

## Tip/Trick: How to Register User Controls and Custom Controls in Web.config

Sunday, November 26, 2006

<a href="/scottgu/Tags/.NET">.NET (/scottgu/Tags/.NET)</a>	<a href="/scottgu/Tags/ASP.NET">ASP.NET (/scottgu/Tags/ASP.NET)</a>	<a href="/scottgu/Tags/Tips%20and%20Tricks">Tips and Tricks (/scottgu/Tags/Tips%20and%20Tricks)</a>
<a href="/scottgu/Tags/Visual%20Studio">Visual Studio (/scottgu/Tags/Visual%20Studio)</a>		

I've been including this technique in my ASP.NET Tips/Tricks talks ([http://weblogs.asp.net/scottgu/archive/2006/11/10/ASP.NET-Connections-Slides-\\_2B00\\_-Demos.aspx?CommentPosted=true](http://weblogs.asp.net/scottgu/archive/2006/11/10/ASP.NET-Connections-Slides-_2B00_-Demos.aspx?CommentPosted=true)) the last year, but given how many people are always surprised by its existence I thought it was worth a dedicated tip/trick post to raise the visibility of it (click here ([http://weblogs.asp.net/scottgu/pages/ASP.NET-2.0-Tips\\_2C00\\_-Tricks\\_2C00\\_-Recipes-and-Gotchas.aspx](http://weblogs.asp.net/scottgu/pages/ASP.NET-2.0-Tips_2C00_-Tricks_2C00_-Recipes-and-Gotchas.aspx)) to read other posts in my ASP.NET Tips/Tricks series).

### Problem:

In previous versions of ASP.NET developers imported and used both custom server controls and user controls on a page by adding `<%@ Register %>` directives to the top of pages like so:

```
<%@ Register TagPrefix="scott" TagName="header" Src="Controls/Header.ascx" %>
<%@ Register TagPrefix="scott" TagName="footer" Src="Controls/Footer.ascx" %>
<%@ Register TagPrefix="ControlVendor" Assembly="ControlVendor" %>

<html>
<body>
    <form id="form1" runat="server">
        <scott:header ID="MyHeader" runat="server" />
    </form>
</body>
</html>
```

Note that the first two register directives above are for user-controls (implemented in .ascx files), while the last is for a custom control compiled into an assembly .dll file. Once registered developers could then declare these controls anywhere on the page using the tagprefix and tagnames configured.

This works fine, but can be a pain to manage when you want to have controls used across lots of pages within your site (especially if you ever move your .ascx files and need to update all of the registration declarations).

### Solution:

ASP.NET 2.0 makes control declarations much cleaner and easier to manage. Instead of duplicating them on all your pages, just declare them once within the new pages->controls section with the web.config file of your application:

```
<?xml version="1.0"?>

<configuration>

  <system.web>

    <pages>
      <controls>
        <add tagPrefix="scottgu" src="~/Controls/Header.ascx" tagName="header"/>
        <add tagPrefix="scottgu" src="~/Controls/Footer.ascx" tagName="footer"/>
        <add tagPrefix="ControlVendor" assembly="ControlVendorAssembly"/>
      </controls>
    </pages>

  </system.web>

</configuration>
```

You can declare both user controls and compiled custom controls this way. Both are fully supported by Visual Studio when you use this technique -- and both VS 2005 Web Site Projects and VS 2005 Web Application Projects support them (and show the controls in WYSIWYG mode in the designer as well as for field declarations in code-behind files).

One thing to note above is the use of the "~" syntax with the user-controls. For those of you not familiar with this notation, the "~" keyword in ASP.NET means "resolve from the application root path", and provides a good way to avoid adding "..\" syntax all over your code. You will always want/need to use it when declaring user controls within web.config files since pages might be using the controls in different sub-directories - and so you always need to resolve paths from the application root to find the controls consistently.

