

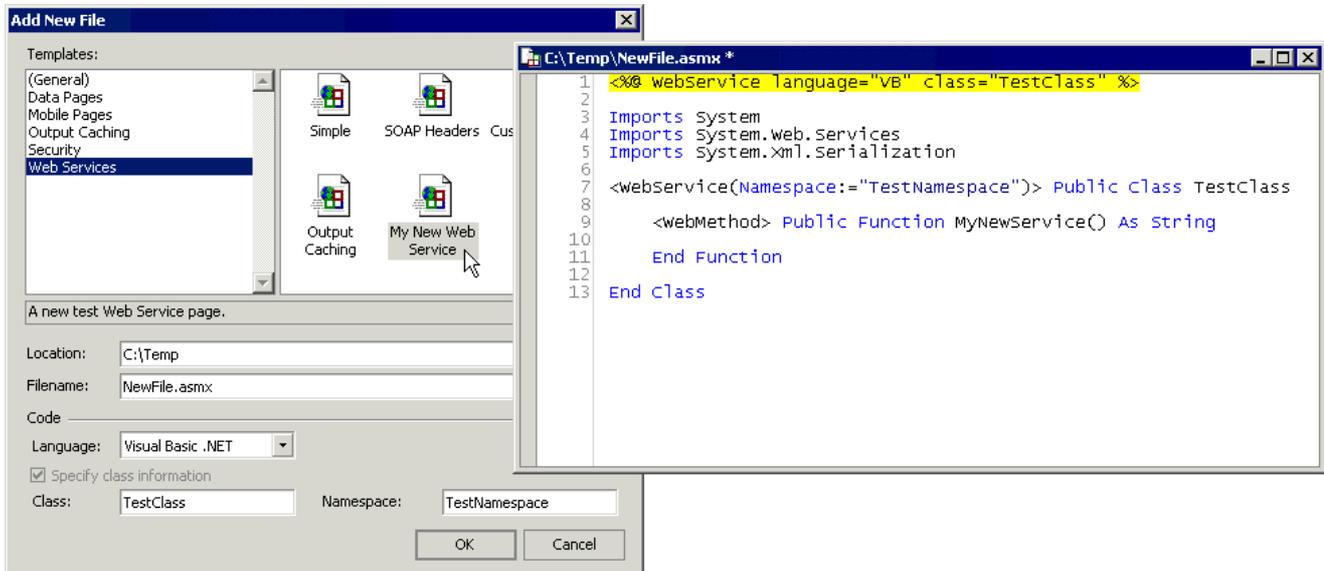
Editing Document Templates Directly

In the current version of Web Matrix, the only way to modify or add document templates is to edit the appropriate files directly, or place new files into the Web Matrix Templates subfolder hierarchy, and then edit the configuration file.

Let's add a new document template to Web Matrix. First, create a new folder named My New Web Service within Web Matrix's Web Services folder. In this folder create two subfolders, VB and C#, which correspond to the two languages. In each of these folders create a file named NewFile.asmx (the file name can actually be anything you like, though all the supplied templates use the name NewFile). Then it's simply a matter of adding the following to the WebMatrix.exe.config file, within the Web Services Templates part of the <documentTypes> section:

```
<templateDocumentType extension="asmx" templateCategory="Web Services"
    name="My New Web Service"
    createNewDescription="A new test Web Service page." />
```

The <templateDocumentType> element defines the name of the subfolder, the file extension, the category it will appear in within the New File dialog, and the file description. The following screenshot shows this dialog, together with the file that our new template creates:



The contents of the new file are, of course, governed by the contents of the template. The following listing shows the VB template we placed in the Web Services\My New Web Service\ folder:

```
<%@ WebService language="VB" class="%%ClassName%%" %>

Imports System
Imports System.Web.Services
Imports System.Xml.Serialization
```

```
<WebService(Namespace:="%%NamespaceName%")> Public Class %%ClassName%%  
    <WebMethod> Public Function MyNewService() As String  
    End Function  
End Class
```

