

# Introduction to ASP.NET AJAX

## Creating and Hosting an ASP.NET AJAX Web Site in Your Windows Shared Hosting Account

ASP.NET AJAX is a free framework that assists you in quickly creating interactive Web applications. This tutorial shows you how to create and host an ASP.NET AJAX Web site in your Windows shared hosting account.

## Download and Install ASP.NET AJAX and ASP.NET AJAX Control Toolkit

To use ASP.NET AJAX you need to download and install the free framework to your local computer. This will add new project templates for creating AJAX-enabled Web sites and new tools in the Toolbox area of Visual Studio 2008.

Download and Install [ASP.NET AJAX](#).

The ASP.NET AJAX Control Toolkit is an open source project that contains a number of AJAX-enabled controls and extenders. If you intend to use these controls, you need to download the runtime binaries to your local computer and add a reference to your Visual Studio 2005 or 2008 project.

## Adding the ASP.NET AJAX Control Toolkit Reference to Your Web Site

1. Download the [ASP.NET AJAX Control Toolkit](#).
2. Extract the files from the archive to a folder on your computer. For example: *C:\AJAXControlToolkit*.
3. Right-click on your project in the Solution Explorer and select **Add Reference**.
4. Select the **Browse** tab and select *AjaxControlToolkit.dll* from the folder where you extracted the toolkit. For example: *C:\AJAXControlToolkit\SampleWebSite\Bin*.
5. Click **OK**.

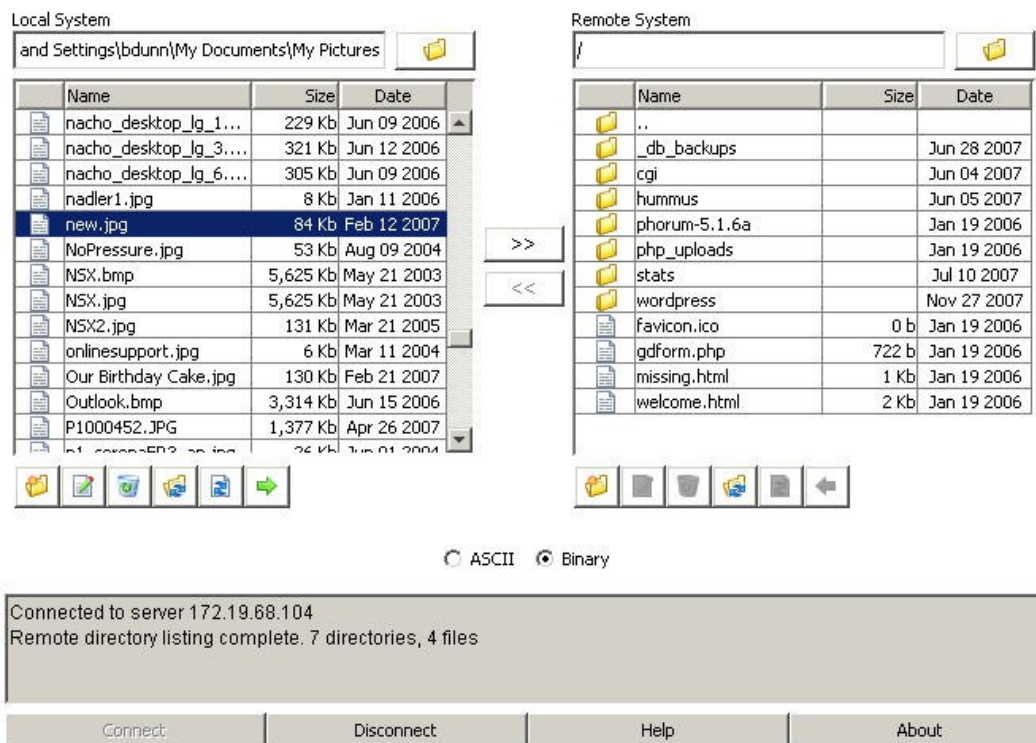
**NOTE:** It is important that this DLL is copied to the bin directory of your Shared Hosting Account when deploying.

## Configuring Your Windows Hosting Account

Before you set up your Web site, decide how you want to manage it. The Hosting Control Center allows you to manage your Web site using an FTP application or using FrontPage.

### FTP

You use an FTP client to publish content to your Web site. You can use your own FTP client or the Web-based FTP client provided in the Hosting Control Center.



You can also create directories and set permissions for them using the IIS Settings feature.

**NOTE:** The IIS Settings feature is only available when FrontPage Extensions is disabled.

### FrontPage Extensions

If you prefer to manage your Web site using FrontPage, you must enable FrontPage Extensions through the **FrontPage Extensions** feature located in the **Settings** section of the Hosting Control Center.

**FrontPage Extensions**

**Current Install Status** Not Installed

**Select Installation Option**  Install FrontPage Extensions

**FrontPage Email Address**

**Enter FrontPage Email Address**

**Continue** **Cancel**

FrontPage Extensions manage all of the directory and subdirectory permissions on your Web site. If you created custom directories on your Web site, or you are using another tool to build and or manage your Web site, switching to FrontPage may cause some problems. It overwrites your directory permissions and removes all application roots that have been configured through the File Manager. It does not, however, remove application roots that have been created for a subdomain.

When using FrontPage extensions, you must publish your Web site with an application that supports FrontPage extensions, such as FrontPage or Visual Studio.

**NOTE:** To create an application root, you must create a subdomain and point it to the directory you would like to be set up as an application root.

