

Web Development: Before you begin

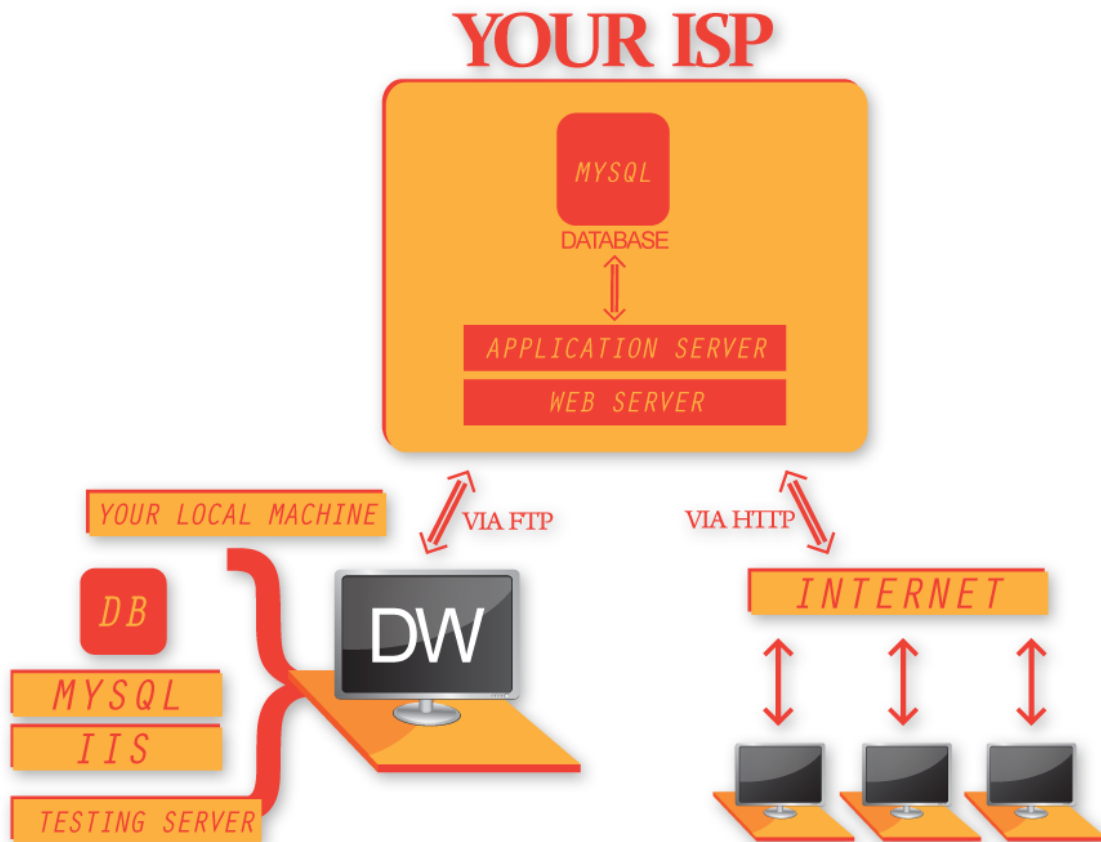
Preparing your system for dynamic development

Ready to make the jump to Web Development? As a Web designer, you have needed to grasp a wide range of front end tools and concepts to stay competitive—including graphics editing software, Dreamweaver, HTML, CSS, web standards, and maybe even a little Ajax, to name a few. There are a number of reasons to make the move to building dynamic websites. By structuring your sites to interact with a database, you can free yourself of constant site maintenance updates, while also creating added value in your ability to build blogs, ecommerce stores, content management systems and forums. Developing web applications that are driven by server side scripts and a database (such as PHP and MySQL) is within reach, and the transition for you is possible.

