

Installing - Apache - MySQL - PHP - phpMyAdmin On Windows XP

Introduction

First and foremost this tutorial is aimed at the person who is setting up a web server on their own computer, for learning and/or development purposes, for the very first time. Wherever possible I will keep the instructions as simple as they can be kept; where this is not possible I will endeavour to explain fully the concepts at hand in a step by step manner. This is being written as the result of bitter experience, and the frustrations caused going through this process myself.

The packages that are needed to construct our fully functioning Server are:-

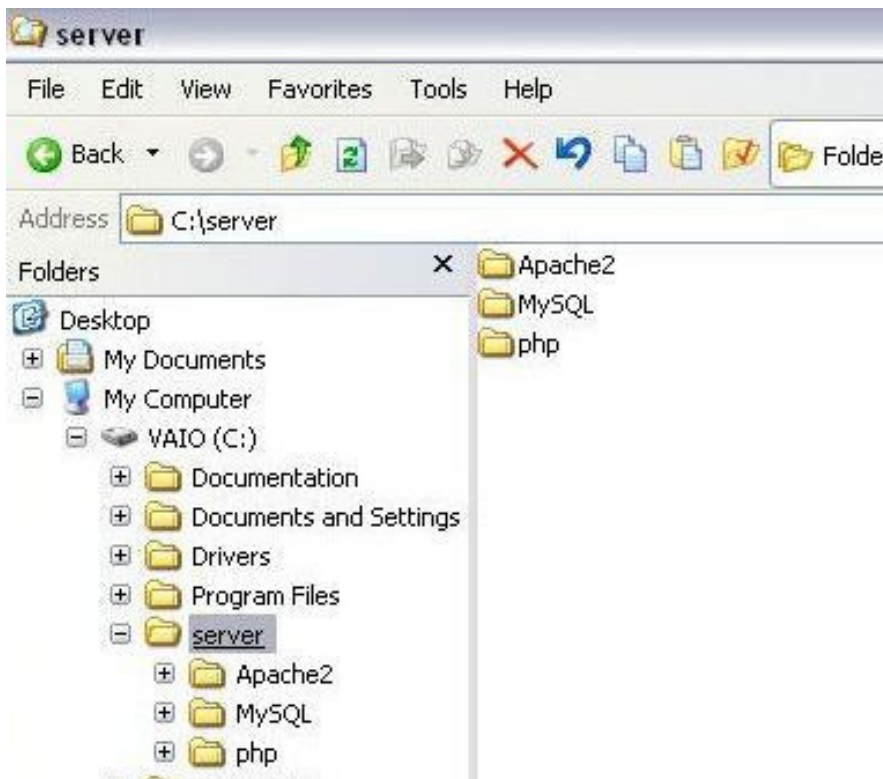
- ✓ Apache 2.2_9 - [download here](#)
- ✓ MySQL 5.0.5 - [download here](#)
- ✓ MySQL GUI Tools - [download here](#)
- ✓ PHP 5.2.6 - [download here](#)

Note: Please ensure that you download the .zip package and **NOT** the one with the Windows installer (this is useless for our purposes)

- ✓ PhpMyAdmin 2.11.7 - [download here](#)

I recommend creating a folder on the desktop to store these until needed, I named mine “server”, I also suggest downloading the most up to date version of each package usually at the top of the list.

Once you have downloaded the above packages, the next step is to create the directory structure where everything is going to be installed. I installed everything on my c: drive in a folder named server, in which I created three more folders named - Apache2, MySQL and php giving the directory structure like that in the following image:-



I think that before proceeding a **warning** is needed; check everything you do very carefully! As I know from bitter experience it is all too easy to make a typo and not spot it, in fact it is easy to get into the situation where the more you look at a mistake, that would be obvious to others, the more you see what you expect to see.

Install Apache

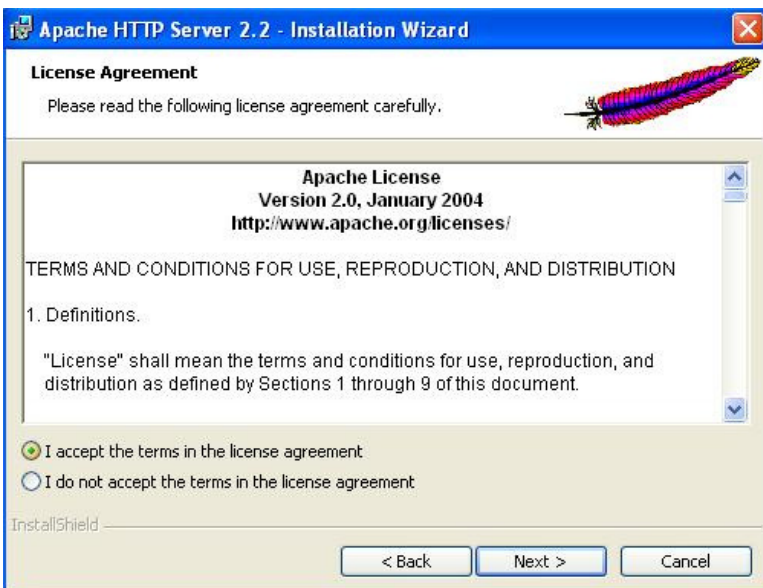
Ok, this is where the serious stuff begins; the first thing we have to install is Apache, because without the server nothing else will work. Well that is a bit of an exaggeration, but without the server you will not be able to use MySQL and PHP together to develop web pages.

So, let's install Apache; open up the folder, or navigate to where you saved the installer and double click on it, which should bring up a screen similar to this.



Click Next >

This screen basically tells you that you are about to install Apache. Clicking Next > brings up the following screen.



Select “I accept the terms in the license agreement” then click Next >

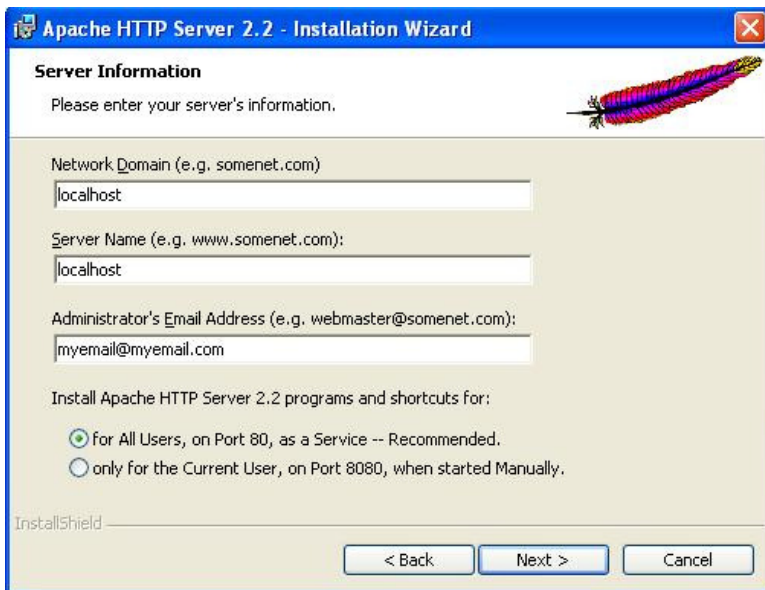
The “Next >” button is only available when you accept the terms, so you have to accept them to be able to install the application. You can read them if you really want to, but it is all pretty boring stuff; 99% of people don't bother.



Click Next >

This screen provides some basic and useful information about the application and where we can locate updates etc., again you don't need to read this unless you really want to.

The next screen is where we have to begin having a little input into the whole process:-



Here we provide the server information, the basic configurations for the server which takes moments. The settings for this screen are as follows, the bold words being what you need to enter:-

Network Domain (e.g. somenet.com): **localhost**

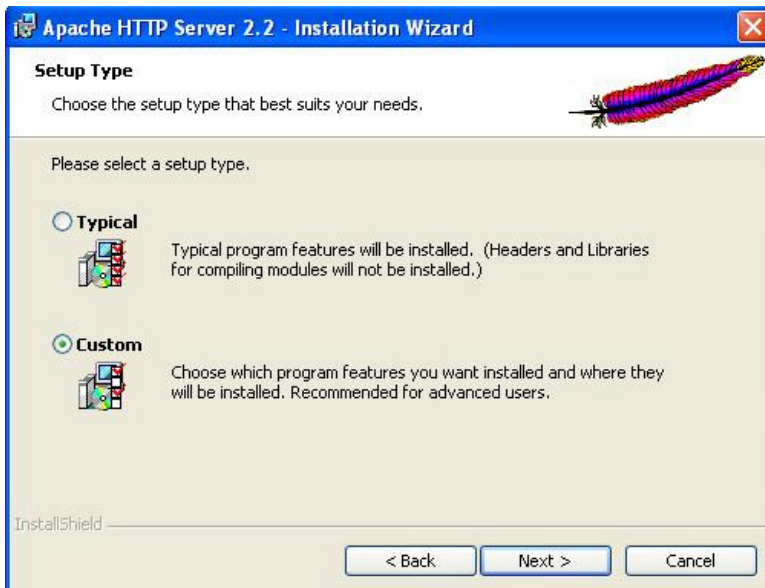
Server Name (e.g. www.somenet.com): **localhost**

Administrator's Email Address: **Your email here**

This is only really necessary if you intend to use the server for email purposes and can easily be left at the default setting

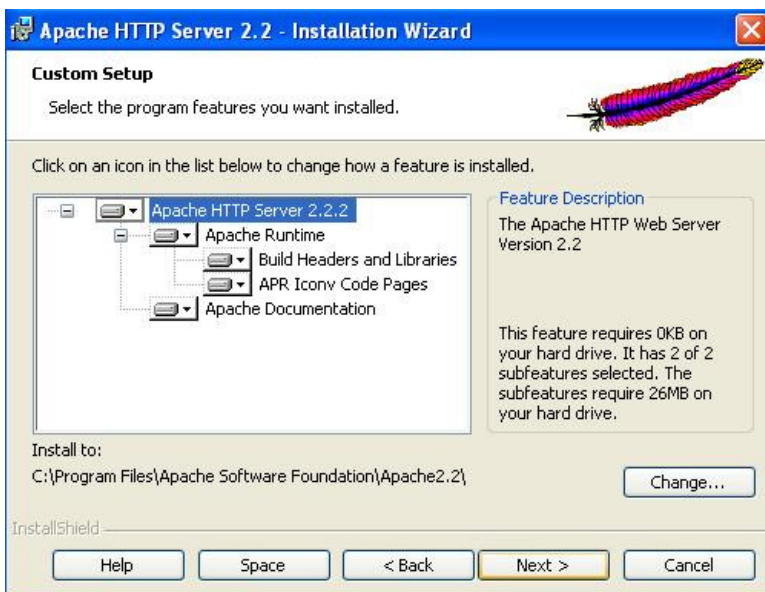
Next select **“Run as a service for all users – Recommended”**. This will start Apache automatically each time you boot up your computer, and is the best solution for users that want something quick, simple and with the minimum maintenance needed.

