

Insider Extra 2

ASP.NET Self-Diagnosis Web Page

This Insider Extra is a Microsoft ASP.NET page that gathers and reports a variety of information about your Web server that affects the operation of Microsoft Office FrontPage 2003. For the most part, it displays the same information as the ASP Self-Diagnosis Web Page described in Insider Extra 1. This page, however, presents an ASP.NET view of things, and it tests whether ASP.NET pages can run on your server.



On the CD

To retrieve similar information from the viewpoint of an ASP page, refer to Insider Extra 1, "ASP Self-Diagnosis Web Page."

Because this is an ASP.NET page, it will work only on Microsoft Web servers that have the Microsoft .NET Framework installed. In addition, you must install it in an executable folder with Script permissions (that is, a folder that can run ASP.NET pages). The Microsoft FrontPage Server Extensions need not be present.

The ASP.NET code for this Web page uses the *inline code model*. This means that the code appears inline with the HTML, all in one file. As a result, there's no need to add the page to a Microsoft Visual Studio .NET project, compile and link it, and then deploy the project.

Running the ASP.NET Self-Diagnosis Web Page

To run the ASP.NET Self-Diagnosis Web Page on your server:

- 1 Install the Insider Extra Web site from the Companion CD. The Sample Files setup program installs this site at [My documents]\Microsoft Press\FrontPage 2003 Inside Out\fp1\extras.
- 2 Copy the selfdiagnet/selfdiag.aspx page into your Web site on the server in question.
- 3 Browse the selfdiag.aspx page from the server location where you copied it. Figure IE2-1 shows some typical results.

Web Server			
Category	Property	Value	Comment
Web Location			
	Protocol	http	OK
	Host Name	collar.interlacken.com	
	IP Address	192.168.180.25	
	HTTP Root	/	c:\inetpub\wwwroot\
Application			
	ASP.NET	Working	OK
	Application Root	/fp11extras	c:\inetpub\wwwroot\fp11extras\
	User Domain	COLLAR	
	User Name	ASPNET	
Computer			
	Computer Name	COLLAR	
	Processor	x86 Family 6 Model 5 Stepping 1	
	Speed (MHz)	398	

Figure IE2-1. This ASP.NET page displays a variety of information that's useful for diagnosing the operation of a FrontPage-based Web site.

Tip Secure Your Self-Diagnosis Information

Don't let copies of the Self-Diagnosis Web Page remain on your public Web site where hackers and other mischief-makers can find it. The more information you give these people, the more problems you'll have.

Caution Because Microsoft Windows SharePoint Services blocks execution of inline code, this Insider Extra won't run in that environment. To gather comparable information, you'll need to contact the server's administrator.

The ASP.NET Self-Diagnosis Web Page gathers information in a variety of ways, including:

- Inspecting files within your Web site
- Reading values from the Web server's registry
- Loading and running various ActiveX objects

None of this is terribly intrusive; every day, ordinary ASP.NET pages and other routine processes access the same information that the ASP.NET Self-Diagnosis Web Page does. Nevertheless, it's possible that the ASP.NET Self-Diagnosis Web Page might falsely report some information as blank or missing because of security settings on your server. If you suspect that this is occurring, contact your server's administrator.

Interpreting Self-Diagnosis Information

This section explains all the information that the ASP.NET Self-Diagnosis Web Page provides. Most of this information is identical to what the ASP Self Diagnosis Web Page reports. Differences occur in the following areas:

- **Application**
 - **ASP.NET** This section replaces the Active Server Pages section on the ASP version of the page.
 - **User Domain** If the ASP.NET service account resides on the local machine, this field will contain the local computer name. The ASP version of the page displays nothing if the service account resides on the local machine.
- **Software**
 - **VB Version** This heading replaces the VBScript heading on the ASP page. It reports the installed version of Visual Basic .NET rather than the installed version of VBScript.
 - **.NET Framework Version** This heading doesn't appear on the ASP page. It reports the version of the .NET Framework in effect for the current page.
- **XML Objects** This heading is absent in the ASP.NET page. The ActiveX objects it identifies on the ASP page have no use to ASP.NET pages.
When ASP.NET pages need to process XML, they use methods in the .NET Framework rather than ActiveX controls.
- **AppSettings** This heading replaces the Database Connections heading on the ASP page.

The explanations that follow assume that you already know the basics of the Microsoft .NET Framework and ASP.NET. Although FrontPage can work with (and even create) ASP.NET pages, this remains a book about FrontPage, and not about .NET fundamentals.

