MultiView Control in ASP.NET 2.0 – Part 1

ASP.NET 2.0 has brought a plethora of new user interface controls to developers. Most have been added to facilitate large amounts of duplicate coding that was required in previous versions of .NET. By expanding the ASP.NET toolbox to include several new controls that can accommodate a vast set of commonly used interface design patterns a developer can focus more on the user experience instead of custom writing their own user controls.

One of these new controls is the MultiView control. The premise behind this control is something that has been available to desktop application developers for quite some time. The idea being, to enable multiple views of control, layered on top of each other, and dynamically displayed at run-time.

In desktop development this was provided through the Tab controls, where the user can select different Tab panels and be presented with different visual layouts, a pattern often used in setting application Options for example. You can see examples of this style of interface all throughout any Windows operating system.

With web development you're limited by HTML, and such advanced visual tricks just aren't available. The most difficult problem is designing the layouts of each view/tab/panel in the actual visual designer without losing track of them or getting lost in layers.

The resolution is fairly simple, all Views contained within the MultiView control are just simply displayed all at once in the Visual Studio designer, while only one View control is displayed on the web page at run-time.

The purpose of this tutorial is to take a simple functional example application, and apply the MultiView control to see it in action. We will build a sample application that allows the user to flip between 3 different Themes in the application dynamically, and display 3 different View controls by clicking on some radio button controls.

During part one of this tutorial we will look at the MultiView and View controls and their documented uses. We will create a sample application with 3 themes and apply Skin files to allow our View controls to have a dynamic display without applying formatting to our controls at all. During the second part of this series we will implement the code required to tie all the pieces together.

Note: Visual Studio 2005 is required for this tutorial. Both VB and C# code samples will be provided

Setting Up

First we'll need a sample ASP.NET application to work with so begin by starting Visual Studio and creating a new web project file system based, named “MultiView” for VB and “MultiViewCS” for C#.

We'll use the Default.aspx page for implementing our example.

Let's first review exactly what we're trying to accomplish. First we'll need a MultiView control which will host 3 View controls, each of which will display a different Theme in our application, therefore we'll also have 3 themes defined with different appearances.

Each View control itself will have a Label, Textbox, Button, and RadioList controls. The first 3 controls are simply there to demonstrate the visual properties of the current theme. The RadioList control is responsible for flipping between the 3 View controls and also the 3 themes concurrently.

Apply dynamic Themes requires a bit of trickery and the use of a Session variable but in the end is fairly straightforward. Let's take a closer look at the MultiView and View controls next.

Copyright © 2006 DMXzone.com All Rights Reserved
To get more go to DMXzone.com

Thanks for your interest in the "MultiView Control in ASP.NET 2.0 – Part 1" article by Kevin Koch. To purchase the full article click here

Copyright © 2006 DMXzone.com All Rights Reserved
To get more go to DMXzone.com