Click Next >



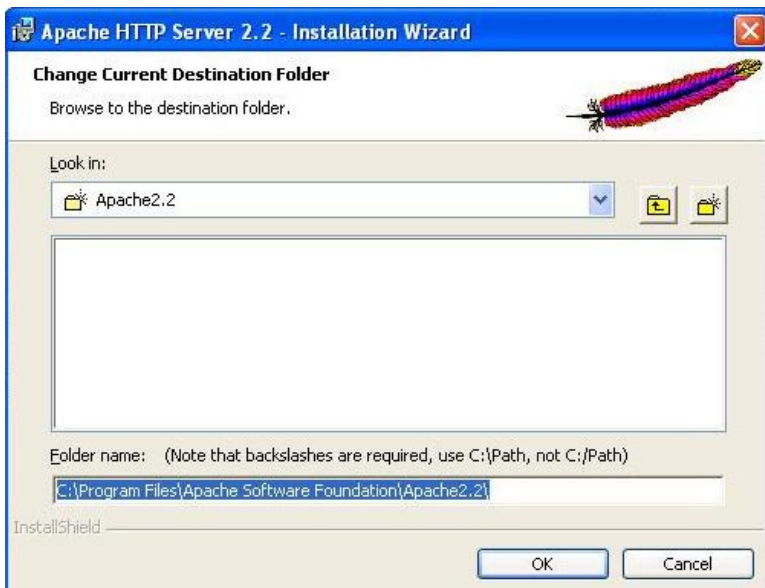
From here we are going to begin to make changes to the installation process by changing the default location to which Apache installs. We are doing this because we want to organize all of our server applications in a single folder for ease of use.

Select Custom
then
Click Next >



Click the “Change...” button located in the bottom right

This is where we decide to change the default installation location from C:\Program Files\Apache Software Foundation\Apache 2.2\ to the location that we previously decided, which in this case is C:\server\Apache2.2\
Clicking change will bring up the following screen where this change is actually made:-

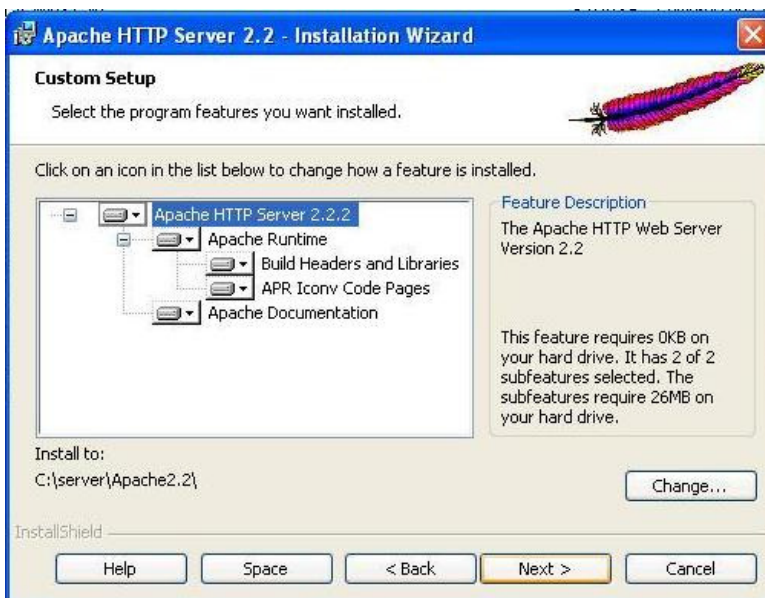


Here, if you are confident enough (I never am), you can type the location directly into the box provided; personally I prefer to navigate to the folder into which I want to install the application using the drop down menu “**Look in:**” it is much easier and you don’t have to remember the back slashes as it puts them in automatically.

Note: If you do type in the path yourself remember to use the back slash “\” and not the forward slash “/”, and always remember that the path ends with a back slash, i.e. C:\folder1\folder2\

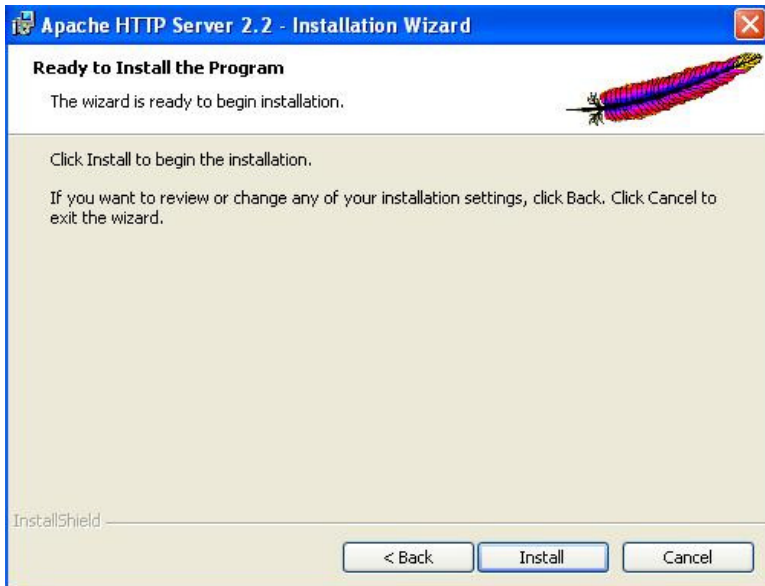
Click Ok

This will bring us back to the previous screen which should now reflect the changes that we have just made for the installation location.



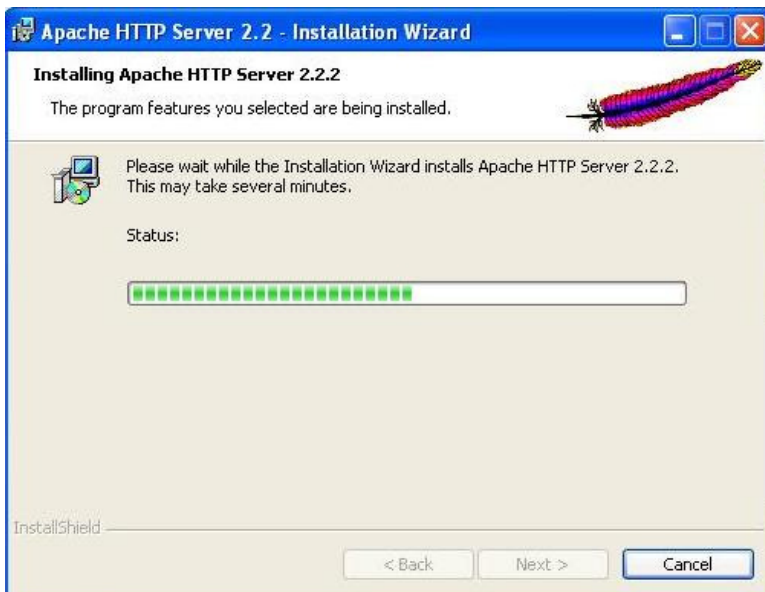
Click Next >

That is the hard work of installation done it is all down hill from here on in.



You are now ready, having (I hope) carefully made the necessary changes, to install Apache.

Click Install



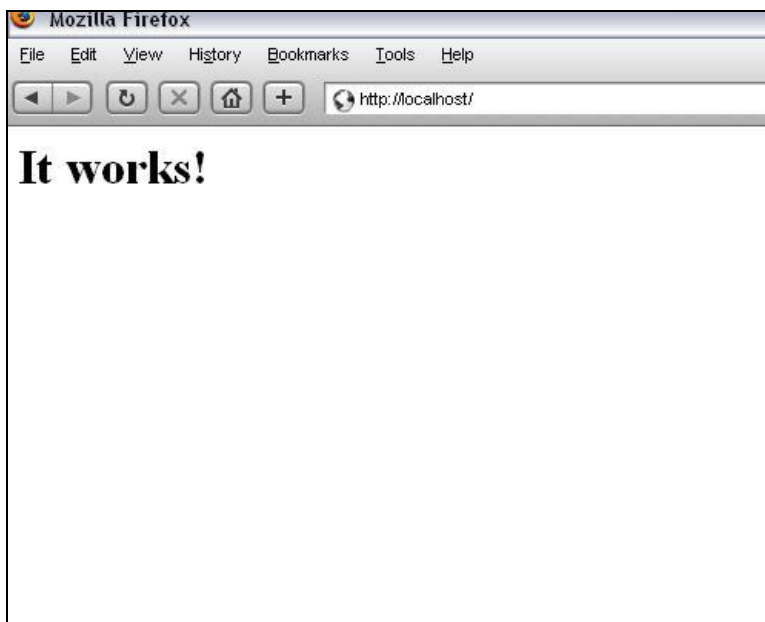
Now we just wait a bit while the installer does its business, go and make a cup of coffee, or kick the cat (only joking), or whatever your preference might be whilst waiting.