## Setting Up Your Web Site Using FTP

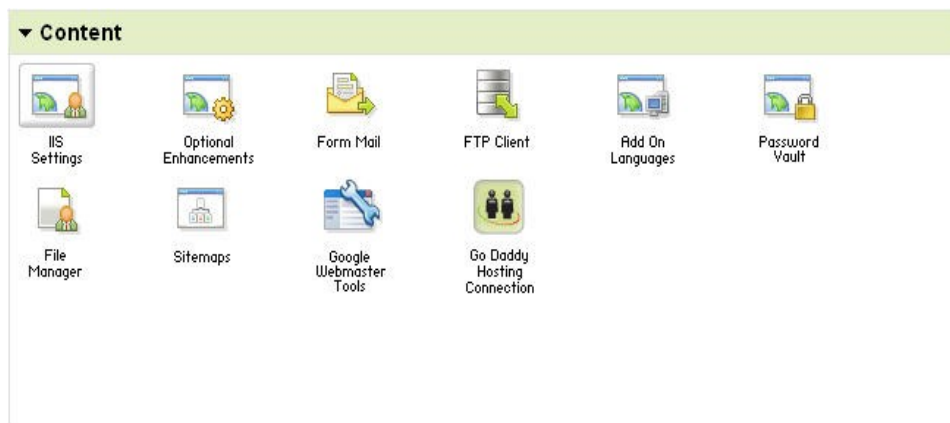
Verify that FrontPage Extensions have been disabled through the FrontPage Extensions tool located in the Hosting Control Panel. If your Web site is not going to reside in the root directory you need to create an application root directory for your Web site. If it is going to reside in the root directory, you can skip the following steps.

## To Create an Application Root Directory

1. Log in to your [Account Manager](#).
2. In the **My Products** section, select **Hosting Account List**.
3. Next to the hosting account you want to modify, click **Open**.



4. In the **Content** section of the Hosting Control Center, click the **IIS Settings** icon.



5. Click the directory name to update settings on an existing directory. To set up a new directory, click **Create Directory**.



6. Type a directory name, select the settings you want to apply to the directory, and click **Continue**. Since you are setting this directory as an application root, make sure the **Web** and **Set Root** options are selected.

**Step 1**

Enter a Name

**Step 2**

Select Permissions

Read  Web  Browse  Write  Set Root

7. Verify the settings for this directory. If the settings listed are correct, click **Continue**.

## Creating ASP.NET AJAX Enabled Pages

To enable partial-page updates via AJAX you will need to add the script manager control to your page. You can do this by dragging and dropping the ScriptManager control to your page in design view or by adding the following to your page source:

```
<asp:ScriptManager ID="ScriptManager1" runat="server" />
```

For more information on using ASP.NET AJAX, you can refer to the ASP.NET AJAX online Documentation at [ASP.NET AJAX Online Documentation](#).

## Developing & Deploying an ASP.NET AJAX Web Application

**NOTE:** You will need to download and install [Visual Studio Service Pack](#) if you want to use the Web Application Project templates.

## To Create an ASP.NET AJAX Web Application in Visual Studio

1. Click **New** from the **File** menu.
2. Select the desired language.
3. Select the **ASP.NET AJAX-Enabled Web Application** project template
4. Enter a Name and Location for your project.
5. Click **OK**.

## To Deploy an ASP.NET AJAX Web Application

1. Select **Publish** from the **Build** menu.
2. Enter the appropriate FTP path. For example: *ftp://www.coolexample.com* or *ftp://www.coolexample.com/webapp*.

**NOTE:** This path must have the Set Root permission set in the IIS Settings feature of the Hosting Control Center.

3. Configure the other options as necessary.
4. Click **Publish**.

## Developing & Deploying an ASP.NET AJAX Web Site

### Creating an ASP.NET AJAX Web Site

1. Select **New Web Site** from the **File** menu.
2. Select **ASP.NET AJAX-Enabled Web Site** project template.
3. Set the location to **File System** and enter the appropriate path. For example: *C:\mysite*. Alternatively, you can connect directly to the server by setting the location to **FTP** and entering the appropriate path. For example: *ftp://www.mysite.com*.

4. Select the desired language.
5. Click **OK**.

### Deploying an ASP.NET AJAX Web Site

1. Click **Copy Web Site** from the **Website** menu.
2. Click **Connect** in the top tool bar.
3. Select **FTP Site**.
4. Enter the appropriate server name. For example: *ftp://www.mysite.com*.
5. Enter port **21**.
6. Enter the directory you want to deploy to or leave it blank for root. For example: *Webapp*.

**NOTE:** If you enter a directory here, you must make sure you have created the virtual directory in the Hosting Control Center as described above.

7. Make sure **Passive Mode** and **Anonymous Login** are unchecked.
8. Enter your hosting account user name and password.
9. Click **Open**.
10. If the directory you entered does not exist, you are prompted to create it.

### Connecting to an Existing ASP.NET AJAX Website

1. Click **Open Web Site** from the **File** menu.
2. Select **FTP Site**.
3. Enter the appropriate server name. For example: *ftp://www.mysite.com*.
4. Enter port **21**.
5. Enter the directory you want to deploy to or leave it blank for root. For example: *Webapp*.

**NOTE:** If you enter a directory here, you must make sure you have created the virtual directory in the Hosting Control Center as described above.

6. Make sure **Passive Mode** and **Anonymous Login** are unchecked.
7. Enter your hosting account user name and password.