Once you register the controls within the web.config file, you can then just use the controls on any page, master-page or user control on your site like so (no registration directives required):

```
<html>
<body>
  <form id="form1" runat="server">
    <scott:header ID="MyHeader" runat="server" />
  </form>
</body>
</html>
```

Hope this helps,

Scott

P.S. Special thanks to Phil Haack (<http://haacked.com/>) who blogged about this technique ([http://haacked.com/archive/2006/11/14/Register\\_Custom\\_Controls\\_In\\_Web.config.aspx](http://haacked.com/archive/2006/11/14/Register_Custom_Controls_In_Web.config.aspx)) as well earlier this month (for those of you who don't know Phil, he helps build the very popular SubText blog engine (<http://subtextproject.com/>) and has a great blog).

---

35 comments

Question: I've been using this sparingly... I'm worried that it might impace performance. Do you think so? In other words, if I register 30 user controls and 5 custom server controls in this way, will my pages slow down? Will compilation slow down?

Thanks!

— **John West** - Sunday, November 26, 2006 10:03:52 PM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24123)

Thanks for the shot out Scott! Much appreciated, especially from such a great blog and blogger. Yep, I'm a ScottGu fanboy. ;)

— **Haacked** (<http://haacked.com/>) - Sunday, November 26, 2006 10:15:03 PM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24124)

Hi John,

The good news is that there isn't any performance difference between registering them in a web.config file vs. at the top of a page.

The runtime performance is exactly the same (they get compiled down to the same instructions in both scenarios).

Compilation should also compile at the same performance.

Hope this helps,

Scott

— **ScottGu** - Sunday, November 26, 2006 11:08:08 PM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24125)

Scott,

There's a mistake in your code. In the "Solution" in which the registrations are placed in the web.config file, this line:

should be ....

because you used a different tag prefix ("scottgu") in the revised version.

Thanks for the info!

— **Chris adf471587879rzq** - Sunday, November 26, 2006 11:28:00 PM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24126)

Hi Chris,

Good catch! ;-)

Hopefully you got the point anyway! :-)

Thanks,

Scott

— **ScottGu** - Sunday, November 26, 2006 11:30:12 PM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24127)

Hi Adam,

There are cases when the designer can't infer the base class of a user-control with VS 2005 Web Application Projects - in which case it just generates a field declaration in the code-behind of the page as type "UserControl".

To get a strongly-typed declaration of the user-control, you can manually add the declaration to your code-behind class (not the .designer class) of the type you want (which is typically the UserControl code-behind class name).

Once you do this the designer will pick this up and use it as the field declaration and you'll get full intellisense.

Hope this helps,

Scott

— **ScottGu** - Monday, November 27, 2006 4:26:05 AM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24128)

good trick

I have been using this trick but for only those controls which are used through out the Site. I always thought if a user control is used only at one or 2 place, It should be placed in the Page itself, but if used at many pages in the site then should be declared in the web.config file

Thanks

Vikram

— **vikram (<http://www.vikramlakhota.com/>)** - Monday, November 27, 2006 4:44:23 AM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24129)

With regards to the user control being declared with the incorrect data type in the designer.cs file - does your solution (manually declare the variable in the code behind file) "force" the invalid declaration to be removed from the designer.cs file? Or will there be a duplicate named variable with a different data type?

I am using Web Application Projects and understand this to be a limitation of that project type currently.

Is this something that is going to be addressed in a service pack or hotfix to the web application project?

— **Ryan S** - Monday, November 27, 2006 12:03:43 PM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24130)

Hi Ryan,

If you declare the control type/name in your code-behind file in a VS 2005 Web Application Project, then the designer automatically avoids adding it to the .designer.cs/.vb file.

This avoids any duplicate name collisions (note: you'll need to flip once to source view or design view and make a change in order to have the designer remove it from the .designer file - from then on out it will be omitted).

The reason you sometimes need to-do this are in cases where the control is implemented inside the same assembly as the pages. You can sometimes get into an un-buildable state where the designer can't reflect on the project to infer the type (this isn't a problem with controls implemented in other projects or separate assemblies - since they are already successfully compiled). This situation is less of a "bug" per-se - and more of a design outcome of building everything within the same assembly. We are going to continue to look for ways to make this case less common - but there will always be some times when you can't reflect and get the type details. In cases like this you can just declare the field in your code-behind file and everything will work.

Hope this helps,

Scott

— **ScottGu** - Monday, November 27, 2006 3:08:06 PM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24131)

quite nice! i didn know it exists.

does this add any overhead to the pages that don't implement the controls?