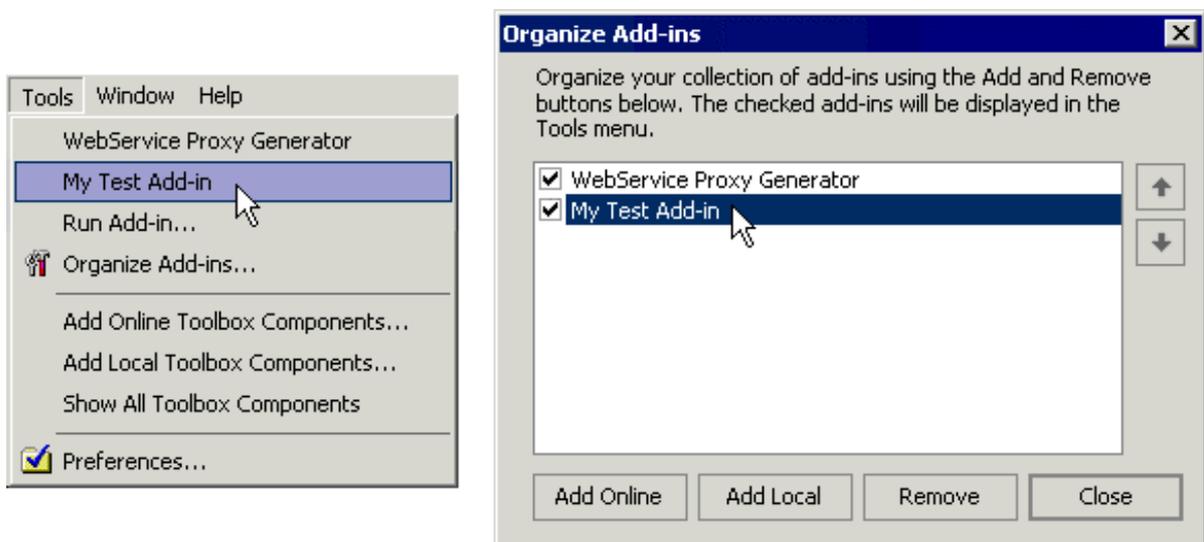
The placeholders for the class and the namespace are enclosed in double percent characters (%%). When the new file is created from the template, the values entered into the Class and Namespace textboxes in the New File dialog are substituted for these placeholders. Of course, the template can contain any code or content that you require – the various Data Pages templates are good examples of this – and you can, if you wish, just edit these rather than adding new ones.

Installing and Using Add-ins and Code Builders

Web Matrix's extensibility includes both Add-ins and Code Builders, which means that you can supplement the ones provided by default with Web Matrix in order to perform specific tasks that you regularly carry out.

An Add-in is a .NET assembly that (usually) has a visual interface, and can be created in Visual Studio .NET using any .NET language. Creating an Add-in isn't a trivial task, but is easy enough if you are competent with Visual Studio .NET. A detailed tutorial, which describes the requirements and the techniques for constructing and installing an add-in, is available from the Web Matrix community web site.

Add-ins can be installed using the Organize Add-ins dialog, where you use the Add Local button to add an assembly. You can also click Add Online and install any of the add-ins available from the ASP.NET Web Matrix site "gallery" page (which we saw when we looked at installing Toolbox components). Once installed, the new add-in will be visible in the Tools menu:



Add-ins can be used to generate code or HTML (perhaps based on a user's input or other environmental features), create or manipulate disk files, run command-line utilities, or even automate other programs outside the Web Matrix environment. Some of the things that you unfortunately cannot do (at least in the current version) are interact with the Web Matrix IDE, or with contents of the edit windows. However, you can copy text to the clipboard, so that users can paste it into an edit window afterwards.

Unlike an add-in, a Code Builder has no visual interface. Instead, it sits in the Code Builders section of the Toolbox, and is used by dragging it onto a page. Like Add-ins, Code Builders are created as .NET assemblies, and are installed by placing the assembly in the CodeBuilders subfolder of the main Web Matrix program folder and adding an appropriate <toolboxItem> element to the <toolbox> section configuration file.

Detailed instructions for creating Add-ins and Code Builders are outside the scope of this document, but you can find more information and examples on the Microsoft ASP.NET Web Matrix Project home page at <http://www.asp.net/WebMatrix/>.

Customizing the Web Matrix Interface

The Classes window and the separate Class Browser tool provide a list of useful Framework classes, from which you can get information about the members of these classes. There are also collapsible folder-based lists of the main ASP.NET classes in the Classes window, which makes it easy to quickly find details of these regularly used classes.