WebAssist is uniquely positioned to help you through this learning process by providing several extensions to speed up your development time—dramatically shortening the time it takes to build applications from scratch. This document outlines the five steps you will need to take to get your system ready to develop and stage your first web application:

1. Web Server (IIS)
2. Application Server (PHP)
3. Database (MySQL)
4. Defining a PHP site & Testing Server (Dreamweaver)

Performing these steps will help you gain a better understanding of the underlying architecture for web application development and deployment.



Step #1 - Web Servers

Web servers are literally the computers that serve up content for websites. The term "Web Server" also refers to the software that runs on these computers that accepts requests from web browsers, and delivers files and pages back to the end users requesting the information. These servers are also referred to as HTTP servers. The two most commonly used web server software packages are [Apache](#) and [Microsoft Internet Information Server](#). You can install and use this web server software on your computer as well. If you are a Macintosh user, you will need to use Apache (shipped installed on your machine), and IIS is standard on Windows machines.



When using IIS to develop a web application, by default the name of your web server is the name of your computer. If you have not named your computer, the IIS web server will use 'localhost' as the name.

The server name corresponds to the root folder of the server; for IIS, it is C:\Inetpub\wwwroot\. You can open any web page placed in this root folder by using the following URL in a web browser on your machine:
http://your_server_name/your_file_name

Installing Internet Information Server: Windows

Administrative Credentials - You must be logged in as the Administrator on your machine in order to perform this procedure.

Installing IIS on Windows XP (Vista users see next section below)

If you're a Windows XP Professional user, check whether IIS is already installed and running on your system. The way to do this is to look for the C:\Inetpub folder. If it doesn't exist, IIS is probably not installed on your system. To add it:

1. Go to Start > Control Panel > Add or Remove Programs.
2. Click on "Add/Remove Windows Components" on the left.
3. In the Windows Components Wizard that opens find Internet Information Services (IIS) and click on the check box next to it.
4. Click "Next" and the features will be enabled and installed.
5. After IIS has been enabled a restart of Windows may be in order. While this step is not always necessary I find it good practice to restart Windows when enabling new services.

Installing IIS on Windows Vista

Optional: Turn off UAC (Universal Access Control) - UAC requires that when making system changes or installing application you are required to

enter an administrator's password. By turning off UAC during this process will make the process much quicker and easier. You can always turn it back on after completing this process.

1. Go to Start > Control Panel then click on "Classic View" on the left.
2. Open Programs and Features.
3. Click on "Turn Windows features on or off" on the left.
4. In the Windows Features box that opens find Internet Information Services and click on the check box next to it.
5. Click OK and the feature will be enabled.
6. If you turned off UAC during this process and would like to turn it back on you can do that now.
7. After IIS has been enabled a restart of Windows may be in order. While this step is not always necessary I find it good practice to restart Windows when enabling new services.

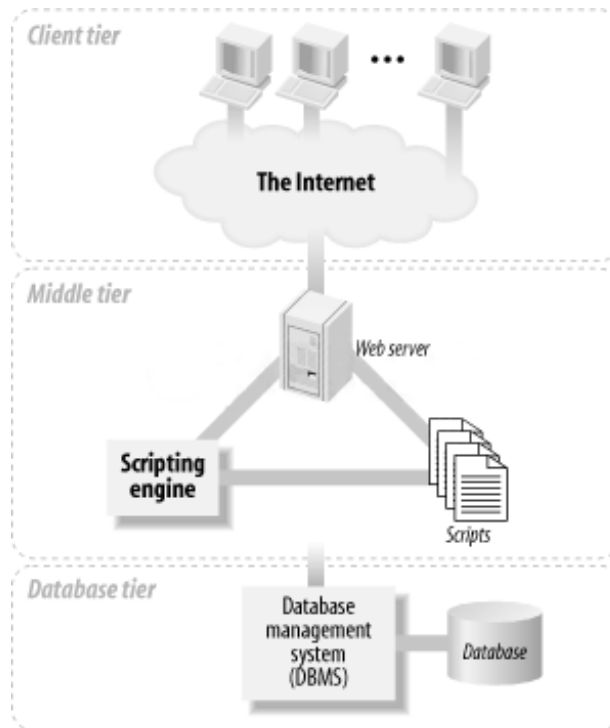
Testing the Internet Information Server (IIS)