— **Arash** - Monday, November 27, 2006 7:50:09 PM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24132)

Hi Arash,

The good news is that there is no performance penalty to using this trick. It is just as efficient as without.

Thanks,

Scott

— **ScottGu** - Monday, November 27, 2006 8:48:31 PM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24133)

Hi Ryan,

I actually just chatted with someone on the VS 2005 Web Application Project feature team, and he actually mentioned that they've added some better detection support for this in SP1. So you should actually see the cases where you need to declare a field in the code-behind to get a type-specific value go down.

Hope this helps,

Scott

— **ScottGu** - Monday, November 27, 2006 8:49:33 PM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24134)



Hi Adrian,

Correct - the collection settings within a web.config file will inherit down to sub-applications by default.

What you can do in your sub-applications is to use instead or - and just specify the setting you wish to remove.

That way you won't clear everything.

Hope this helps,

Scott

— **ScottGu** - Wednesday, November 29, 2006 8:42:33 PM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24135)

Hey Scott,

I have a usercontrol that exists in my master page, and I access its properties with:

```
CType(Master, botwmaster).myusercontrol.Visible = False
```

(for example)

Doing this requires that I add a reference to the control from the page:

My question is, how would I add that reference in the web.config so I don't have to do that in every page? I'd love to eventually get all control declarations and references to exist in a centralized place (for those that exist on multiple pages)

More food for thought - it would be great if things declared in a master page would bubble down to your aspx page - the control is registered in the master page with @register already.

— **Dave Frank (<http://davefrank.com>)** - Thursday, November 30, 2006 2:29:09 AM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24136)

Thanks for all the information, Scott.

This approach seems cleaner to me than @Register'ing the same controls all over the place. So, I embarked on a 5-minute journey through my application to take the registrations out of the .ASPX pages and all into web.config.

I ran into a problem though on pages where I would instantiate one of my user controls in codebehind and add it to a table cell. I've got a UC story.ascx with a public storyId that renders the content for a story - simple enough. I use the following in codebehind on various pages to stick stories in table cells:

```
story myStory;  
myStory = (story)LoadControl("userControls/story.ascx");  
myStory.storyId = 1234;  
cell.Controls.Add(myStory);
```

However when I moved all the UC registrations to web.config, I ran into "The type or namespace name 'story' could not be found" - can you tell me why or how else I can accomplish this? Thanks!

— **BT** - Friday, December 1, 2006 7:26:23 PM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24137)

Hi BT,

If you are dynamically loading and then type-casting the control with a VS 2005 Web Site Project, you can add a element to the top of your page. This will add an assembly reference for you automatically and allow you to reference it like above.

Hope this helps,

Scott

— **ScottGu** - Saturday, December 2, 2006 11:46:20 PM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24138)

Hi Dave,

One option to avoid the directive would be to use the VS 2005 Web Application Project. This compiles the entire project into a single assembly, and so avoids the need to explicitly reference page/controls.

Hope this helps,

Scott

— **ScottGu** - Sunday, December 3, 2006 12:39:37 AM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24139)

Hi Scott,

how do i force a page to load my usercontrol 's dll file from another path, instead of the /bin folder.

reasons for my wanting to do this.

- 1) i want to keep my control and its dll separate.
- 2) i don't want the website restarting each time the admin uploads a new usercontrol.

thnx.

— **Anupam** - Tuesday, December 5, 2006 12:43:47 PM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24140)

Hi Scott,

This is very useful, but it brings some trouble along in my project: if I use a user control in another user control, and both controls are in the same directory in the project (in your example, ~/Controls/ ), this compile error occurs:

The page '/Controls/Outer.ascx' cannot use the user control '/Controls/Inner.ascx', because it is registered in web.config and lives in the same directory as the page.

I'd rather not move nested controls to a different directory. Is there a solution for this codewise, or will there be a fix in SP1?

Cheers

— **JP** (<http://www.onestat.com>) - Wednesday, December 6, 2006 9:53:45 AM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24141)

Nice tip.

I changed my pages to use it.

One issue:

I have a user control which uses another user control. They both reside in a folder named \Controls.

