

Adding Animation to a 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. This tutorial shows how to set up such an animation.

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 a panel of text which looks like this:

```
<asp:Panel ID="panelShadow" runat="server" CssClass="panelClass">
  ASP.NET AJAX is a free framework for quickly creating a new
  generation of more efficient,
  more interactive and highly-personalized Web experiences that
  work across all the
  most popular browsers.<br />
  ASP.NET AJAX is a free framework for quickly creating a new
  generation of more efficient,
  more interactive and highly-personalized Web experiences that
  work across all the
  most popular browsers.<br />
  ASP.NET AJAX is a free framework for quickly creating a new
  generation of more efficient,
  more interactive and highly-personalized Web experiences that
  work across all the
  most popular browsers.<br />
</asp:Panel>
```

The associated CSS class for the panel defines a background color and a width:

```
<style type="text/css">
  .panelClass {background-color: lime; width: 300px;}
</style>
```

Next up, we need the **AnimationExtender**. After providing an **ID** and the usual **runat="server"**, the **TargetControlID** attribute must be set to the control to animate—in our case, the panel:

```
<ajaxToolkit:AnimationExtender ID="ae" runat="server"
  TargetControlID="Panel1">
```

The whole animation is applied declaratively, using an XML syntax, unfortunately currently not fully supported by Visual Studio's IntelliSense. The root node is **<Animations>**; within this node, several events are allowed which determine when the animation(s) take(s) place:

- **OnClick** (mouse click)
- **OnHoverOut** (when the mouse leaves a control)
- **OnHoverOver** (when the mouse hovers over a control, stopping the **OnHoverOut** animation)
- **OnLoad** (when the page has been loaded)
- **OnMouseOut** (when the mouse leaves a control)
- **OnMouseOver** (when the mouse hovers over a control, not stopping the **OnMouseOut** animation)

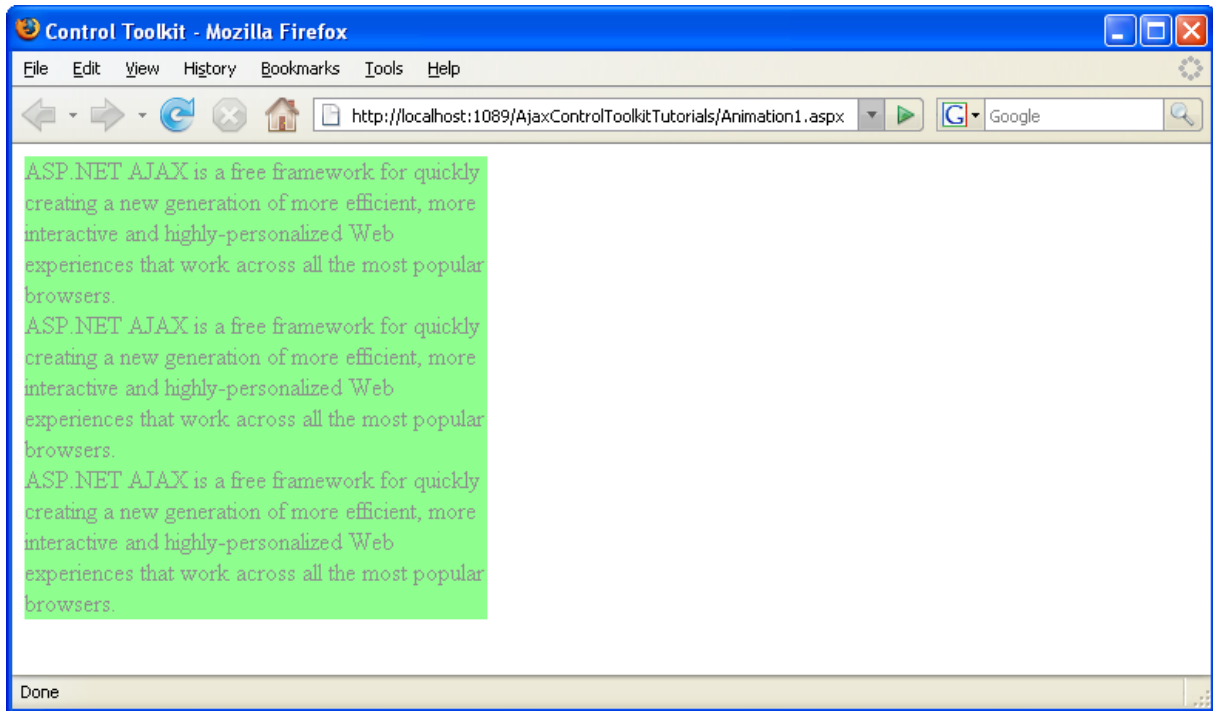
The framework comes with a set of animations, each one represented by its own XML element. Here is a selection:

- **<Color>** (changing a color)
- **<FadeIn>** (fading in)
- **<FadeOut>** (fading out)
- **<Property>** (changing a control's property)
- **<Pulse>** (pulsating)
- **<Resize>** (changing the size)
- **<Scale>** (proportionally changing the size)

In this example, the panel shall fade out. The animation shall take 1.5 seconds (**Duration** attribute), displaying 24 frames (animation steps) per second (**Fps** attribute). Here is the complete markup for the **AnimationExtender** control:

```
<ajaxToolkit:AnimationExtender ID="ae" runat="server"
  TargetControlID="Panel1">
  <Animations>
    <OnLoad>
      <FadeOut Duration="1.5" Fps="24" />
    </OnLoad>
  </Animations>
</ajaxToolkit:AnimationExtender>
```

When you run this script, the panel is displayed and fades out in one and a half seconds.



The panel is fading out