Web Location

This group of values reports the Web server's location:

- **Protocol** This field identifies the network protocol you used to access the Web page. Correct values are *http* and *https*. If the protocol is *file*, the page will display an error message in the Comment column, warning you that because you're loading the page directly from disk, the ASP.NET code can't possibly run.
- **Host Name** This field reports the host name you specified in the URL that accessed the ASP.NET Self-Diagnosis Web Page.
- **IP Address** This field reports the server's network address.
- **HTTP Root** The Value column always displays a slash (/), which is always the location of the server's HTTP root. The Comment column shows the corresponding location in the Web server's file system, such as `c:\inetpub\wwwroot`.

Application

This group of values reports information about the Web application that contains the ASP.NET Self-Diagnosis Web Page:

- **ASP.NET** This value should always be Working. If it's blank, ASP.NET pages aren't working, and most other values on this page will be blank as well.
- **Application Root** The Value column displays the starting URL of the current IIS application. If this value doesn't equal the URL of your FrontPage-based Web site, database connections and other features that depend on the `global.asax` and `web.config` files are unlikely to work. The Comment column displays the corresponding physical path.
- **User Domain** This field reports the Windows domain in which the ASP.NET service account resides. If this field equals the Computer Name value under the Computer heading, ASP.NET pages are running under the authority of a local account on the Web server.
- **User Name** This field shows the name of the user account under which ASP.NET pages are running. This determines the security privileges that Windows allots to your ASP.NET pages.

Computer

This group of values reports information about the computer where the Web server is running:

- **Computer Name** This is the Windows name of the entire computer (that is, the name you would see on the Computer Name tab of the System Properties Control Panel applet).
- **Processor** This field reports the manufacturer's model identification for the first CPU in the server.
- **Speed (MHz)** This field reports the server's approximate CPU speed in MHz.

Software

This group of values reports the presence and version numbers of several kinds of software running on the Web server:

- **Operating System** This field reports the operating system that the server is running.
- **CSD Version** This field reports the service pack or other release level that the server is running.
- **Web Server Software** This field reports the name and version of the Web server software.
- **VB Version** This field reports the version of Microsoft Visual Basic .NET installed on the server, and therefore the version available to ASP.NET pages.

ASP.NET Self-Diagnosis Web Page

- **ADO Version** This field reports the version of ADO.NET installed on the server and available to ASP.NET pages. ASP.NET pages use ADO.NET to access databases.
- **MDAC Version** This field reports the most recent version of the Microsoft Data Access Components installed on the server. This is a collection of database drivers including, but not limited to, ADO.NET.
- **ListSystemDSNs** If this field is not available or non-zero, the server's Open Database Connectivity (ODBC) software will supply a list of System Data Source Names (DSNs) on demand.

If this field is 0, no such list is available. Administrators sometimes choose this value because if the list is available, it includes all System DSN's on the server, whether or not they belong to the current site. Suppressing the list makes it harder for the operator of one site to guess the location of another site's databases.

This setting affects FrontPage after you take these actions:

- 1 Choose Site Settings from the Tools menu.
- 2 Click the Database tab.
- 3 Click the Add button. This displays a New Database Connection dialog box.
- 4 Select System Data Source On Web Server.

At this point, if ListSystemDSNs value is Not Detectable or non-zero, you can click the Browse button and choose the DSN you want from a list.

If the ListSystemDSNs value is 0, the Browse option and the Name text box in the New Database Connection dialog box will be dimmed. Your options in this case are:

- Create the database connection some other way. For example, select the Custom Definition option, click the Advanced button, type a connection string in the Connection string field, and then click OK and verify.
 - Ask the server's administrator to change the ListSystemDSNs value for you. The full registrykey is HKEY_Local_Machine\Software\Microsoft\Shared Tools\Web Server Extensions\All Ports>ListSystemDSNs.
- **.NET Framework Version** This value reports the version of the .NET Framework that's executing the current Web page.

FrontPage

This group of values provides information about the FrontPage Server Extensions:

- **Server Extensions** This field reports the version of the FrontPage Server Extensions installed on the server, or reports Absent if the server extensions don't appear to be installed.

Microsoft Office FrontPage 2003 Inside Out

- **Current Web Site** This field reports the root folder location of the current FrontPage-based Web site.

If your Web site contains any FrontPage database connections, any ASP.NET pages that rely on the `global.asa` file, or any ASP.NET pages, this value and the Application Root value should be equal. If they're not, try adding a FrontPage database connection to your Web site. If that doesn't solve the problem, ask the server's administrator to flag your Web site as an IIS application.

Indexing Service

This group of values provides information about Microsoft Indexing Service. The presence or absence of Indexing Service affects the way that the FrontPage Web Search component works.

If Indexing Service is present, all Web Search components in all virtual servers on the computer will try to use it. This holds true even if Indexing Service isn't running, or if it's running but has no catalog that includes your site. In these latter cases, your search will, of course, produce no results.

