

Animating an UpdatePanel Control

Christian Wenz

Overview

The Animation control in the ASP.NET AJAX Control Toolkit is not just a control but a whole framework to add animations to a control. For the contents of an **UpdatePanel**, a special extender exists that relies heavily on the animation framework: **UpdatePanelAnimation**. This tutorial shows how to set up such an animation for an **UpdatePanel**.

Steps

The first step is—as usual—to include the **ScriptManager** in the page so that the ASP.NET AJAX library is loaded and the Control Toolkit can be used:

```
<asp:ScriptManager ID="asm" runat="server" />
```

The animation in this scenario will be applied to an ASP.NET **Wizard** web control residing in an **UpdatePanel**. Three (arbitrary) steps provide enough options to trigger postbacks:

```
<asp:UpdatePanel ID="UpdatePanel1" runat="server">
  <ContentTemplate>
    <asp:Wizard ID="Wizard1" runat="server">
      <WizardSteps>
        <asp:WizardStep runat="server" StepType="Start"
          Title="Ready!">
        </asp:WizardStep>
        <asp:WizardStep runat="server" StepType="Step"
          Title="Set!">
        </asp:WizardStep>
        <asp:WizardStep runat="server" StepType="Finish"
          Title="Go!">
        </asp:WizardStep>
      </WizardSteps>
    </asp:Wizard>
  </ContentTemplate>
</asp:UpdatePanel>
```

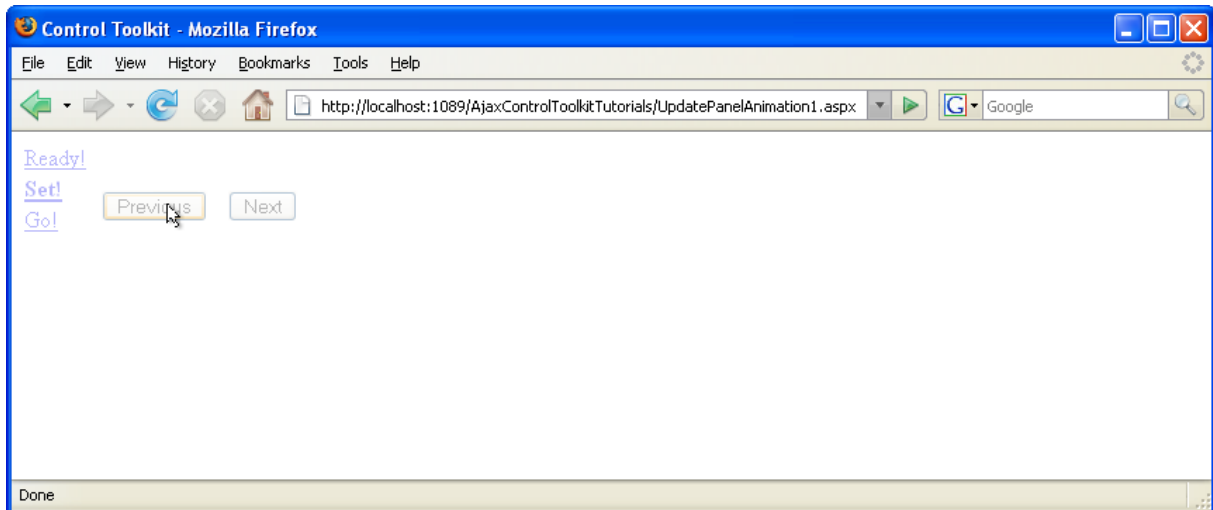
The markup necessary for the **UpdatePanelAnimationExtender** control is quite similar to the markup used for the **AnimationExtender**. In the **TargetControlID** attribute we provide the **ID** of the **UpdatePanel** to animate; within the **UpdatePanelAnimationExtender** control, the **<Animations>** element holds XML markup for the animation(s). However there is one difference: The amount of events and event handlers is limited in comparison to **AnimationExtender**. For **UpdatePanels**, only two of them exist:

- **<OnUpdated>**—when the UpdatePanel has been updated
- **<OnUpdating>**—when the UpdatePanel starts updating

In this scenario, the new content of the **UpdatePanel** (after the postback) shall fade in. This is the necessary markup for that:

```
<ajaxToolkit:UpdatePanelAnimationExtender ID="upae" runat="server"
  TargetControlID="UpdatePanel1">
  <Animations>
    <OnUpdated>
      <FadeIn Duration="1.0" Fps="24" />
    </OnUpdated>
  </Animations>
</ajaxToolkit:UpdatePanelAnimationExtender>
```

Now whenever a postback occurs within the UpdatePanel, the new contents of the panel fade in smoothly.



The next wizard step is fading in