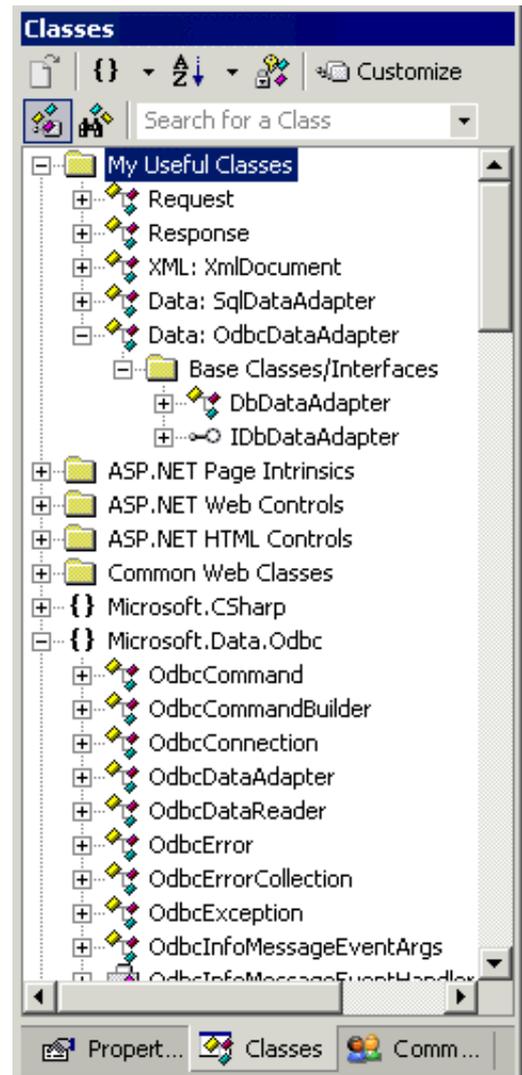
The Customize button at the top right of the Classes window can be used to add assemblies that are already in the GAC (Global Assembly Cache), so that the classes they contain are shown in the list. However, this does not modify the collapsible folders of classes, and the modifications you make using the Customize dialog are lost if you delete the WebMatrix.settings file. However, editing the WebMatrix.exe.config configuration file provides a way to change these lists of folders and create new folders, as well as permanently adding classes to the list.

Customizing the Classes Window

In screenshot you can see that a new folder named My Useful Classes has been added to the Classes window. This folder has been populated with a few classes from the .NET Framework Class Library. The first four are classes that appear in the other (existing) folders, though a new item has been added to the list (the OdbcDataAdapter class) as well:

Adding a new folder that contains classes from the existing namespaces in the Classes window simply involves adding a new <group> element in the configuration file. To edit the list of classes in an existing folder, you just edit the content of the appropriate <group> element.

Find the <classView> element in the WebMatrix.exe.config configuration file. This element defines which classes that are already qualified in the <runtime> section of the file will appear in the Classes window. In the following listing, you can see the default assemblies, the Microsoft.Data.Odbc assembly we added ourselves, and the custom group we added to create the My Useful Classes folder (and the fact that we used a customized display name in the name attribute):



```
<classView>
  <assembly name="mscorlib" displayName="mscorlib"/>
  <assembly name="System" displayName="System"/>
  <assembly name="System.Data" displayName="System.Data"/>
  <assembly name="System.Drawing" displayName="System.Drawing"/>
  <assembly name="System.Web" displayName="System.Web"/>
  <assembly name="System.Web.Mobile" displayName="System.Web.Mobile"/>
  <assembly name="System.Web.Services" displayName="System.Web.Services"/>
  <assembly name="System.Xml" displayName="System.Xml"/>
  <assembly name="Microsoft.Data.Odbc" displayName="Microsoft.Data.Odbc"/>
  <group name="My Useful Classes">
    <groupItem name="Request" type="System.Web.HttpRequest, System.Web"/>
    <groupItem name="Response" type="System.Web.HttpResponse, System.Web"/>
    <groupItem name="XML: XmlDocument" type="System.Xml.XmlDocument, System.Xml"/>
  </group>
</classView>
```

```

type="System.Xml.XmlDocument, System.Xml" />
  <groupItem name="Data: SqlDataAdapter"
    type="System.Data.SqlClient.SqlDataAdapter, System.Data" />
  <groupItem name="Data: OdbcDataAdapter"
    type="Microsoft.Data.Odbc.OdbcDataAdapter, Microsoft.Data.Odbc" />
</group>
<group name="ASP.NET Page Intrinsic">
  ... etc ...
</group>
...