Once the installer has done its work you will get a window similar to this.

Simply click Finish

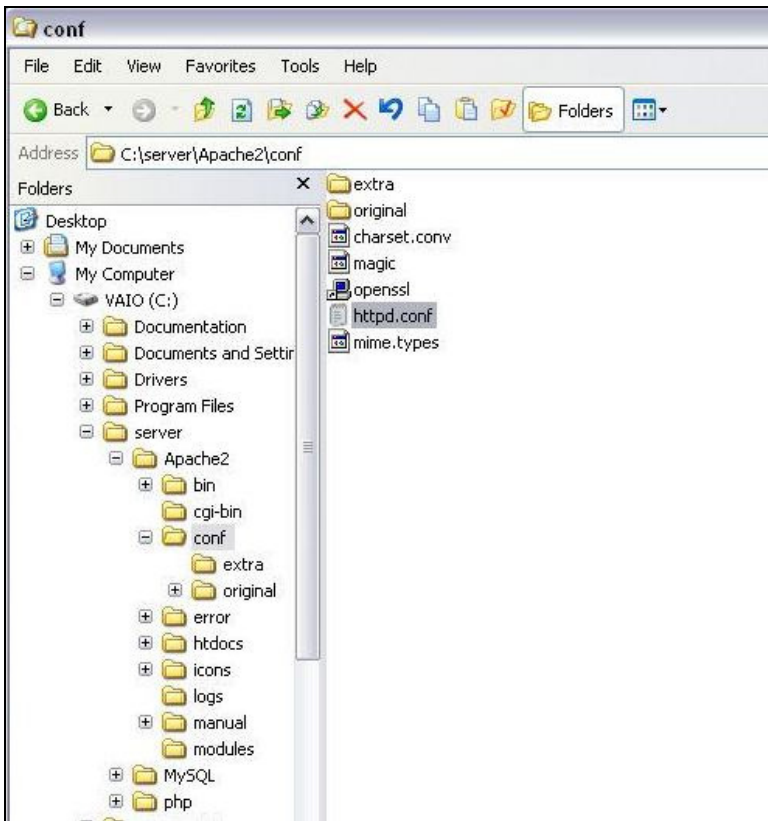
Now we need to check if Apache is installed and running correctly; fire up your favourite browser (as you will see mine is Firefox, which is by far the best browser) and type **http://localhost/** in as the URL and hit enter. You should see something like the following:-



If you see this, then everything worked and Apache is installed and working properly.

When we started we setup Apache to run as a service to make life run more smoothly. However there will be times when you will need to stop and restart the server manually; for example, when we changes to Apache's configuration file (httpd.conf) the restart is needed for any changes we make to take effect. This file contains all of the configuration settings needed for the server operate effectively.

Any web pages we create will be saved to the web root (sometimes document root) directory (in this case "C:\server\Apache2.2\htdocs"), this is where the browser will look for anything with a "localhost" url i.e. http://localhost/info.php. The location of "localhost" can be changed by making the relevant changes to the httpd.conf file.



To change the document root location navigate to the httpd.conf file as shown above, double click and it will open in Notepad, then find these lines:

```
#  
# This should be changed to whatever you set DocumentRoot to.  
#  
<Directory "C:/server/Apache2.2/htdocs">
```

and change the line `<Directory "C:/server/Apache2.2/htdocs">` to read whatever you want it to be; for example `<Directory "C:/server/Apache2.2/mynewwebrootfolder">`. Personally I see no reason to change this default setting at all unless you need to for a specific purpose.

All browsers when give a web address like `http://localhost/` will search for a file named `index.html`, that is unless the web address is pointed to a specific file like `http://localhost/info.php` when it will look specifically for the file `info.php`

If you want to use a file other than `index.html` as your default file when a directory is requested, find these lines and change "DirectoryIndex" or add to it the file you want the browser to find:

```
# DirectoryIndex: sets the file that Apache will serve if a directory  
# is requested.  
#  
<IfModule dir_module>  
DirectoryIndex index.html  
</IfModule>
```

If we change the above into:

```
# DirectoryIndex: sets the file that Apache will serve if a directory  
# is requested.  
#  
<IfModule dir_module>  
DirectoryIndex index.htm index.html  
</IfModule>
```


This change would mean that if there were to files in the directory (index.htm and index.html) then Apache would return index.htm to the browser rather than index.html. Again, personally, I can see no reason for making such modifications, that is unless there is a specific reason to do so.

Remember if you make any changes to the httpd.conf file Apache has to be restarted before any of the changes will take effect.



Restarting Apache is quite simple, there is a little feather icon (indicated by the arrow in the picture above) in the system tray down in the right hand corner near the clock, double click it to bring up the Apache Service Monitor:-

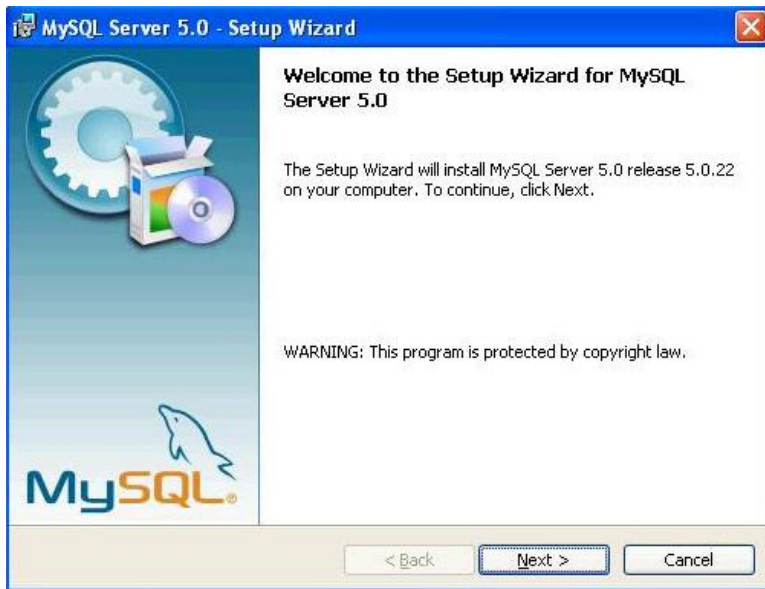


Now there are two options available, you can either click on **“Stop”** at which point **“Start”** will appear and you click on that; or there is the easier option of simply clicking **“Restart”** which will simply stop Apache and immediately start it again.

One down Three to go!

Install MySQL

So, we have installed Apache, now it is the turn of MySQL. Find and open the MySQL .zip file that you downloaded and extract the setup.exe file it contains, I suggest to the desktop where it is more easily available. Next double click on the setup.exe file to begin the process, the first screen will look something like this depending on the version you have downloaded.



Click Next >

Just as with Apache we are going to install MySQL in a location other than the default, and this next screen gives us the opportunity to customise our installation.



As we want to customise the installation, on this screen we select the **Custom** radio button.

Then

Click Next >



The resulting screen is very similar to the one we got with Apache to change the installation location. Once again we are deciding to install the application to a location other than the default location, so we

Click Change...

bringing up the following screen.



The default location in the box below “**Folder name:**” will be “C:\Program Files\MySQL\MySQL Server 5.0\”, and we want to change this to “C:\server\mysql” in keeping with our ethos of keeping all of the applications in one place. As before, you can either type the path “C:\server\mysql” directly into the “**Folder name:**” box (ensuring that it is correct with no typo’s), or navigate to the correct folder from the drop down menu box below “**Look in:**”

Then

Click Ok.

Which has the net result of returning us to the previous screen, which should now have the new installation location showing instead of the default location below “**Install to:**”.



Just make a final check that the location below “**Install to:**” reads “C:\server\mysql” (or whatever you have chosen).

Click Next >



This screen gives us a final opportunity to ensure that we have got everything as we want it, if not click “<Back” and rectify any errors; although I am quite sure this wont be necessary.

Click Install



Again we have to wait a bit while the installer does its business, go and have a beer, or beat the kids (only joking), or whatever your preference might be whilst waiting. Eventually you will get the following screen:-



Unless you are actually desperate to “Create a new free MySQL.com account”, I was not, select the “**Skip Sign-up**” radio button.

Click Next >

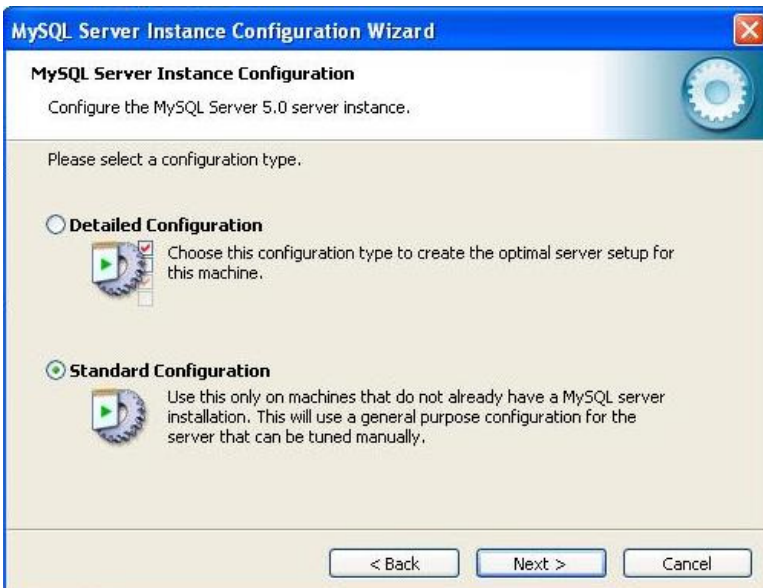


Depending on the version being installed you may get a couple of other screens before this one, they are of little consequence and Next. Can be clicked until you come to this screen, which is the start of the configuration process.