You can now test to see if your IIS is running. Open a browser window and type: <http://localhost/>. You should see an IIS welcome screen. The <http://localhost/> address is the address mapped to the webroot of the server; by default this directory is located at C:\inetpub\wwwroot. There are two files in the wwwroot directory by default, iisstart.htm and welcome.png. You can create sub-directories in the webroot to establish individual sites on the server. For example if you create a webassist directory in wwwroot you can save html files to it and browse those files by going to http://localhost/webassist/your_files_here.html

Please see Microsoft's website for more information about [IIS](#).

Step #2 – Application Servers

The next step is to set up your application server, in this case PHP. An application server is software that helps process web pages containing server-side scripts or tags. When a dynamic page is requested by a browser, the web server hands the page off to the application server, which then executes the server-side scripts on the page, and/or requests specific information from the database. Application Servers are the backbone of a three-tiered architecture, which includes the client (front end), application server (middleware) and database (backend).



Installing an Application Server locally (PHP)

This section describes how to install PHP 5 on your Windows computer. PHP is open source software that you can use for developing, testing, and deploying web applications. For more information, see the PHP documentation at www.php.net/download-docs.php.

Installing PHP on XP (if you are using Vista see below)

1. Download the PHP 5.2.10 zip file at <http://www.php.net/downloads.php>

Note: These instructions only work for PHP version 5.2. PHP 5.3.0 requires different instructions that are not covered in this document.

2. Extract the PHP zip file to your hard drive, C:\php is what will be used here.
3. Create a temp directory in the php install directory, this will be used to store session data. The path to this folder should be C:\php\temp.
4. Add the php directories to the path system variable.
 - . Right click on My Computer and go to Properties.

- . Click the "Advanced" tab.
- . Under the Advanced Tab of System Properties click on the Environment Variables button.
- . In the System Variables area select Path and click edit.
- . Without changing any of the existing values add in the following paths, make sure to have the ; between the paths:
;C:\php;C:\php\dev;C:\php\ext

Close the Environment Variables window.

5. In the C:\php directory, copy php.ini-recommended to C:\Windows and rename it to php.ini. This is the configuration file for php.

Note: Now that you have both your Web server and your application server installed, you will need to configure them to work seamlessly together.

Configuring PHP on XP

During this process, it is easiest to search for settings using Ctrl F. Make sure that each time you search, your cursor is at the top of the page so that you search the entire document for the settings.

1. Open the php.ini file in notepad or any other text editing program to adjust some settings.
2. Enable display errors, look for the display_errors setting and make sure it is set to display_errors = On. This ensures that any errors are written to the screen so you will have immediate knowledge of any problem. Make sure there is not a ; before the setting.
3. Turn the cgi.force_redirect off, the value should look like this:
 cgi.force_redirect = 0, make sure to remove the ; at the beginning of the line.
4. Set the session.save_path to the temp directory, the value should look like this: session.save_path = "C:\php\temp", make sure to remove the ; at the beginning of the line.
5. Set the extension directory to the proper location. The setting is called extension_dir and should be set to "C:\php\ext".
6. Enable MySQL and other extensions by removing the ; before the extension name in the list. The extension for MySQL should look like this: extension=php_mysql.dll.
7. Turn all error reporting on. This is important so that you can catch any errors that may be occurring on your php pages. In the ini file do a search for the setting error_reporting and set it to error_reporting = E_ALL, make sure there is not a ; before the setting.

Save the php.ini file.