I got this error:

The page '/MyApp/Controls/CtlA.ascx' cannot use the user control '/MyApp/Controls/CtlB.ascx', because it is registered in web.config and lives in the same directory as the page.

I just left it the old way.

BTW - that was a very good error message - it allowed me to pinpoint the problem right away.

— **Joe** - Wednesday, December 6, 2006 5:26:12 PM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24142)

Scot - In a blog from Aug 28 2005 you discussed building a library of user controls and the ability to then consume them in other web applications. Twice in the blog it was asked how to reference the user control in the a page of the consuming application. For example, one entry said

Hi Scott. I am having a similar problem to a previous comment. This all works fine if I include the separate .ascx file. However, as soon as I set the build flag to not allow updating (which as a consequence does not create a separate .ascx file), I cannot get it to work. Your previous response was that ASP.NET can resolve the .ascx references at runtime. How do you register the control? without registering it, compilation fails. Thanks in advance Curtis.

I had been trying to do this for better part of two days and when I say the blog entry I thought, "Finally, I'll get the secret sauce". But your response was completely disappointing..

Hi Curtis,

What I'd recommend is for you to have the user-control project be marked as updatable, and copy the .ascx file into the consuming project.

You could then compile the consuming project as non-updatable. This will then remove the HTML for both the pages on the site, as well as the user-control you are using.

Hope this helps,

Scott

How can one develop a distributable library of user controls and protect the source, as you claim is possible with the plethora of build options in VS2005, if you can't do this. I just want to know if it's possible or not and what one has to do?

— **gbrock\_isi** (<http://www.intraprise.com>) - Thursday, December 7, 2006 2:47:13 PM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24143)

Hi Greg,

Here is a blog post that discusses how to convert a user control into a compiled control:  
<http://blogs.msdn.com/davidebb/archive/2005/10/30/487160.aspx>

Thanks,

Scott

— **ScottGu** - Friday, December 8, 2006 5:23:04 AM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24144)

To those of you receiving this error: (JP)

The page '/Controls/Outer.ascx' cannot use the user control '/Controls/Inner.ascx', because it is registered in web.config and lives in the same directory as the page.

One solution I have read is to move the control out of the directory its currently sharing with outer.ascx, but this isn't feasible in my project.

I found that you can simply re-register the control inside of the outer.ascx:

and the problem goes away.

— **Brian Lowry** - Wednesday, December 13, 2006 3:05:37 PM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24145)

How about Web Services? I have this Web User Control with a static method that uses LoadControl and executes some methods that use the controls inside the WUC. I am using it to render HTML which I then embed into an email. I want to do this inside a Web Service and nothing seems to be working. The error, of course, is that UserControl type is not valid in current context, but I can't add a register directive to the asmx, I can't use an assembly directive (I get ASP.NET runtime error: The file 'ucusercontrol.ascx.cs' cannot be processed because the code directory has not yet been built.) and declaring the user control in the web.config as in this article doesn't work.

Any idea how I can use inside a web service the type declared in the code behind of a web user control ?

Thanks.

— **Siderite** (<http://siderite.blogspot.com>) - Monday, January 15, 2007 10:43:32 AM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24146)

Hi Siderite,

Have you seen this article of mine: [http://weblogs.asp.net/scottgu/archive/2006/10/22/Tip\\_2F00\\_Trick\\_3A00\\_-Cool-UI-Templating-Technique-to-use-with-ASP.NET-AJAX-for-non\\_2D00\\_UpdatePanel-scenarios.aspx](http://weblogs.asp.net/scottgu/archive/2006/10/22/Tip_2F00_Trick_3A00_-Cool-UI-Templating-Technique-to-use-with-ASP.NET-AJAX-for-non_2D00_UpdatePanel-scenarios.aspx)

It enables you to use .ascx templates within web-services to dynamically generate HTML.

Thanks,

Scott

— **ScottGu** - Saturday, January 20, 2007 3:47:05 AM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24147)

&nbsp;Thanks, Scott! It's a nice implementation, but it uses reflection. I was looking for an ASP.NET 1.1 like use of the web user control object in other objects, but I guess that's out of the question now.