Click Finish



Click Next >



As this is a new installation we want the “**Standard Configuration**” so select that radio button. The Detailed Configuration is for those upgrading their MySQL which I will not be covering here.

Click Next >



Now we begin the basic configuration of MySQL, there are only two things we need to ensure here, and they are that both the “**Install As Windows Service**” and the “**Launch the MySQL Server automatically**” boxes are both checked. The second one is to ensure that the MySQL server is started when the computer is booted, or if, for any reason, the server is shut down.

Click Next >



Two things to attend to here, check the “**Modify Security Settings**” box if it not already checked, then choose and enter your password for the root access. You will see an “**Enable root access from remote machines**” check box, the only circumstances in which you would need to check this box would be if you actually need this facility. It is highly unlikely that it would be needed for learning\development purposes unless this was actually being done remotely with MySQL etc. on a host machine, and as we are installing on our own computer it is not really needed.

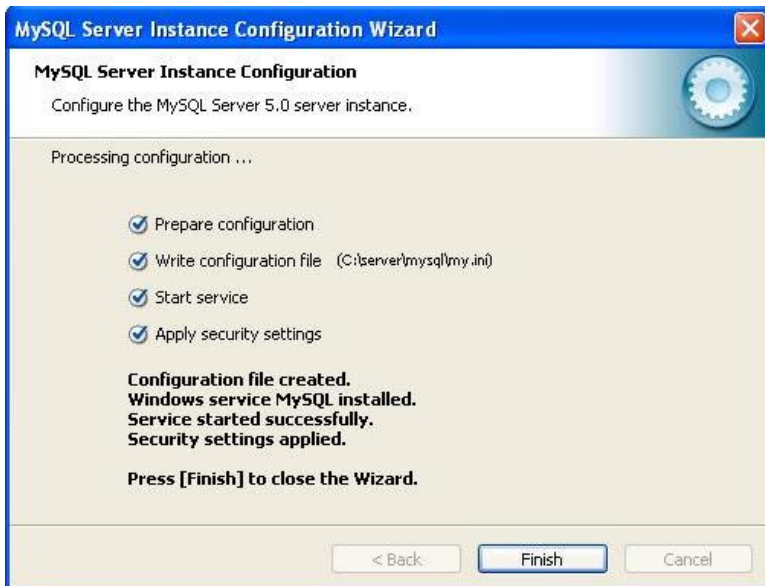
Click Next >



This screen is simply to implement the configuration settings that we have just entered, letting us know exactly what is happening.

Click Execute

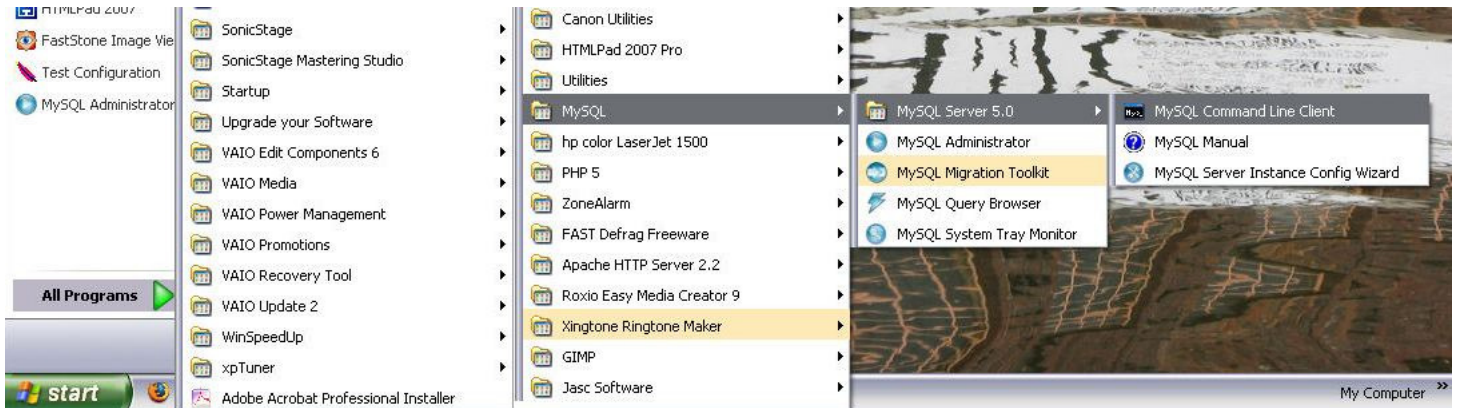
Apparently there could be some problems at this stage, although, having installed MySQL several times, I have never encountered any myself, that is other than telling Windows Firewall to “unblock” the application. But then, who would be using Windows Firewall when there are far better products out there, and for free; products like Zone Alarm which I find excellent. Solutions to any problems can either be found in the online documentation for MySQL, or through one of the many forums out there, I find PHP Freaks an excellent source.



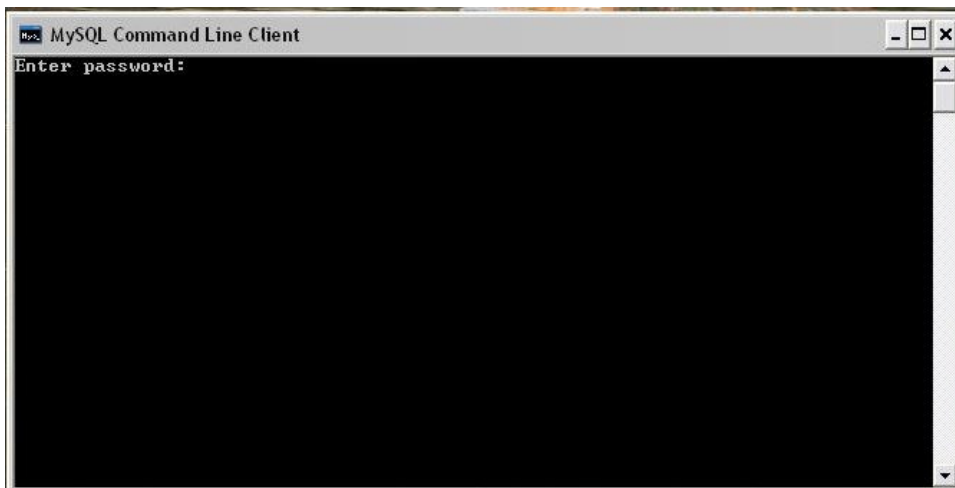
If everything has happened as I would expect it to you will be looking at the above screen telling you what has been done, and that the configuration process was successful.

Click Finish

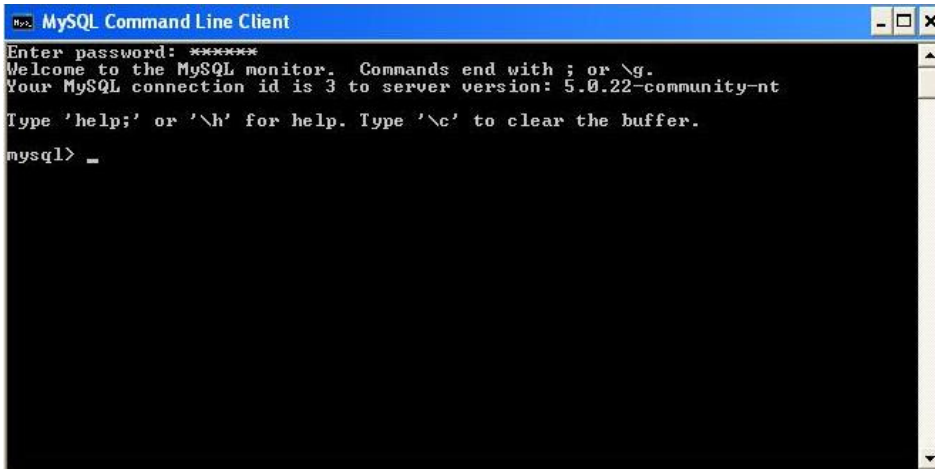
Now we need to know that MySQL is working properly and to do that we open up the Command Line Client for MySQL.



As can be seen above the sequence for this is Start > All Programs > MySQL > MySQL Server 5.0 > MySQL Command Line Client.



The above screen should appear, just type your chosen password for root access and hit Enter. If all is well you should get a screen similar to the following one:-



```
MySQL Command Line Client
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 3 to server version: 5.0.22-community-nt
Type 'help;' or '\h' for help. Type '\c' to clear the buffer.
mysql> _
```

If you are seeing this screen then all is working properly, if not you have done something wrong, although if this were the case I would have expected you not to be here anyway.

MySQL is now installed and working correctly type “quit” or “exit” at the prompt and hit Enter to close the Command Line Client.

Install MySQL GUI Tools

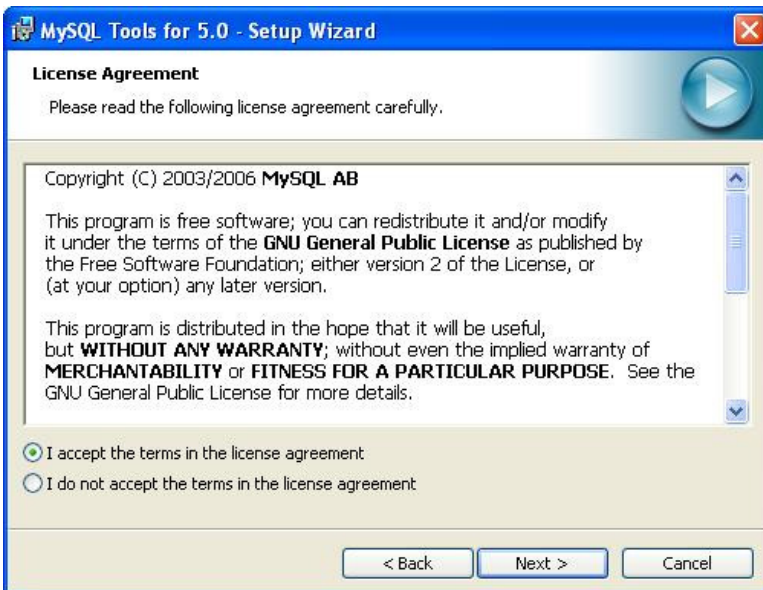
#####

Sometimes, managing all MySQL users, maintain MySQL databases, make Query application and migrating databases would be very hard and takes to much time if we only use MySQL command line to do that. Fortunately for us, MySQL has develop a Windows Interface to do that. They create MySQL Tools. It's Free and simple to use.

