



Test Automation FX

UI test automation for .NET developers

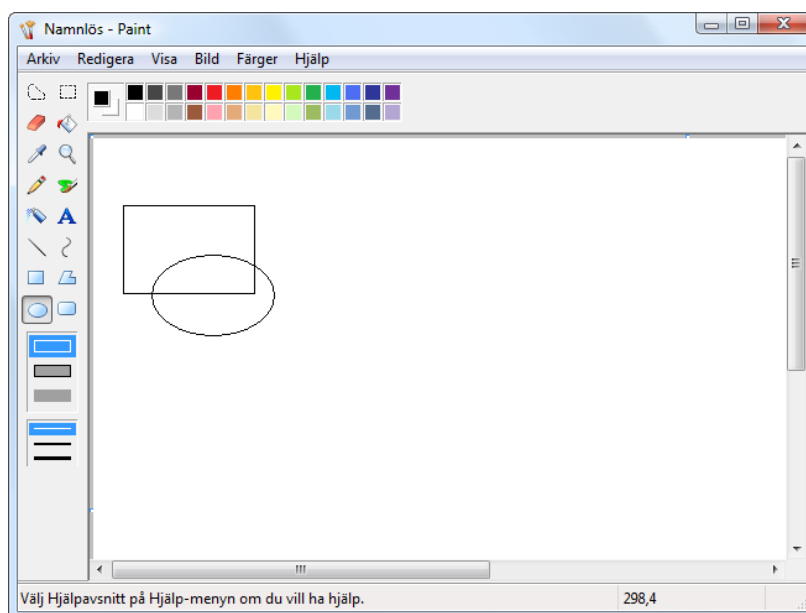
Control Parts

The recording support in Test Automation FX has been developed to produce readable code with recognizable names. There are times however when objects or areas on the screen cannot be recognized for what they are. When this happens you can solve this problem by using a feature in TAFX called **Control Parts**.

Example

If we for instance use the **Paint** application in Windows and record a test where we perform the following actions:

- Click the **rectangle tool**.
- Draw a **rectangle**.
- Click the **ellipse tool**.
- Draw an **ellipse**.



Then the code is recorded as:

```
[UITest ()]
public void Test1 ()
{
    ToolWindow1.Click(19, 167);
    Client.Drag(27, 57, 134, 129);
    ToolWindow1.Click(15, 194);
    Client.Drag(50, 97, 150, 163);
}
```

This is all true, but we can see one problem. The circle tool and rectangle tool are both



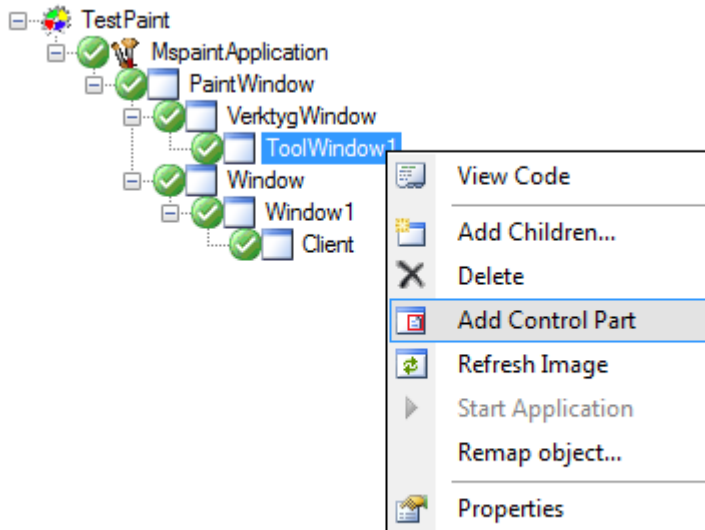


Test Automation FX

UI test automation for .NET developers

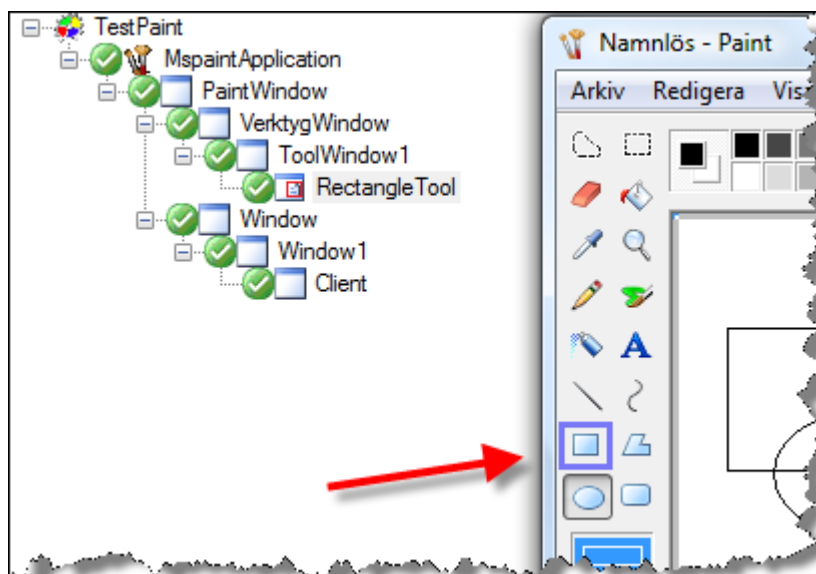
identified as ToolWindow1. This is due to TAFX not being able to get any additional information about the content of the ToolWindow.

To help TAFX we can add a ControlPart to the ToolWindow. We do this by selecting the **Add ControlPart** context menu in the 'UIMap designer' for the ToolWindow.



When you have added a ControlPart you can move it and resize it on the screen with the mouse (make sure that the **ControlPart** is selected in the 'UIMap designer' and the **Highlight object checkbox** is checked).

Position it over the **rectangle tool** and rename it 'RectangleTool' in the 'Properties' window.



Add another Control Part, position it over the **ellipse tool** and rename it to 'EllipseTool'. Now, if we make another recording with the same actions, we get the following much more precise code:





Test Automation FX

UI test automation for .NET developers

```
[UITest ()]  
public void Test2 ()  
{  
    RectangleTool.Click(12, 6);  
    Client.Drag(26, 59, 135, 126);  
    EllipseTool.Click(9, 13);  
    Client.Drag(59, 102, 159, 162);  
}
```