If Indexing Service isn't present, all Web Search components will use the WAIS search engine that comes with the FrontPage Server Extensions.

There is, however, one exception to these rules. If Indexing Service is present, but a registry value named `NoIndexServer` equals `1`, all Web Search components will use the WAIS search engine (as if Indexing Service weren't present).

For more information about diagnosing problems with the Web Search component, refer to the Troubleshooting sidebar "Search Component Returns No Results," on page 740.

- **Indexing Service** This field displays one of four possible values:
 - **Unavailable** Means that Indexing Service doesn't seem to be installed.
 - **Installed** Means that Indexing Service seems to be installed, but that the ASP.NET Self-Diagnosis Web Page can't determine its status.
 - **Running** Means that Indexing Service is installed and running.
 - **Stopped** Means that Indexing Service is installed but not running.
- **Indexing Service Startup** This field displays one of four possible values:
 - **Not Detectable** Means that the ASP.NET Self-Diagnosis Web Page can't determine when Indexing Service will start. This usually means that Indexing Service isn't installed.
 - **Automatic** Means that Indexing Service will start automatically every time the server boots.
 - **Manual** Means that Indexing Service will start only when an administrator issues a command.
 - **Disabled** Means that Indexing Service will never start.

ASP.NET Self-Diagnosis Web Page

- **NoIndexServer** This field reports the value of a registry entry that overrides the normal logic for using Indexing Service:
 - If this value is 1, all FrontPage Web Search components will use the FrontPage search engine even if Indexing Service is present.
 - If this value is Not Detectable, the NoIndexServer registry value is probably absent. In this case, if Indexing Service is installed, Web Search components will use it.
 - If this value is anything else, all FrontPage Web Search components will use Microsoft Indexing Service, even if Indexing Service isn't running or has no catalog for the Web site that contains the Web Search component.



On the CD

To perform further testing on Microsoft Indexing Service, try Insider Extra 7, "Scripting Web Searches."

E-Mail

This heading reports the status of an object named *CDONTS.NewMail*. ASP.NET and ASP.NET pages frequently use this object to send e-mail.

- **CDO Version** If this value is Unavailable, the *CDONTS.NewMail* object is either not installed or inaccessible because of security restrictions. If a version number appears, the given version of the object is available for use.

Note that for ASP.NET pages to successfully send mail, the Microsoft SMTP Service must also be properly configured and running.

AppSettings

This heading identifies a list of all settings currently defined in the <AppSettings> section of your application's web.config file. When you create a Database Results Region that uses ASP.NET, FrontPage stores the connection information as several distinct values in this section of the web.config file.

In the interest of completeness, the ASP.NET Self-Diagnosis Web Page lists all the AppSettings entries in your application. The database connection entries begin with the connection name and an underscore (_). In addition, the ASP.NET Self-Diagnosis Web Page uses shading to group all the entries for the same connection. Figure IE2-2 provides an example.

Microsoft Office FrontPage 2003 Inside Out

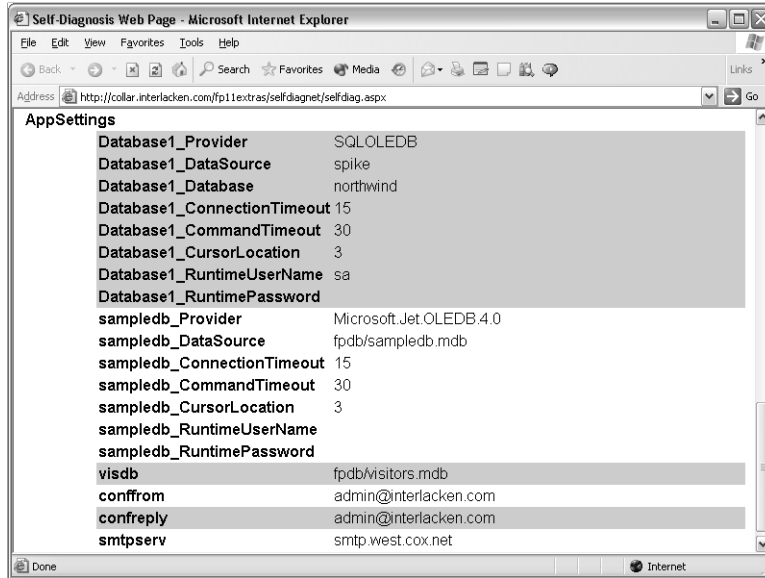


Figure IE2-2. The ASP.NET Self-Diagnosis Web Page displays all AppSettings for the current application. Shading groups settings for the same database connection.