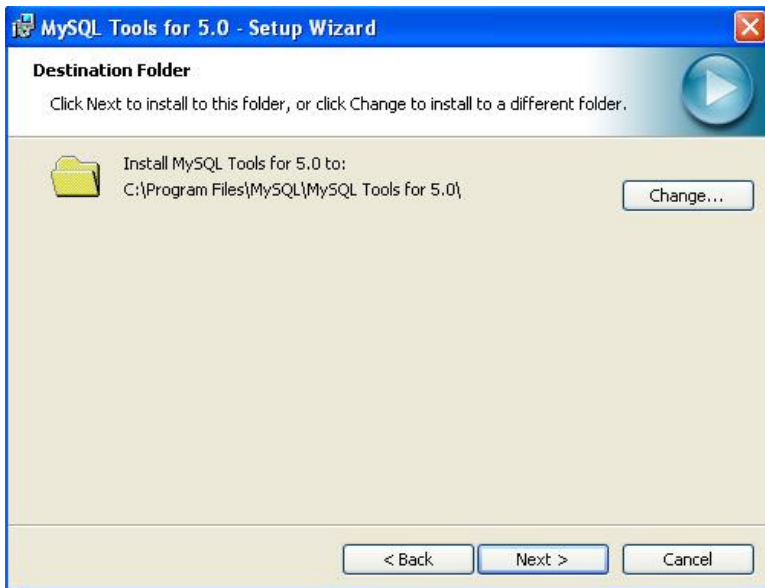
So let's start off by executing the installer that we downloaded in the beginning.



This will bring us to a window similar to this one, simple click on Next.



Next we must agree to MySQL's License Agreement, simply click on "I Accept the terms in the license agreement".



Next the application would like to know where to install MySQL Tools. Because we need to gather all the application into c:\server directory, simply click on Change.



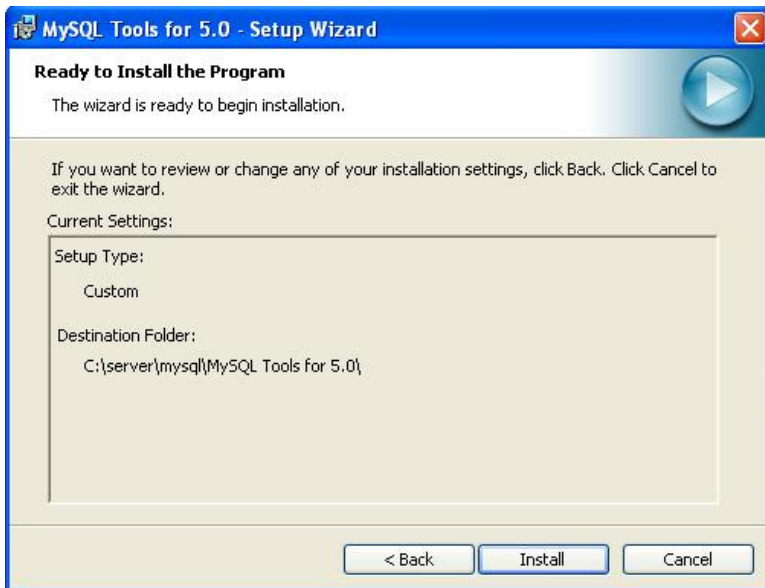
Find the new location, or type it in the Folder Name. In this case we're using "C:\server\mysql\MySQL Tools for 5.0\" as we'd like to keep everything in its own folder. Click OK



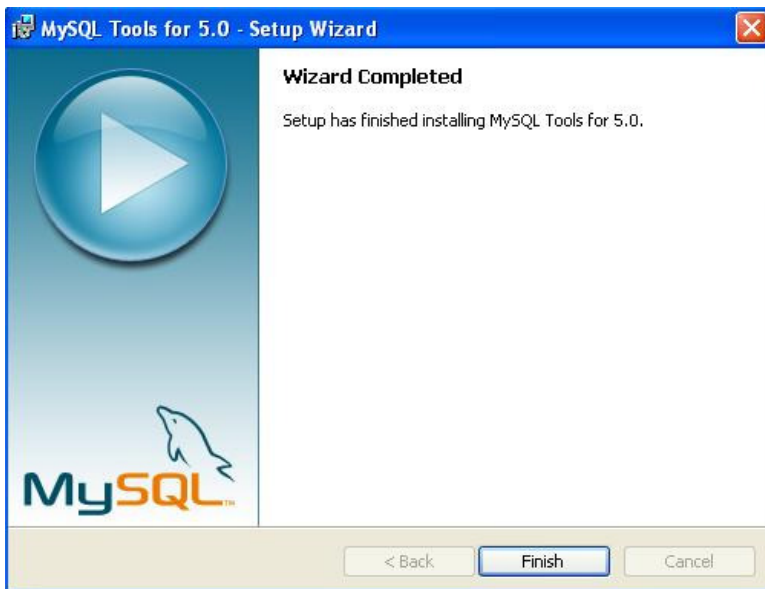
Here's the new location where we want to install MySQL Tools. Simply click Next.



Next the installer would like to know whether we need to install all components or we customize certain components only (MySQL Administrator, MySQL Query Browser, MySQL Migration Toolkit, MySQL Workbench and Language Support). Just choose Complete and click Next



After that, the application will bring us to ready to install window. Simple click on Install and the installer will begin the installation.



Once everything is finished, we'll get a window similar to this. Simply click on Finish and we're done. MySQL Administrator, MySQL Query Browser, MySQL Migration Toolkit and MySQL Workbench is now installed in the location we provided.

You can see all the MySQL Tools application under MySQL folder, click at Start - Programs - MySQL. Just like this:



Install PHP

In the last few articles we've gotten our basic [server](#) applications installed and running. This will give us a great foundation to continue with the next few applications. The next thing we'll need to get installed is PHP, as this will be the language we're going to use to develop with, and we can't use PHP without having a PHP engine phrasing all our code.

We didn't use the PHP installer because it isn't the preferred method for installing PHP and it's not secure. So we prefer manually install the PHP.

1. First, we want to create PHP folder under our server directory, so we must create c:\server\PHP folder
2. Open the PHP Zip file we downloaded at the beginning of this project and extract all the files at c:\server\PHP. If you use Winzip, make sure you check "Use Folder Names" option
3. After you've extracted the files, you'll see these directory structure at PHP folder:

```
c:\server\PHP
```

```
|
```

```
+--dev
```

```
| |
```

```
| |-php5ts.lib
```

```
|
```

```
+--ext -- extension DLLs for PHP
```

```
| |
```

```
| |-php_bz2.dll
```

```
| |
```

```
| |-php_cpdf.dll
```

```
| |
```

```
| |..
```

```
|
```

```
+--extras
```

```
| |
```

```
| | +--mibs -- support files for SNMP
```

```
| |
```

```
| | +--openssl -- support files for Openssl
```

```
| |
```

```
| | +--pdf-related -- support files for PDF
```

```
| |
```

```
| |-mime.magic
```

```
|
```

```
+--pear -- initial copy of PEAR
```

```
|
```

```
|
```

|*-go-pear.bat -- PEAR setup script*

|

|*-fdftk.dll*

|

|*-..*

|

|*-php-cgi.exe -- CGI executable*

|

|*-php-win.exe -- executes scripts without an opened command prompt*

|

|*-php.exe -- CLI executable - ONLY for command line scripting*

|

|*-..*

|

|*-php.ini-dist -- default php.ini settings*

|

|*-php.ini-recommended -- recommended php.ini settings*

|

|*-php5activescript.dll*

|

|*-php5apache.dll*

|

|*-php5apache2.dll*

|

|*-..*

|

|*-php5ts.dll -- core PHP DLL*

|

|*-...*

4. After that, we need to create 2 new folder, and we can name it uploadtemp (it's for temporary directory for HTTP uploaded files) and sessiondata (it's for storing all the session files). So, it will be c:\server\PHP\uploadtemp and c:\server\PHP\sessiondata

5. The next step is to set up a valid configuration file for PHP, php.ini. There are two ini files distributed in the zip file, php.ini-dist and php.ini-recommended. We advise you to use php.ini-recommended, because it has been optimized the default settings in this file for performance and security. So copy php.ini-recommended and put it in the same folder (c:\server\PHP) with the name php.ini

6. We still need some customization at PHP configuration, so open php.ini with your favorite text editor, such as notepad. Find these strings and change the values according to these:

- *variables_order = EGPCS* => The environment variables are hashed into the `$_ENV`
- *register_long_arrays = On* => Enables registration of the older (and deprecated) long predefined array variables (`$HTTP_*_VARS`)
- *register_argc_argv = On* => Enables registration of the `$argv` and `$argc` variables
- *magic_quotes_gpc = On*
- *upload_tmp_dir = c:\server\PHP\uploadtemp* => after you've change the values, make sure you've delete the comment character (;) at the beginning of the syntax, to make it active
- *session.save_path= "c:\server\PHP\sessiondata"* => after you've change the values, make sure you've delete the comment character (;) at the beginning of the syntax, to make it active. If there's more than one of these syntax, just use one and uncommented it
- *extension_dir = "c:\server\PHP\ext"*
- *short_open_tag = On* => Allow the `<? tag`
- *max_execution_time = 120* => Maximum execution time of each script, in seconds
- *display_errors = Off* => You would like to turn this on in development server, but would be better to turn this off on production server. If you set `log_errors = on` and you didn't specified the file for `error_log` file, you can see the error at Apache error log file