```

To add a class from an assembly that is not already in the default list, you also have to add that assembly to the `<runtime>` section of the configuration file. This also ensures that the assembly will be included in the list even if you delete the `WebMatrix.settings` file at sometime in the future (for example when installing your own Add-ins or Code Builders). In the next listing, you can see the `Microsoft.Data.Odbc` assembly we added here so that the classes it contains will permanently appear in the **Classes** window list:

```

<runtime>
  <assemblyBinding xmlns="urn:schemas-microsoft-com:asm.v1">
    <probing privatePath="IDE;Packages;AddIns;Components;CodeBuilders" />
    <qualifyAssembly partialName="mscorlib" fullName="mscorlib,
      Version=1.0.3300.0, Culture=neutral,
      PublicKeyToken=b03f5f7f11d50a3a" />
    ... etc ...
    <qualifyAssembly partialName="Microsoft.Data.Odbc" fullName="Microsoft.Data.Odbc,
      Version=1.0.3300.0, Culture=neutral,
      PublicKeyToken=b77a5c561934e089" />
  </assemblyBinding>
</runtime>

```

The `fullName` attributes of these elements should be all on one line, and not wrapped as we've had to do in the listing here. This also applies to the `type` attribute of the `<groupItem>` elements we showed in the previous listing.

To get the information you require to add an assembly to the "qualified" list, you can open the assembly in the `ildasm.exe` tool that is provided with the .NET Framework SDK (in the `Framework SDK\Bin` folder). Once the assembly is opened you can double-click the **MANIFEST** entry to see the version and public key token values. The `Microsoft.Data.Odbc` assembly that we added implements the .NET ODBC Provider, which is not included in the default installation of the .NET Framework SDK. You can obtain it from the Microsoft Data Access web site at <http://www.microsoft.com/data/>. By default, it installs the assembly that implements the ODBC provider in:

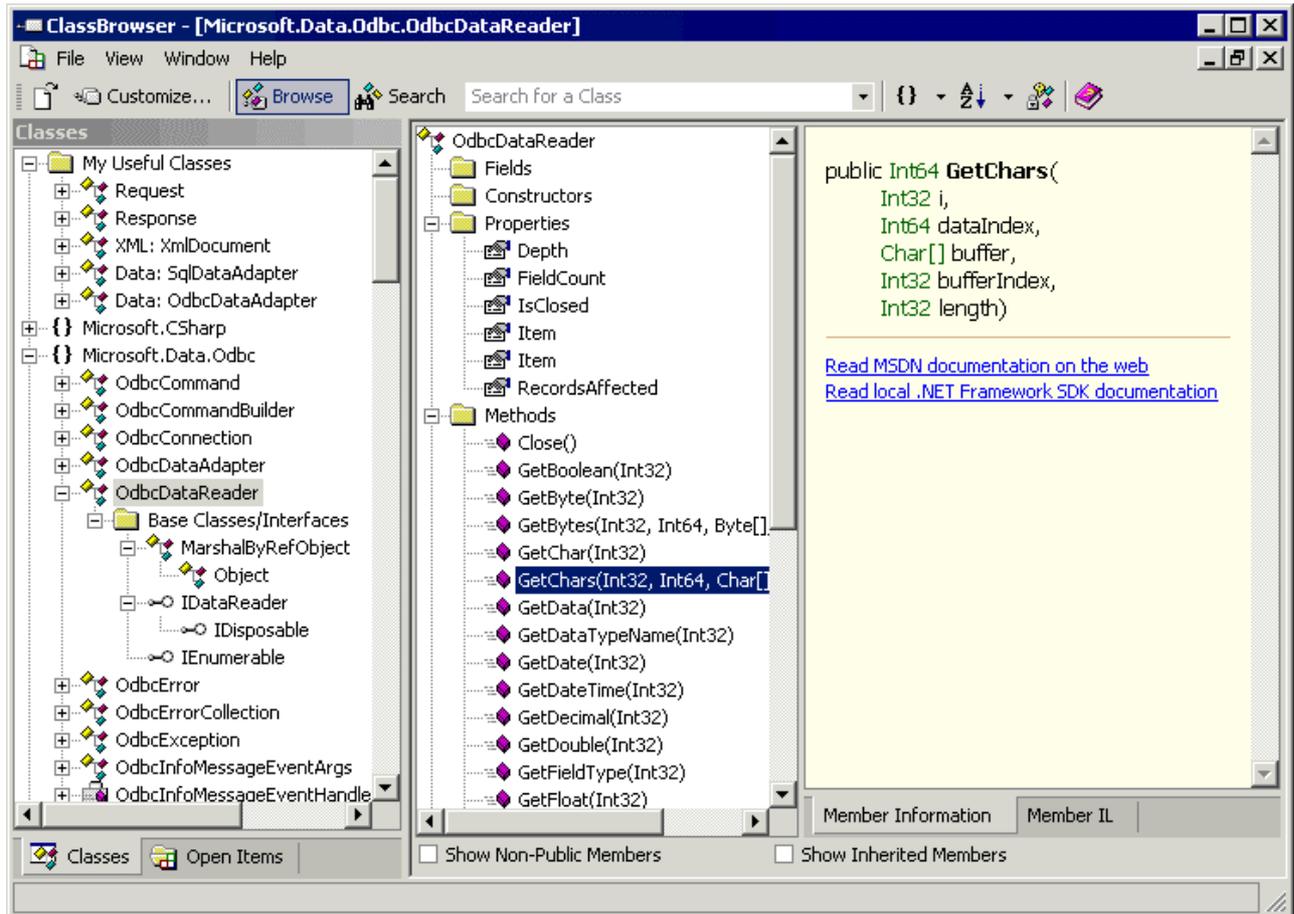
```

\Program Files
  \Microsoft.NET
    \Odbc.Net
      \Microsoft.Data.Odbc.dll

