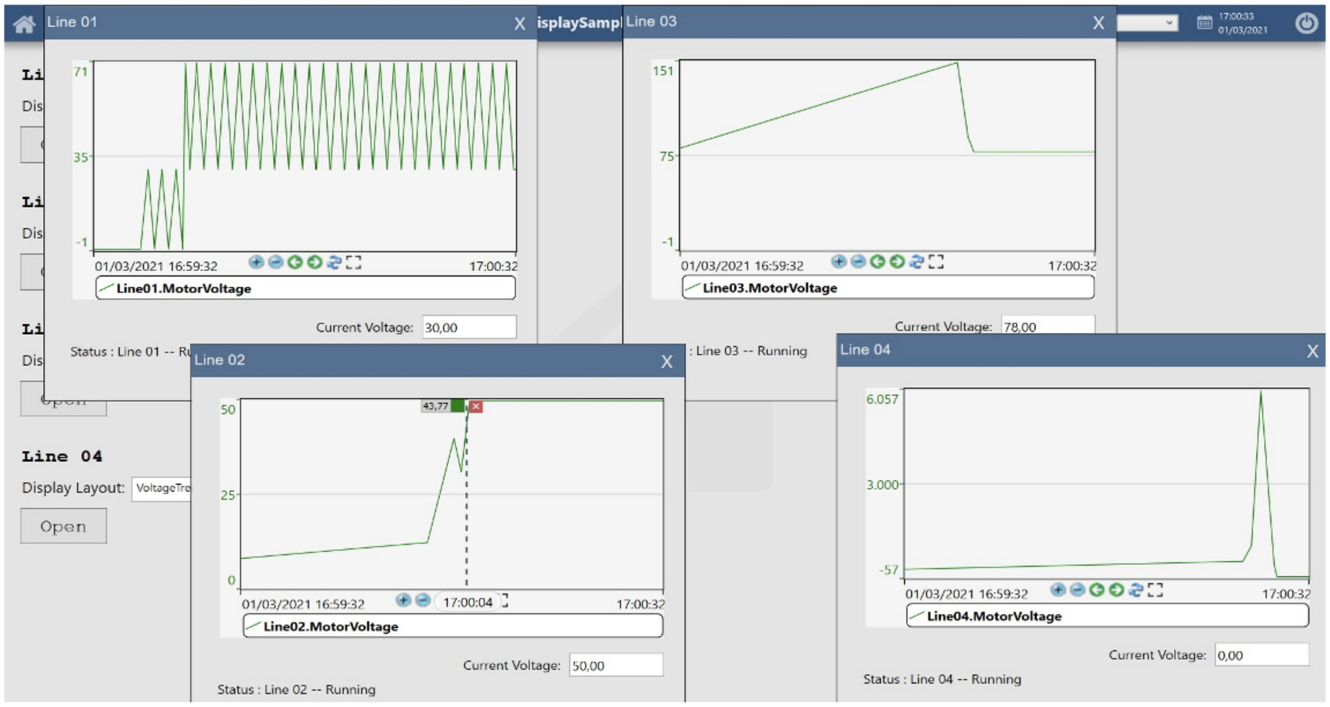
It is also possible to use fixed parameters in the LabelList. In this example, we will have our title be dynamic but unrelated to any Tags.

An example of this method is:

```
// Mapping for Tag.Line03 @Client.NewPopup("MyPopup","#Line=Line03;#Title=’Line 03 -- Title’");
// Mapping for Tag.Line04 @Client.NewPopup("MyPopup","#Line=Line04;#Title=’Line 04 -- Title’");

where [MyPopup] is the display name of this example.
```

In the end, the project’s behavior should be the same despite the method used.



Current Voltage: 30,00
Status : Line 01 -- Ru
Current Voltage: 78,00
: Line 03 -- Running
Current Voltage: 50,00
Status : Line 02 -- Running
Current Voltage: 0,00
Status : Line 04 -- Running