

Executing Animations Using Client-Side Code

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. The animation execution may also be triggered using custom client-side JavaScript code.

Steps

First of all, include the **ScriptManager** in the page; then, the ASP.NET AJAX library is loaded, making it possible to use the Control Toolkit:

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

The animation 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.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>
```

In the associated CSS class for the panel, define a nice background color and also set a fixed width for the panel:

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

Then, add the **AnimationExtender** to the page, providing an **ID**, the **TargetControlID** attribute and the obligatory **runat="server"**:

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

Within the **<Animations>** node, use **<OnClick>** to run the animations once the user clicks on the panel. Add two animations to be executed parallelly:

```
<Animations>
  <OnClick>
    <Parallel>
      <FadeOut Duration="1.5" Fps="24" />
      <Resize Width="1000" Height="150" Unit="px" />
    </Parallel>
  </OnClick>
</Animations>
</ajaxToolkit:AnimationExtender>
```

For the sake of demonstration, this animation (and any other animation created using the Control Toolkit) is executed using JavaScript code, once the page runs. First of all we need access to the **AnimationExtender** control. The ASP.NET AJAX library provides the **\$find()** function for this task:

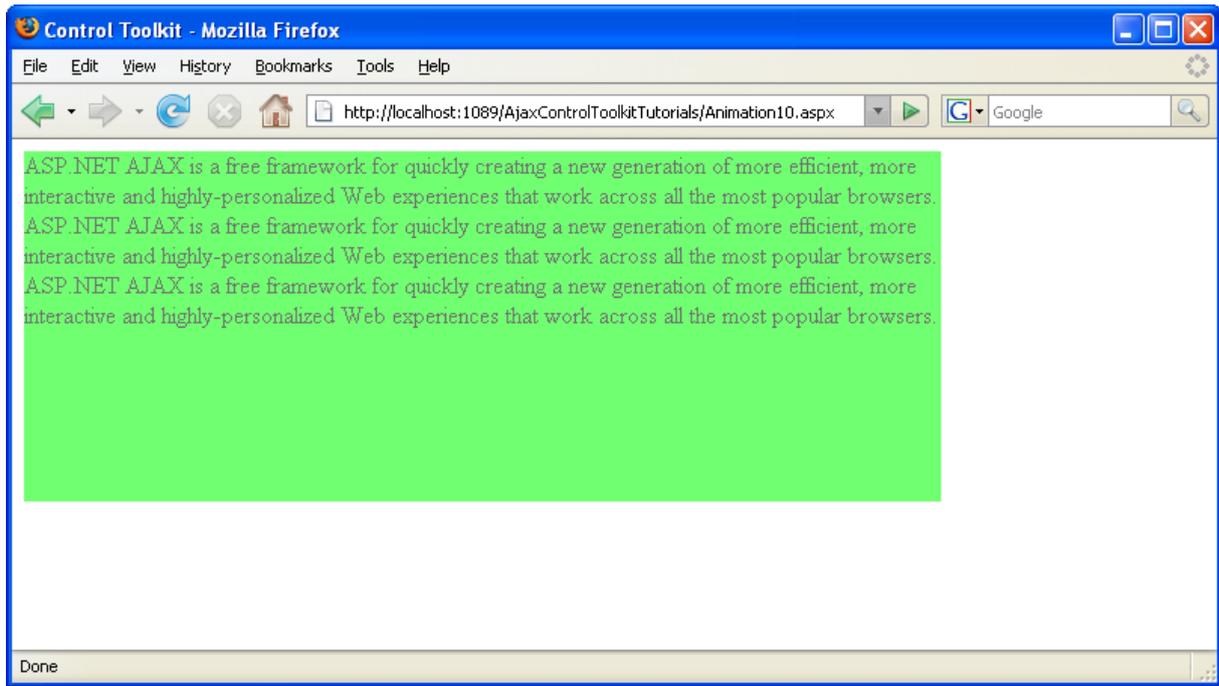
```
var ae = $find("ae");
```

The **AnimationExtender** control exposes a rich API, including methods with names identical to the event handlers used in the XML markup: **OnClick()**, **OnLoad()**, and so on. For instance, a call of the **OnClick()** method executes the animation within the **<OnClick>** element of the **AnimationExtender** control:

```
ae.OnClick();
```

Here is the complete client-side JavaScript code that emulates the click on the panel once the page has been fully loaded—note that the **pageLoad()** function name is used which is called by ASP.NET AJAX once the page and all included JavaScript libraries have been loaded.

```
<script type="text/javascript">
  function pageLoad() {
    var ae = $find("ae");
    ae.OnClick();
  }
</script>
```



The animation runs immediately, without a mouse click