```

Customizing the Class Browser Assembly List

It's also possible to edit the list of classes that are displayed in the Class Browser tool, and add folders to the list just as we did for the Classes window in the Web Matrix IDE. The `ClassBrowser.exe.config` file contains the same `<classView>` and `<runtime>` elements as the Web Matrix configuration file, so the technique is the same. In the following screenshot, you can see that the My Useful Classes folder has been added to the Class Browser simply by copying it from the `<classView>` section of the `WebMatrix.exe.config` file:



The Class Browser stores its current settings in your Documents and Settings folder in the same way as Web Matrix does. As with Web Matrix, if your changes to the `ClassBrowser.exe.config` file are not picked up when you next open the Class Browser, you should manually delete the `Classbrowser.settings` file to force a re-read of the `ClassBrowser.exe.config` file and the generation of a new `Classbrowser.settings` file when you next close the Class Browser.

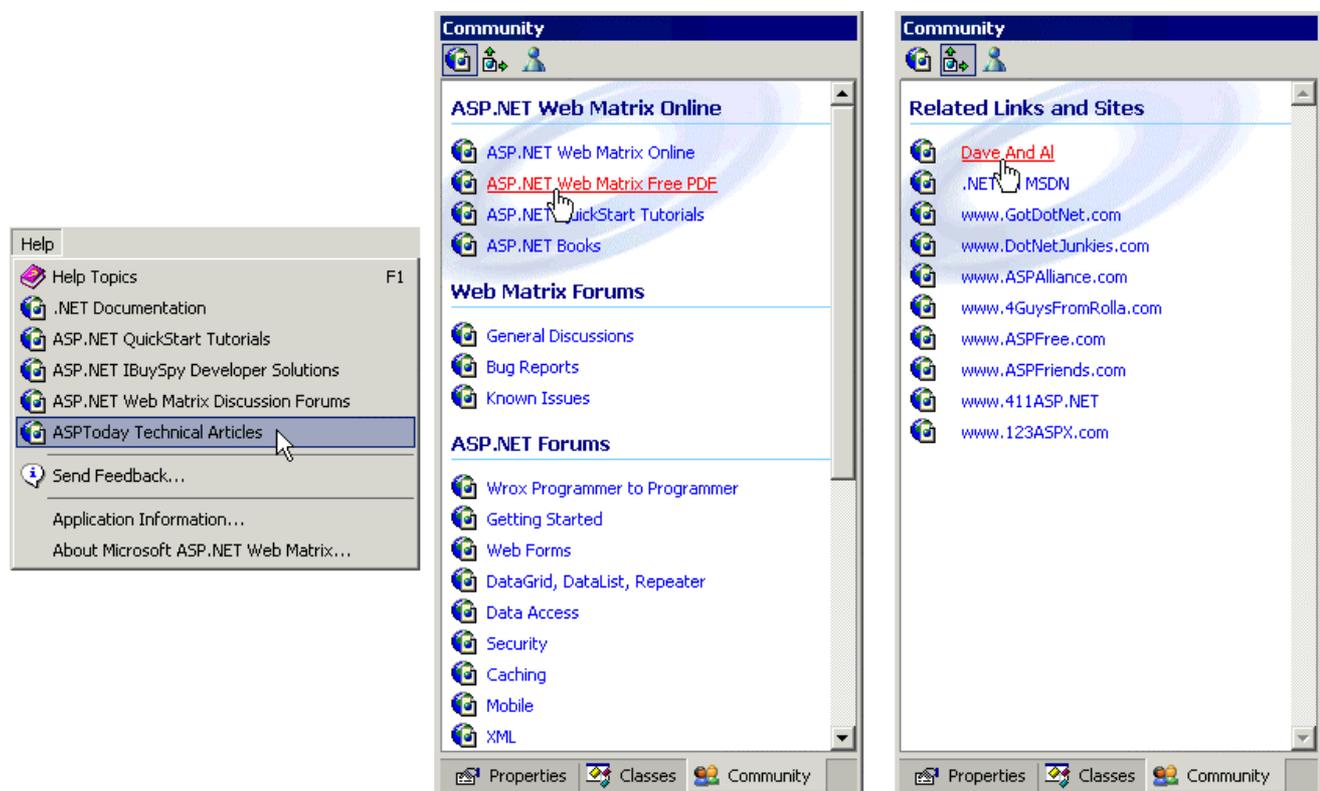
Customizing Help and Resource Links

Web Matrix provides links to useful resources in several places within the IDE. There are links to the Web Matrix team so that you can send e-mail, links to get help on the Help menu, and a host of links in the Community window. All these can be changed using the `WebMatrix.exe.config` file.

The <webLinks> section of the configuration file contains a list of <webLink> elements. The name attribute for each <webLink> element defines where the link is used. The options are:

- ❑ Help – the link appears in the main Help menu