You can change all of the above configuration if you need to do so, but you must understand what you're about to do

7. We must tell PHP to locate the php.ini at c:\server\PHP by doing these:

- Go to Control Panel and open the System icon (Start -> Control Panel -> System)
- Go to the Advanced tab
- Click on the 'Environment Variables' button
- Look into the 'System variables' panel
- Click on 'New' and enter 'PHPRC' as the variable name and the directory where php.ini is located as the variable value (c:\server\PHP)
- Press OK and restart your computer

8. After we've finished configuring the PHP, we must configure the Apache webserver to work with PHP files. There is 2 way, by installing as a CGI binary or installing as an Apache module. Installing as a CGI binary will make our server is open to several possible attacks. So we'd prefer installing as an Apache module

9. Open Apache configuration file, just like this:



10. Add these lines at the very bottom:

```
LoadModule php5_module "c:/server/PHP/php5apache2.dll"  
AddType application/x-httpd-php .php .htm .html  
PHPIniDir "c:/server/PHP"
```

if you have PHP version 5.2.x or later, you can use this line instead the above syntax (Thanks to **Monique Pengel**):

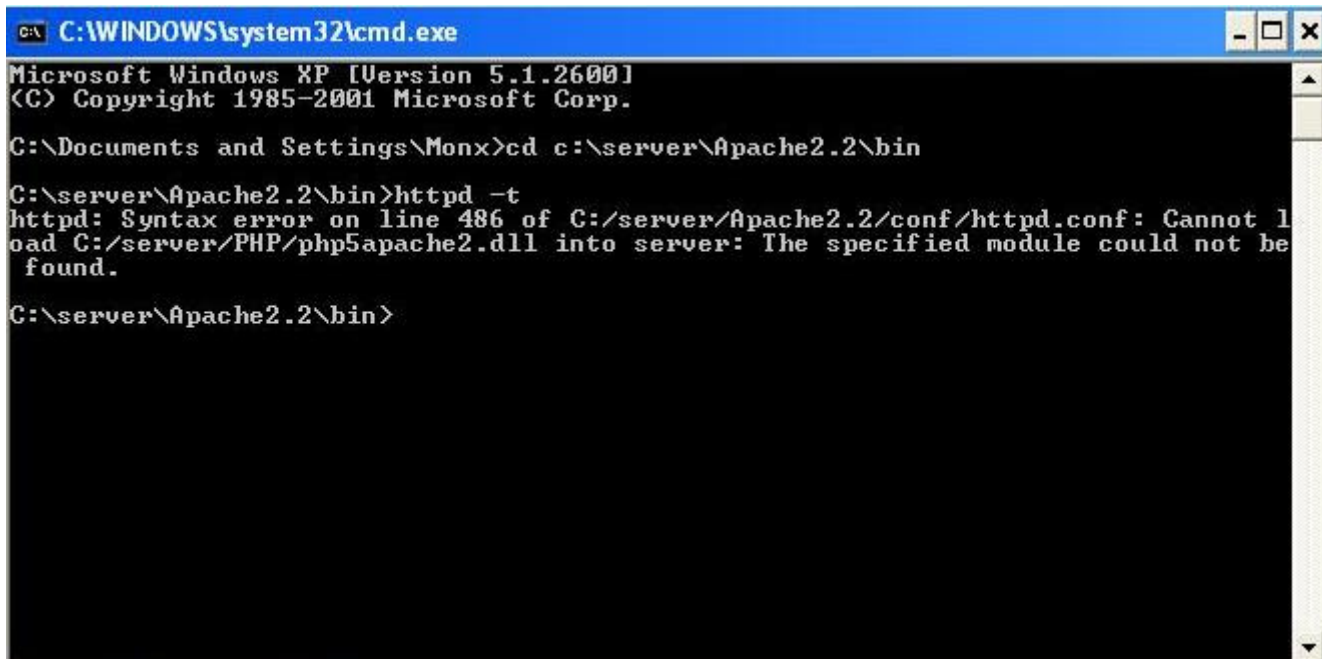
```
LoadModule php5_module "c:/server/PHP/php5apache2_2.dll"
```

```
AddType application/x-httpd-php .php .htm .html
```

```
PHPIniDir "c:/server/PHP"
```

The first line will let Apache know the installation of PHP and the next two will let Apache know how to handle PHP files (sometimes, in html file, we can put PHP scripts on it, so we must also configure PHP to handle .htm and .html file).

11. Restart the Apache server. If the Apache server can't started, then do:



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Monx>cd c:\server\Apache2.2\bin

C:\server\Apache2.2\bin>httpd -t
httpd: Syntax error on line 486 of C:/server/Apache2.2/conf/httpd.conf: Cannot load C:/server/PHP/php5apache2.dll into server: The specified module could not be found.

