

My First ColdFusion Builder Extension: "Set IIS Root"

Posted At : October 20, 2009 10:15 AM | Posted By : Steve

Related Categories: Productivity, ColdFusion, ColdFusion Builder, Web Servers

I had a magic moment this weekend when my wife and both of our babies were all asleep. I thought to myself "What should I do with this free time?". So, of course, I used it to make my first **ColdFusion Builder** extension, "**Set IIS Root**".

The extension allows me to right-click on any project or folder and set it as the web root in IIS directly from ColdFusion Builder. The functionality to do this was originally written by my friend, **Jason Holden**. Admittedly this is now only useful on flavors of Windows that have IIS, but don't allow multiple sites (Windows XP Pro, but not Windows 2003 or Vista), but I am in that group so it saves me time.

I was a little concerned about how difficult it would be to make an extension for ColdFusion Builder. It turned out to be really easy, taking me about an hour (or less) including the time to learn how to make an extension and to test it out. The ColdFusion Builder development team deserves major kudos here as do the authors of the **documentation**.

My extension required only three files:

- **ide_config.xml**: To define the interactions with the IDE
- **handlers/Application.cfm**: Empty file just to ensure I don't accidentally inherit any other functionality
- **handlers/setiisroot.cfm**: The file to do the actual work

Here is the code for **ide_config.cfm**:

```
<application>
  <name>Set IIS Root</name>
  <author>Steve Bryant</author>
  <version>1.0 Alpha</version>
  <email>steve@bryantwebconsulting.com</email>
  <description>Allows you to set the IIS web root to the selected project or folder.</description>

  <menucontributions>
    <contribution target="projectview" >
      <menu name="Set IIS Root">
        <filters>
          <filter type="folder" />
          <filter type="project" />
        </filters>
        <action name="Set IIS Root" handlerid="setiisroot">
          </action>
        </menu>
      </contribution>
    </menucontributions>
  <handlers>
    <handler id="setiisroot" type="CFM" filename="setiisroot.cfm" />
  </handlers>
</application>
```

Only 25 lines of code, the XML syntax here was really easy to use. The **menucontributions** element allows me to add a right click menu to the projectview, rdsview or outlineview (or any combination thereof). I defined **filters** so that it would show up only if a project or folder was clicked (it doesn't make sense to set a file as the web root).

I set the **handlerid** as "setiisroot" and defined that handler as being handled by "setiisroot.cfm" (handlers can only be .cfm files right now).

Here is the code for setiisroot.cfm:

```
<!--- Get XML from IDE --->
<cfset xInfo = XMLParse(Form.ideEventInfo) >

<!--- Get path that was selected --->
<cfset path = xInfo.event.ide.projectview.resource.XmlAttributes.path>
<!--- If path doesn't end in directory delimiter, IIS won't use last folder in path --->
<cfif Right(path,1) NEQ "/">
    <cfset path = "#path#/">
</cfif>

<!--- This batch file in which the work is done --->
<cfset BatchFile = "#path#load_iis.bat">
<!--- Jason Holden's code here does the work --->
<cfset LoadIIS = 'cscript %SystemDrive%\inetpub\adminscripts\adsutil.vbs SET /W3SVC/1/ROOT/path "%~dp0"'>

<!--- Create the batch file --->
<cffile action="WRITE" file="#BatchFile#" output="#LoadIIS#">
<!--- Execute the batch file --->
<cfexecute name="#BatchFile#" timeout="999" />
<!--- Remove the batch file (no reason to keep it around) --->
<cffile action="DELETE" file="#BatchFile#">
```

The only part of this code that is specific to ColdFusion Builder is dealing with ">[Form.ideEventInfo](#). This is an XML string with information about the user's event. In my case, I just needed to find out what folder (or "resource" in the IDE's parlance) the user clicked. After that, it was just a matter of running Jason's code to set the web root.

I think I can make the code a bit shorter and cleaner in the next build, but this was a good start.

The [installation](#) was quick and easy, which allowed me to test and use the extension with little trouble (I just had to remember to [reload](#) it if I changed the XML file). [Packaging](#) it as an extension was a simple matter of zipping it up.

ColdFusion Builder also offers a lot more data as well as the ability to display a form to get data from the user and/or show them a response back. For now, however, I am really happy that I was able to quickly put together a very simple extension to make my life a little easier.