- ❑ CommunityLink – the link appears in the top section of the first page in the Community window under the ASP.NET Web Matrix Online heading
- ❑ MxForumLink – the link appears in the first page in the Community window under the Web Matrix Forums heading
- ❑ ForumLink – the link appears in the top section of the first page in the Community window under the ASP.NET Forums heading
- ❑ NewsgroupLink – the link appears under the Newsgroups heading in the first page of the Community window
- ❑ ListservLink – the link appears under the Listservs heading in the first page of the Community window
- ❑ RelatedLink – the link appears under the Related Links and Sites heading in the second page of the Community window

You can see the Help menu and some of the headings listed above in the following screenshots, which show that the default list has been edited and a few of our own links have been added:



Each link type name (such as Help) has a number appended to it when used in the name attribute of the <webLink> element, so the Help menu links shown in the previous screenshot are created by the elements shown in the next listing – we added the last one ourselves:

```

<webLinks>
  <webLink name="Help0" title=".NET Documentation"
    url="http://msdn.microsoft.com/net"/>
  <webLink name="Help1" title="ASP.NET QuickStart Tutorials"
    url="http://www.asp.net/Tools/redirect.aspx?path=quickstart"/>
  <webLink name="Help2" title="ASP.NET IBuySpy Developer Solutions"
    url="http://www.asp.net/Tools/redirect.aspx?path=ibuyspy"/>
  <webLink name="Help3" title="ASP.NET Web Matrix Discussion Forums"

url="http://www.asp.net/Forums/ShowForumGroup.aspx?tabindex=1&ForumGroupID=4"/>
  <webLink name="Help4" title="ASPToday Technical Articles"
    url="http://www.asptoday.com"/>
  ...