&nbsp;It was particularly interesting to see that your code to do the same thing as mine used a completely different method. You used LoadControl, I used Controls.Add (and had a lot of problems with it, of course) and then ServerExecute instead of RenderControl... Very useful stuff. Thanks!

— **Siderite** (<http://siderite.blogspot.com>) - Friday, February 2, 2007 6:19:02 AM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24148)

Hi

as u said that instead of registering a user control once per web page that v r going to use it within,its better to register it in web.config file

I did that and gave tagprefix,tagname and src attributes proper values.but the problem is

DO I NEED TO DRAG THE USER CONTROL ON MY WEB PAGE(S)

OR WUD I HAV TO MANUALLY WRITE THE TAGPREFIX AND TAGNAME ETC IN XYZ.ASPX PAGE TO SET THE VALUE OF ID ETC

IF I DO LATER WAY THEN THE VALUES OF TAGPREFIX AND TAGNAME I SPECIFIED ARE NOT PICKED UNDER XYZ.ASPX PAGE.(THEY ARE NOT SHOWN IN POP UP LIST)

— **nishant** - Friday, February 23, 2007 10:59:07 AM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24149)

This works great but when I view the page(in Design mode) that contains the controls the controls show up in red. An error shows for each control that states: This control cannot be displayed because its TagPrefix is not registered in this Web Form. Also, the same error shows up in the Error List. The odd thing is that I can build the solution, run the project and view the page/controls perfectly. Is there anything that can be done to get rid of this annoying error message?

— **Richard** - Tuesday, February 27, 2007 1:39:49 AM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24150)



Hi Richard,

Can you try closing the project and re-open it to see if it makes any difference?

Are you opening things the same way in VS as they get invoked via IIS? Meaning, is the web.config file in the root of the project?

Thanks,

Scott

— **ScottGu** - Tuesday, February 27, 2007 7:13:02 AM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24151)

Superb tip, works a treat and really helps to make the aspx pages look much neater.

— **Stephen Kinsey** (<http://www.stevekinsey.com/>) - Thursday, March 1, 2007 3:32:58 PM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24152)

Hi Scott !!!

thanks for sharing this useful information with us....

I also want to know how I can refers to User Control which resides in different directory than my project root directory

— **Shekhar** (<http://shekhar007.blogspot.com>) - Friday, March 2, 2007 6:39:19 AM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24153)

Hi Shekhar,

You can do this by using the "~" syntax when referencing the controls from the root:

~/subdir/mycontrol.ascx

Hope this helps,

Scott

— **ScottGu** - Sunday, March 4, 2007 2:24:40 AM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24154)

Hi Scott !!!

Thanks a lot for ur reply.....

The solution u have given works perfectly when controls are present in inner directory of root directory.

Wat about if the user controls are present in outside of root directory?

is it possible to refer that controls ?

Thanks again for your kind help

Shekhar

— **Shekhar** (<http://shekhar007.blogspot.com>) - Monday, March 5, 2007 6:22:14 AM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24155)

Hi Shekhar,

Unfortunately I don't think it is possible to register user controls outside of the application root directory.

Sorry!

Scott

— **ScottGu** - Wednesday, March 7, 2007 3:25:57 AM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24156)

Hi Scott,

I had previously used your tips above and added all of my usercontrols into the web.config file. Great! It's a really useful feature and it worked perfectly.

Now it doesn't. For some reason all of my code has stopped working and when trying to debug the build fails (along with intellisense etc.)

I was adding controls dynamically e.g.

```
Dim News As Controls_News = LoadControl("~/Controls/News.ascx")
```

Where Controls\_News is the partial class name.

Adding a register directive to the page solves this issue, I am just wondering if you or any of your readers here would know why it worked for a while and then stopped?

I created a new project, cleaned ASP.NET temp files directory and so on and the problem persists.

— **Justin Wignall** - Wednesday, March 7, 2007 3:46:32 PM (/scottgu/tip-trick-how-to-register-user-controls-and-custom-controls-in-web-config#comment-24157)

Comments have been disabled for this content.

---

Terms Of Use (<http://www.asp.net/terms-of-use>) - Powered by Orchard (<http://www.orchardproject.net>)