Configuring IIS on XP

1. Open the IIS manager by going to Start > Control Panel > Administrative Tools > Internet Information Services.
 2. In Internet Information Services go to your computer > Web Sites; right click on Default Web Site and select Properties.
 3. Click on the "Home Directory" tab and then click on the "Configuration" button.
 4. In the Application Configuration window select the "Mappings" tab and click "Add".
 5. In the interface that opens enter the following values
 - . Executable: C:\php\php-cgi.exe
 - . Extension: .php
 - . Verbs: All Verbs
 - . Checked: Script Engine
 - . Checked: Check that file exists
 6. Click OK, then Apply, in all of the dialog interfaces until you are back at Internet Information Services.
- Restart IIS to ensure there are no errors.

Test PHP on XP

You can now test to see if PHP is running. First create a php info page by opening up notepad or any other text editor. In the new document write the following line of code:

```
<?php phpinfo(); ?>
```

Save the document in C:\inetpub\wwwroot (the default webroot) as phpinfo.php. Open a browser window and type: <http://localhost/phpinfo.php>. If all things went correctly you should see a PHP information page. The information page shows all of the configuration info for PHP; if you enabled MySQL or any other extensions you should see them listed in the page.

Installing PHP on Vista

1. Download the PHP 5.2.10 zip file at <http://www.php.net/downloads.php>.

Note: These instructions only work for PHP version 5.2. PHP 5.3.0 requires different instructions that are not covered in this document.

2. Extract the PHP zip file to your hard drive, C:\php is what will be used here.
3. Create a temp directory in the php install directory, this will be used to store session data. The temp directory should be C:\php\temp.
4. Add the php directories to the path system variable.
 - . Right click on Computer and go to Properties.
 - . Click "Advanced System Settings" on the left.
 - . Under the Advanced Tab of System Properties click on the Environment Variables button.
 - . In the System Variables area select Path and click edit.
 - . Without changing any of the existing values add in the following paths, make sure to have the ; between the paths:
;C:\php;C:\php\dev;C:\php\ext
5. Close the Environment Variables window.
6. In the C:\php directory, copy php.ini-recommended to C:\Windows and rename it to php.ini. This is the configuration file for php.

Configuring PHP for Vista

During this process, it is easiest to search for settings using Ctrl F. Make sure that each time you search, your cursor is at the top of the page so that you search the entire document for the settings.

1. Open the php.ini file in notepad or any other text editing program to adjust some settings.
2. Enable display errors, look for the display_errors setting and make sure it is set to display_errors = On. This ensures that any errors are written to the screen so you will have immediate knowledge of any problem. Make sure there is not a ; before the setting.
3. Turn the cgi.force_redirect off, the value should look like this:
cgi.force_redirect = 0, make sure to remove the ; at the beginning of the line.
4. Set the session.save_path to the temp directory, the value should look like this: session.save_path = "C:\php\temp", make sure to remove the ; at the beginning of the line.
5. Set the extension directory to the proper location. The setting is called extension_dir and should be set to "C:\php\ext".
6. Enable MySQL and other extensions by removing the ; before the extension name in the list. The extension for MySQL should look like this: extension=php_mysql.dll.
7. Turn all error reporting on. This is important so that you can catch any errors that may be occurring on your php pages. In the ini file do a search for the setting error_reporting and set it to error_reporting = E_ALL, make sure there is not a ; before the setting.
8. Save the php.ini file.

Configuring IIS for Vista

1. Open the IIS manager by going to Start > Control Panel > Administrative Tools > Internet Information Services.
2. In IIS Manager open up Handler Mappings in the IIS area.
3. Click on "Add Script Map..." in the upper right of the window.
4. In the interface that opens enter the following values
 - . Request path: *.php
 - . Executable: C:\php\php5isapi.dll
 - . Name: PHP
5. Click OK, the Add Script Map dialog appears and ask if you would like to enable this ISAPI extension, click yes if you want to give permission to the php module to run on IIS.
6. Restart IIS to ensure there are no errors.

Test PHP for Vista

You can now test to see if PHP is running. First create a php info page by opening up notepad or any other text editor. In the new document write the following line of code:

```
<?php phpinfo(); ?>
```

Save the document in C:\inetpub\wwwroot (the default webroot) as phpinfo.php. Open a browser window and type: <http://localhost/phpinfo.php>. If all things went correctly you should see a PHP information page. The information page shows all of the configuration info for PHP; if you enabled MySQL or any other extensions you should see them listed in the page.