C:\server\Apache2.2\bin>
```

- open command prompt (Start -> Run -> Type "cmd", click OK)
- Type "cd c:\server\Apache2.2\bin" and then press Enter button
- Type "httpd -t" to see if there's an error on our Apache configuration
- If the error said "*httpd: Syntax error on line 486 of C:/server/Apache2.2/conf/httpd.conf: Cannot load C:/server/PHP/php5apache2.dll into server: The specified module could not be found*", you can continue on the next steps (the line number can be different, though). Otherwise, you must check your Apache configuration again by examine the error logs

12. By looking at the error messages, it seems that there's something missing at the php5apache2.dll module. After searching on the net, i have found a solution (Thanks to Steffen, <http://www.apachelounge.com> webmaster). Just download these file:

- <http://www.apachelounge.com/download/mods/php5apache2.dll-php5.1.x.zip> or at my backup server at <http://other.monx007.com/tools/php5apache2.dll-php5.1.x.zip> (it's 05/05/2006 version). You must check the file at <http://www.apachelounge.com> first for the newest version
- if you haven't installed the .NET framework, then you must also download http://download.microsoft.com/download/d/3/4/d342efa6-3266-4157-a2ec-5174867be706/vcredist_x86.exe or you can find the link from <http://www.microsoft.com/downloads/details.aspx?familyid=32BC1BEE-A3F9-4C13-9C99-220B62A191EE&displaylang=en>

13. After that, just read the instruction file (Readme.txt), or, i can describe it here anyway:

- Copy php5apache2.dll to c:\server\PHP

- Copy httpd.exe.manifest to c:\server\Apache2.2\bin

- If you haven't installed the .NET framework, then run vcredist_x86.exe that we've downloaded before (Click Yes to Accept the agreement).

14. Try to start the apache again. If it can't started, check the Apache configuration file just like steps 11.

15. If the Apache start successfully, we must test to see if PHP is installed correctly

16. We'll want to create a file in the htdocs folder located in the main folder of Apache. This is where all of our web documents are going to be stored. Create a new file in this folder can name it phpinfo.php. Be sure that it saves with a .php extension and not .php.txt or whatever you'll use to create the file. Once the phpinfo.php file has been created, open it up and place the following code within it.

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

Save it, close the file and then open your favorite browser and type this in the location bar.

<http://localhost/phpinfo.php>

17. If you see the following page (like below image), then PHP is installed and setup correctly on your server.



System	Windows NT M 5.1 build 2600
Build Date	May 4 2006 10:30:29
Configure Command	cscript /nologo configure.js "--enable-snapshot-build" "--with-gd=shared"
Server API	Apache 2.2 Handler (www.apachelounge.com)
Virtual Directory Support	enabled
Configuration File (php.ini) Path	C:\server\PHP\php.ini
PHP API	20041225
PHP Extension	20050922
Zend Extension	220051025
Debug Build	no
Thread Safety	enabled
Zend Memory Manager	enabled
IPv6 Support	enabled
Registered PHP Streams	php, file, http, ftp, compress.zlib
Registered Stream Socket Transports	tcp, udp
Registered Stream Filters	convert.iconv.*, string.rot13, string.toupper, string.tolower, string.strip_tags,

phpMyAdmin

In this article we're going to learn how to install phpMyAdmin with our new server tools so we can better manage our MySQL database. There are a few things in this article that may not seem correct or right and some users may have phpMyAdmin working very easily, it all depends on your setup.

First let's start by extracting the files to the proper location. Start by opening the phpMyAdmin zip file we downloaded in the beginning. Extract all of these files into the "htdocs" folder under the folder name "phpMyAdmin". This will help us to keep up with where it's installed, so now once we've unzipped everything we could load it up by using our favorite browser and point it to :

<http://localhost/phpMyAdmin/index.php>

phpMyAdmin - Error

Cannot load [mysql](#) extension. Please check your PHP configuration. - [Documentation](#)

The error message that displayed above means that we didn't include MySQL library to PHP configuration

To adding MySQL library to PHP configuration, just follow these steps:

1. Open PHP configuration file (php.ini) at c:\server\PHP using your favorite text editor (such as notepad)
2. Search for this syntax: ;extension=php_mysql.dll
3. If you found it, just delete the comment character (;)
4. Save the file and restart the Apache webserver. After that, try to run the phpMyAdmin again from your browser
5. If it's come with the same errors, then it means Apache couldn't find libmysql.dll file (this file is require for php_mysql.dll). Actually, this file is located at c:\server\PHP
6. So, we must change the Windows path to that location.
7. Go to Control Panel and open the System icon (Start -> Control Panel -> System)
8. Go to the Advanced tab
9. Click on the 'Environment Variables' button
10. Look into the 'System variables' panel, and highlight Path variable by click on it
11. Click Edit button, and at the Variable value, at the end of the syntax, add this code:
;c:\server\PHP
12. Click OK and restart your computer
13. Try again to run the phpMyAdmin from your browser

Error

✘ MySQL menyatakan: ?

#1045 - Access denied for user 'root'@'localhost' (using password: NO)

