

Re: ASP .NET 2.0 Unanswered questions...

Source:

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.framework.aspnet/2007-10/msg01843.html>

- *From:* "Scott M." <s-mar@xxxxxxxxxxxxxx>
 - *Date:* Mon, 15 Oct 2007 12:50:23 -0400
-

Yes, this would be in scenarios where you just deploy your pages (.aspx), but no assembly. This is what folks who don't use the code-behind model would experience. But, I was clear in my earlier posts that this is **not** the scenario I was referring to.

"Rory Becker" <RoryBecker@xxxxxxxxxxxxxx> wrote in message news:b0ac48a01f1b98c9dd69045d44b0@xxxxxxxxxxxxxx

!> He has also said in this thread that VS 2003 produced !> multiple assemblies (one for each page). He was talking about a different kind of assembly: JIT-compiled assemblies. Those go in the Temporary ASP.NET Files directory.

I still say these are not Jit Compiled. These are shadow copies. and there is still only 1 per project not 1 per page.

That's where we disagree Rory. The assembly in the temp folder **is** the JIT version of the project assembly.

<http://www.codeproject.com/aspnet/PreCompileAspx.asp>

I also found this (look at the temporary files section)

http://msdn2.microsoft.com/en-us/library/aa479328.aspx#aspnet_http_run_time_topic5

Thanks Scott

Interestingly this 2nd article seems to suggest that Bruce is correct under certain circumstances

By design, the HttpApplication object looks for a class named after the requested ASPX file. If the page is named sample.aspx, then the

Re: ASP .NET 2.0 Unanswered questions...

corresponding class to load is named ASP.sample_aspx. The application object looks for such a class in all of the assembly folders of the Web application—the Global Assembly Cache (GAC), the Bin subfolder, and the Temporary ASP.NET Files folder.

If no such class is found, the HTTP infrastructure parses the source code of the .aspx file, creates a C# or Visual Basic .NET class (depending on the language set on the .aspx page), and compiles it on the fly. The newly created assembly has a randomly generated name and is located in an application-specific subfolder of the following path:

C:\WINDOWS\Microsoft.NET\Framework\v1.1.4322\Temporary ASP.NET Files.

I have yet to try this but it seems to apply when the ASPX represents a class which is not in the project assembly. I have no idea what would cause that though.

Interesting.....

--

Rory