```

Likewise, the extra links in the two pages of the Community window are created with additional <webLink> elements, which we also inserted into the <webLinks> section:

```

...
<webLink name="CommunityLink1" title="ASP.NET Web Matrix Free PDF"
  url="http://www.asp.net/WebMatrix/download/pdf"/>
...
<webLink name="ForumLink0" title="Wrox Programmer to Programmer"
  url="news://p2p.wrox.com/aspnet"/>
...
<webLink name="NewsgroupLink0" title="news.wrox.com"
  url="news://news.wrox.com/aspnet"/>
...
<webLink name="ListservLink0" title="ASP.NET Articles"
  url="news://ls.p2p.wrox.com/aspnetnews"/>
...
<webLink name="RelatedLink0" title="Dave And Al"
  url="http://www.daveandal.com"/>
...
</webLinks>

```

This allows you to insert your favorite links, as well as allowing you to change the order in which they are displayed.

Summary

As you can see – even from the very swift tour that we've provided in this document – ASP.NET Web Matrix is already a powerful, intuitive, and extremely useful tool for creating web sites and web pages, User Controls, Web Services, and many other types of ASP.NET-related project files. It combines the ease of use of tools like Visual Studio .NET with the simplicity of files created with a text editor or other third-party tools. By avoiding the code-behind approach of Visual Studio .NET, it also produces ASP.NET pages that are generally easier to debug, modify, and extend in the future without requiring the original development tool to be used.

In this document, we looked at the fundamental differences between Web Matrix and Visual Studio .NET, and then went on to tour the IDE itself. We explained the features provided, the templates and wizards that are included, and the way that you can get help and support directly from within the IDE. Remember that Web Matrix is a "community project" that will evolve over time in line with feature requests and feedback from users. The links that are provided to other resources help you to interact with the community at large (outside Microsoft) that has consistently driven the adoption of ASP and ASP.NET.

We went on to put Web Matrix to work and built an example application that demonstrates many of its features. While our application is by no means a real-world commercial application, it did allow us to show several types of files, wizards and other features that are part of Web Matrix. It also indicates just how much time and effort Web Matrix can save you when you are working with ASP.NET.

Finally, we looked at how you can configure and extend Web Matrix to suit your own requirements. This includes setting preferences as well as modifying the user interface content and layout. We also indicated that you could build your own add-ins and other tools that directly integrate with Web Matrix. There will be an increasing number of ready-built tools and add-ins available as Web Matrix matures and the user community grows. The focus for all this is the Microsoft ASP.NET Web Matrix site at <http://www.asp.net/WebMatrix>.

Bear in mind that this document looks at the Beta 1 version of Web Matrix, released in the summer of 2002. There is no fixed timetable for a release of the "final" product at the moment – in fact there may never be a "final" version. Only you, as a member of the user community, can help to influence and decide the ultimate fate of Web Matrix. Use it and enjoy it...



More WROX ASP Programmer Resources

Books

There's a whole heap of ASP and ASP.NET books in print from Wrox Press ~ visit <http://www.wrox.com> for detailed book breakdowns and sample code downloads. Our latest titles are proving themselves invaluable to professional and novice Active Server Page programmers:

On-Line

ASP Today
www.asptoday.com

Every day, 1000s of professional ASP developers dig themselves out of a jam by using the most respected code and method resource on the web:

www.ASPToday.com

This "Today" site has been going strong for 2 years and is packed full of technical articles, case studies, practical code and advice. ASPToday.com is the professional resource of choice, recommended by web rings, user groups, Microsoft, training companies and working programmers.

Drop in and sample a free article or two...there's a new one EVERY WORKING DAY!