If you encounter an error that reads "HTTP Error 404.3 - Not Found" with the words "Handler: StaticFile" in the error message you will need to reset the static handling. To do so please follow these steps.

1. Go to Start > Control Panel. Click Classic View on the left and open Programs and Features.
2. Click on Turn Windows Features on or off on the left.
3. In the Turn Windows Features on or off control find Internet Information Services.
4. From here go to Internet Information Services > World Wide Web Services > Common Http Features > Static Content.
5. Examine the Static Content feature. If it is not checked, check it now and click OK. If it is checked un-check it, click OK, then repeat steps 1-5 to re-enable the feature.

Step #3 – The Database (MySQL)

A database defines a structure for storing information. Databases are typically organized into tables, which are collections of related items. You can think of a table as a grid of columns and rows similar to an Excel spreadsheet.

The instruction to extract data from a database is called a *database query*. A query consists of search criteria expressed in a database language called SQL (Structured Query Language). The SQL query is written into the page's server-side scripts or tags. The query is executed against the database, and a recordset is created. A *recordset* is a set of data extracted from one or more tables in a database. The recordset is returned to the application server, which uses the data to complete the page.

You can use a MySQL database server running on your computer to develop PHP applications in Dreamweaver. This section describes how to install the MySQL Community Server 5.0 on your Windows XP machine. MySQL is open source software that you can use for developing and testing web applications. For more information, see the MySQL documentation at <http://dev.mysql.com/doc/refman/5.0/en/index.html>.

MySQL install sets up a service that runs in the background that allows you and programs such as Dreamweaver to interact with your MySQL database. There are two ways to interact with this needed service: through the command line interface (which can be challenging) or through the MySQL GUI tools. There are a few other tools that can help you with this process but we find the MySQL GUI tools to be quick and easy. You can download them here: <http://dev.mysql.com/downloads/gui-tools/5.0.html>

Once you have downloaded the software, please follow the instructions listed below to get MySQL set up and running.

Installing MySQL on Windows (both XP and Vista)

This section describes how to download and install the MySQL 5.0 Community Edition database server on your Windows XP computer.

Note: The instructions in this section apply to the "Windows Essentials" installer for MySQL 5.0 in Windows XP. For other versions, please see the MySQL documentation for possible differences in the installation procedure.

1. Download the "Windows Essentials (x86)" installer from the Windows Downloads category on the MySQL download page:
<http://dev.mysql.com/downloads/mysql/5.0.html#win32>
2. Double-click the MSI file you downloaded and follow the onscreen instructions.

Note: Select Typical Install when prompted.

3. When the setup wizard is done, run the configuration wizard by making sure the Configure the MySQL Server Now option is selected on the setup wizard's last page and clicking Finish.

Note: If you missed the option in the setup wizard, you can open the configuration wizard by selecting Start > All Programs > MySQL > MySQL Server 5.x > MySQL Server Instance Config Wizard.

4. In the MySQL Server Instance Configuration Wizard, make the following choices:
 - For configuration type, select Detailed Configuration.
 - For server type, select Developer Machine
 - For database usage, select Non-Transactional Database Only.
 - For the number of concurrent connections, select Decision Support(DSS)/OLAP.
 - For networking options, accept the default settings.
 - For the default character set, accept the default setting.
 - For the Windows options, select both options – Install As Windows Service, and Include Bin Directory in Windows Path.
 - For security options, enter a root password and confirm it.
5. Click Execute to configure your database server. Now you can verify that the server is working correctly. Now use the MySQL administrator tool to connect to and test the MySQL install.
6. From the Start menu, choose Programs > MySQL and select the MySQL Administrator.
7. Log in using the credentials you set up earlier.
8. Choose Catalogs, and double-click one of the default schemas that are provided. This confirms that your installation is correct and that you are ready to test your PHP/MySQL pages.

Note: The MySQL Administrator refers to individual databases as *schemas*.

How to test your PHP/MySQL pages (both XP and Vista)

You should now be able to create and test all your PHP/MySQL pages on your home machine. We have utilized the already provided local web directory, wwwroot. When creating and testing your pages, simply store them in this folder, and then test the page in your browser.

1. Open your **C: drive**.
2. Open **inetpub**.
3. Open **wwwroot**.

The wwwroot folder is your testing server folder that you will need to store your PHP/MySQL pages for testing.

4. Open your browser.

Note: When viewing pages stored in your wwwroot folder, the URL to enter in browser is <http://localhost>.

5. In the URL field, enter <http://localhost> and then the name of your php page or directory.

If you are using Dreamweaver, it is recommended that you configure this directory as your Testing Server when defining a site. This will allow you to simply click F12 to preview your page in your browser.

Step #4 – Defining a Site & Testing Server

If you plan to develop dynamic pages, Dreamweaver needs the services of a testing server to generate and display dynamic content while you work. The testing server can be your local computer, known as <http://localhost> (see Step #1 – Web Servers), a development server, a staging server, or a production server. The development, staging and production servers are all remote servers, and are not your local machine.

Before you set up a testing server folder, you must define a local and remote folder. You can often use the settings of your remote folder for your testing server because dynamic pages placed in the remote folder can normally be processed by an application server.

Setting up Dreamweaver (both XP and Vista)

1. Select one of the following:
 - To set up a new site, Select Site > New Site.
 - To edit the setup of an existing site on your local disk or a remote site (or a branch of a remote site), whether it was created in Dreamweaver or not, select Site > Manage Sites, select your site from the list, and click Edit.

Note: If you haven't defined any Dreamweaver sites, the Site Definition dialog box appears and you don't need to click New.

2. Select Testing Server in the Advanced tab of the Site Definition dialog box.
3. Select the server technology you want to use for your web application.
4. Accept the default Testing Server settings or enter different settings.

By default, Dreamweaver assumes an application server is running on the same system as your web server. If you defined a remote folder in the Remote Info category in the Site Definition dialog box, and if an application server runs on the same system as the remote folder (including your local computer), accept the default settings in the Testing Server category.

If you did not define a remote folder in the Remote Info category, the Testing Server category defaults to the local folder you defined in the Local Info category. You can leave this setting alone only if you meet the following two conditions:

Your web server and application server both run on your local computer. For example, if you're a PHP developer running both IIS and PHP on your local Windows system.

Your local root folder is a subfolder of your website's home directory. For example, if you're using IIS, your local root folder is a subfolder of the c:\inetpub\wwwroot\ folder, or the folder itself.

If your local root folder is not a subfolder of your home directory, you must define the local root folder as a virtual directory in your web server.

Note: The application server can run on a different system than your remote site. For example, if the Access option you selected in the Remote Info category is Microsoft Visual SourceSafe, then you must specify a different server in the Testing Server category.

5. In the URL Prefix text box, enter the URL that users type in their browsers to open your web application, but do not include any filename.

The URL prefix comprises the domain name and any of your website's home directory's subdirectories or virtual directories, but not a filename. For example, if your application's URL is `www.adobe.com/mycoolapp/start.jsp`, you would enter the following URL prefix: `www.adobe.com/mycoolapp/`.

If Dreamweaver runs on the same system as your web server, you can use one of the *localhost* options as a placeholder for your domain name. For example, if you run IIS and your application's URL is `http://buttercup_pc/mycoolapp/start.jsp`, enter the following URL prefix:

```
http://localhost/mycoolapp/
```

For a list of localhost options that pertain to different web servers, see the next section.

6. Click OK, and then click Done to add the new site to your list of sites.

Congratulations! Now that you have completed these steps, your computer and Dreamweaver are enabled to work with you in building dynamic web sites.