When you're convinced ASP Today has the depth to aid your career ~ take advantage of a SPECIAL OFFER available to Web Matrix downloaders of:

**20% OFF ASPToday annual subscription
(\$79 instead of the standard \$99)**

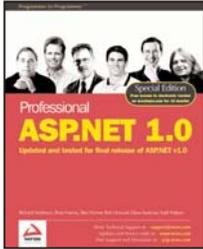


www.ASPToday.com

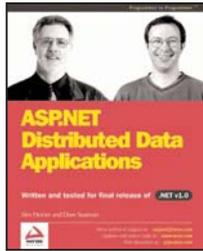
Getting Your Discount

To obtain your discounted annual subscription, you will first need to:

- 1) **Register** an email address and password with the **Wrox Developer Community**. Members of the WDC have access to free content on Wrox websites. You are a member if you have registered on ASPToday, CSharpToday or Wrox.com (for code download registration). Register here <http://www.wrox.com>
- 2) **Log in** by clicking on the link below. Enter your new WDC details then enter **matrix@wrox.com** at the prompt for 'referrer', follow online instructions to purchase 12 months access to ASPToday for \$79.
MATRIX SPECIAL OFFER



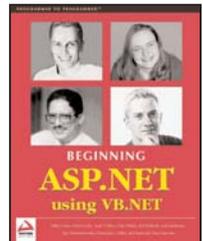
ISBN: 7035



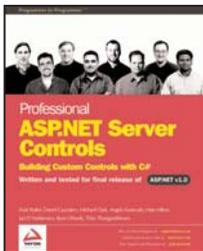
ISBN: 4923



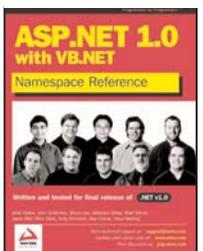
ISBN: 7248



ISBN: 5040



ISBN: 5644



ISBN: 7450

Programmers Lists

p2p.wrox.com
The programmer's resource center

All developers are invited to join the Wrox P2P (Programmer to Programmer) discussion lists. ASP is a favoured subject and it's broken out into dozens of specialised sub lists where fellow programmers and Wrox readers share experience, answers and work-a-rounds that can prove a real time saver. All technologies covered by Wrox are served in a discussion list...from ADO to XSLT. P2P is YOUR free community resource - Web Matrix list already started !

There's a lot more going on with Wrox than turning pages ...also check out our On-line library, Developers Journal Mail-Out, and Amazon E-Docs.

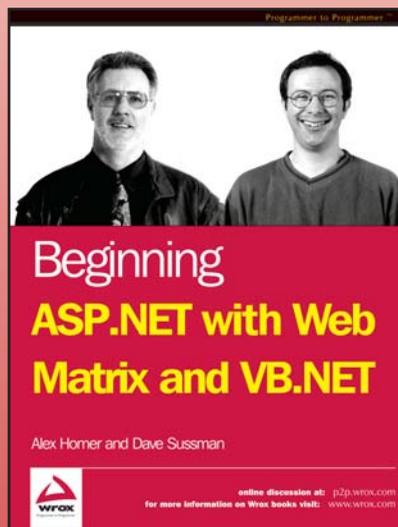
wroxbase
www.wroxbase.com

Amazon E-Docs

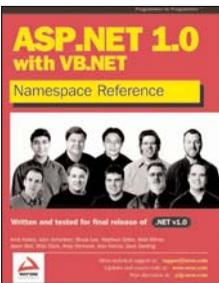
Our case studies and popular backlist titles

Developer Journal, our regular e-mail notifier to 100,000 programmers

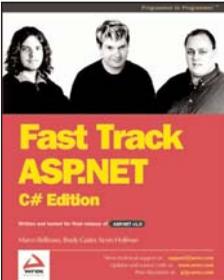
Coming Soon



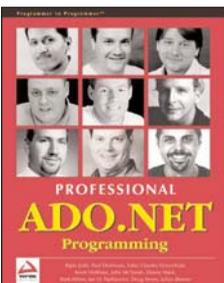
ISBN: 7922



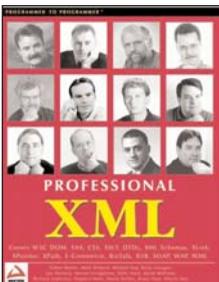
ISBN: 7450



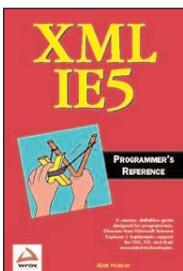
ISBN: 7191



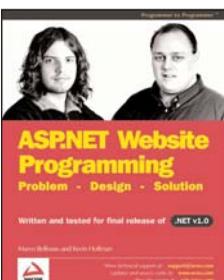
ISBN: 527X



ISBN: 3110



ISBN: 1576



ISBN: 6934