This error message means that our [login](#) or password to connect to mysql database is wrong. In order for phpMyAdmin could connect to mysql database, we must supply the Login and password at phpMyAdmin configuration file.

Almost all configurable data is placed in config.inc.php. This file only needs to contain the parameters you want to change from their corresponding default value in phpMyAdmin/libraries/config.default.php. If config.inc.php does not exist at the phpMyAdmin directory, just copy config.default.php from libraries directory to phpMyAdmin directory, and rename it to config.inc.php.

Locate this string: \$cfg['PmaAbsoluteUri'] = '';

Inside the '' place the location of where phpMyAdmin is installed on the server. In this case we installed it within the root directory under its own folder name. So we'll use "http://localhost/phpMyAdmin/" as our absolute URL location.

There are 3 authentication methods that we can use to connect to phpMyAdmin:

1. config method (the default method)

This method is the least secure, as it requires storing your MySQL username and password in the file on the server. Anyone who would find the directory to your phpMyadmin installation would then be able to access your databases. This is the default method, however.

If you use the 'config' authentication method, it is strongly recommended that you secure the directory in which phpMyAdmin is installed, such as adding password protect to phpMyAdmin directory. This extra step is not needed when using either the 'http' or 'cookie' authentication methods.

Locate this string: \$cfg['Servers'][\$i]['auth_type'] = 'config';

Locate this string: \$cfg['Servers'][\$i]['user'] = 'put your MySQL login here';

Locate this string: \$cfg['Servers'][\$i]['password'] = 'put your MySQL password here';

2. http method

These methods are more secure, as your username and password are not stored on the server in your configuration file. The 'http' option uses the Apache http authentication method.

Locate this string: \$cfg['Servers'][\$i]['auth_type'] = 'http';

Locate this string: \$cfg['Servers'][\$i]['user'] = '';

Locate this string: \$cfg['Servers'][\$i]['password'] = '';

3. cookie method (recommended method)

These methods are more secure, as your username and password are not stored on the server in your configuration file. The 'cookie' method uses cookies, and has the advantage that you can "Log Out" at the end of your session, deleting the cookie.

Locate this string: \$cfg['Servers'][\$i]['auth_type'] = 'cookie';


Locate this string: \$cfg['Servers'][\$i]['user'] = '';

Locate this string: \$cfg['Servers'][\$i]['password'] = '';

Locate this string: \$cfg['blowfish_secret'] = 'fill anything here';

For the blowish_secret values, you can fill any alphabets, in example you can use 716e3c0.88759346, but, keep in mind that the maximum length seems to be 46 characters. The passphrase is just used internally to encrypt the password - you will not be prompted for it later.

After we've finished configuring phpMyAdmin, try to run phpMyAdmin again by using our favorite browser and point it to <http://localhost/phpMyAdmin/index.php>

 The mbstring PHP extension was not found and you seem to be using a multibyte charset. Without the mbstring extension phpMyAdmin is unable to split strings correctly and it may result in unexpected results.

This error tells us that the mbstring library didn't load at PHP. Without the mbstring extension, phpMyAdmin is unable to split strings correctly and it may result in unexpected results.

In order to load the mbstring library, we must do:

1. Open PHP configuration file (php.ini) at `c:\server\PHP` using your favorite text editor (such as notepad)
2. Find this syntax:
`extension=php_mbstring.dll`
3. Delete the comment character (;)
4. Save the file and restart the Apache
5. Try to open up phpMyAdmin again



The screenshot displays the phpMyAdmin interface. On the left, there is a sidebar with the phpMyAdmin logo and navigation icons. The main content area is titled "MySQL - 5.0.21-community-nt" and "phpMyAdmin - 2.8.". The interface shows various configuration options for the MySQL server, including the protocol version (10), server address (localhost via TCP/IP), user (root@localhost), and the MySQL charset (UTF-8 Unicode (utf8)). There is also a section for creating a new database with a dropdown menu for the collation (utf8_unicode_ci) and a "Ciptakan" button. The interface is in Indonesian, with the language setting visible as "Bahasa".

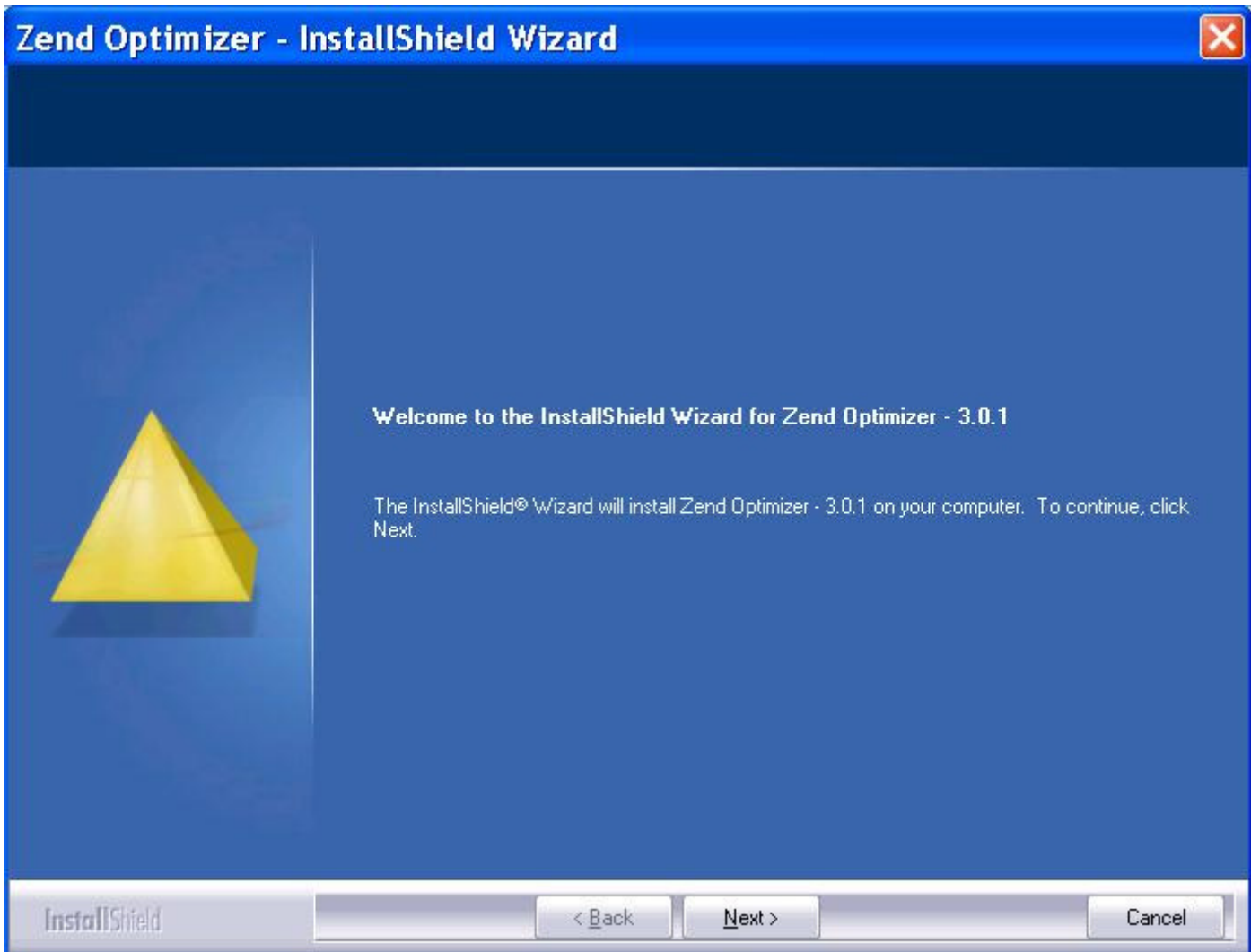
If you're able to login with no errors, you'll get the result like the image above, but it will be a bit different depending on your language setting. I've used Indonesian language setting.

6. You should deny access to the libraries subfolder in your webserver configuration. For Apache web server, you didn't have to do anything, because in the library folder there is already a .htaccess file that we can use. Such configuration prevents from possible path exposure and cross-site scripting vulnerabilities that might happen to be found in that code

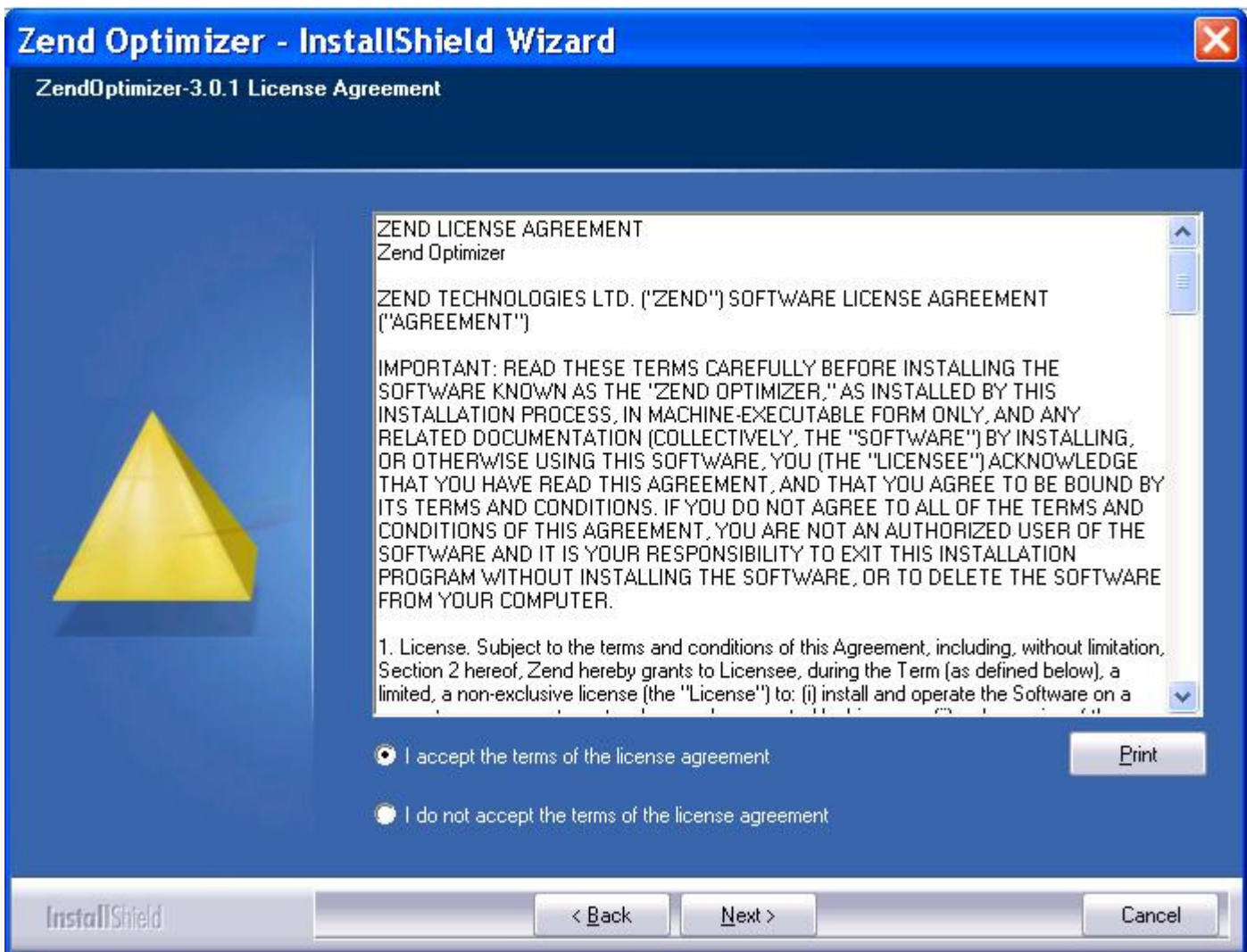
Zend Optimizer

We need an optimizer for all of our PHP code. So we're going to use Zend Optimizer as it's the industry standard and well it's free and they also [make](#) a 32 bit version, so it will work on Windows.

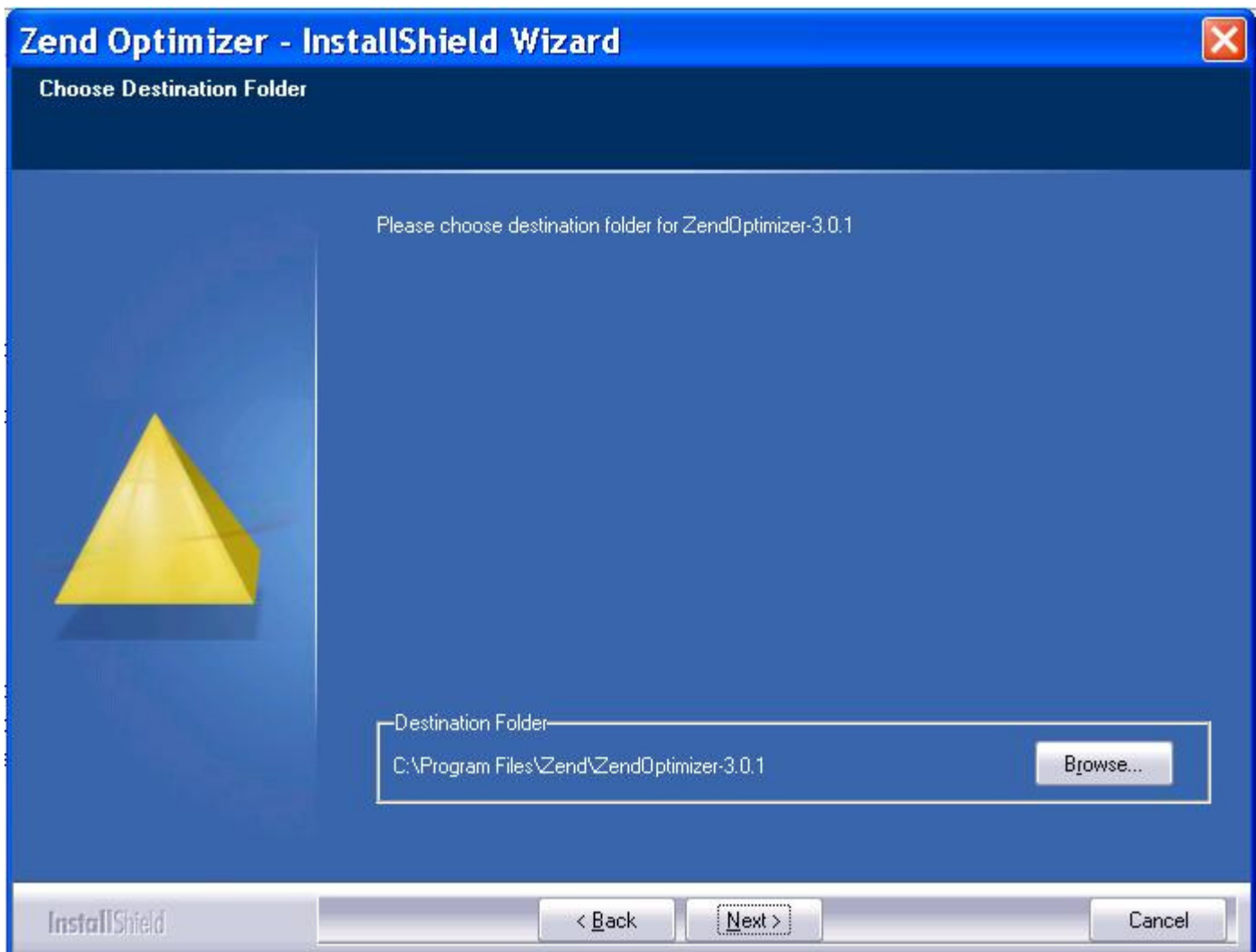
So let's [start](#) off by executing the installer that we downloaded in the beginning.



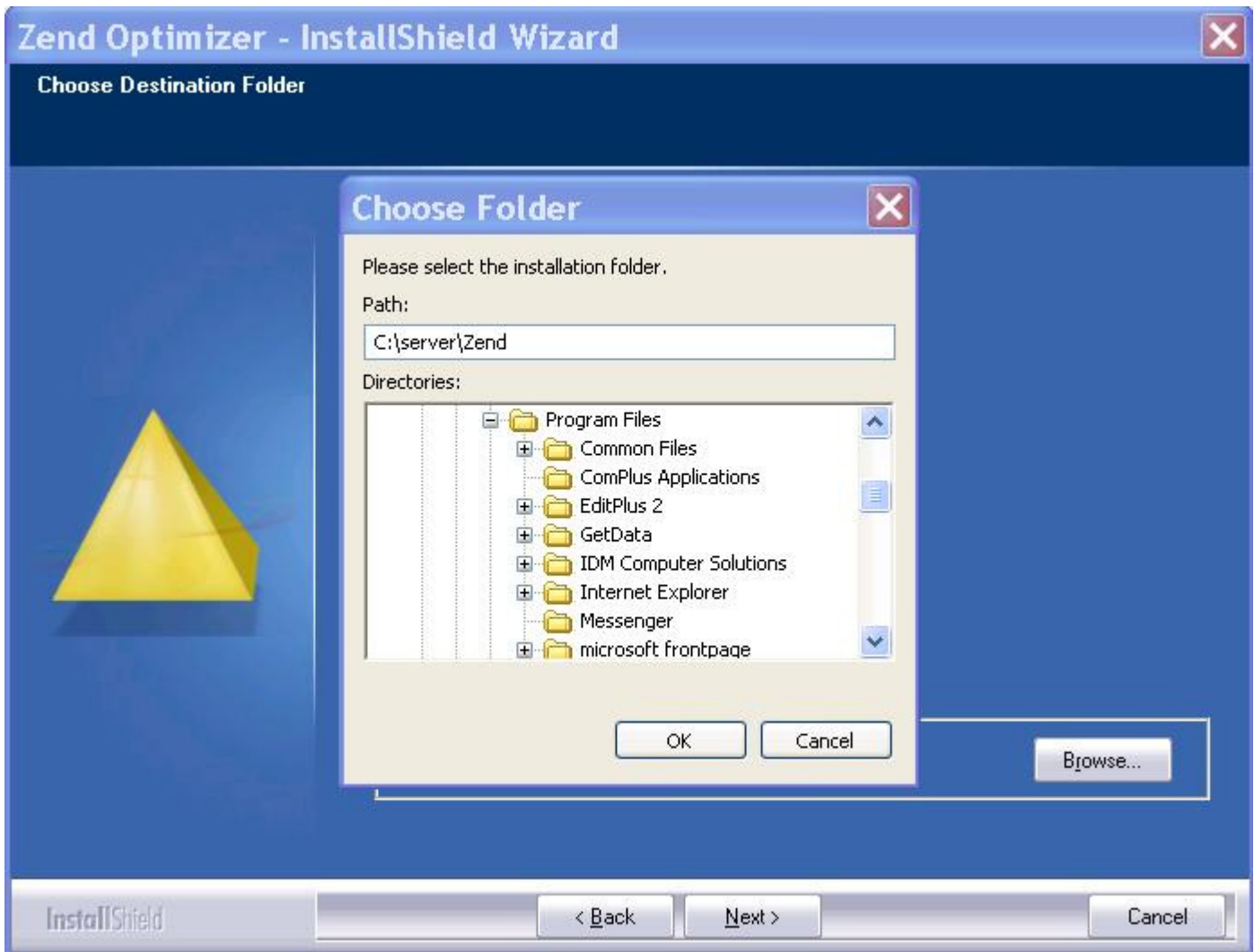
This will bring us to a window similar to this one, simple click on Next.



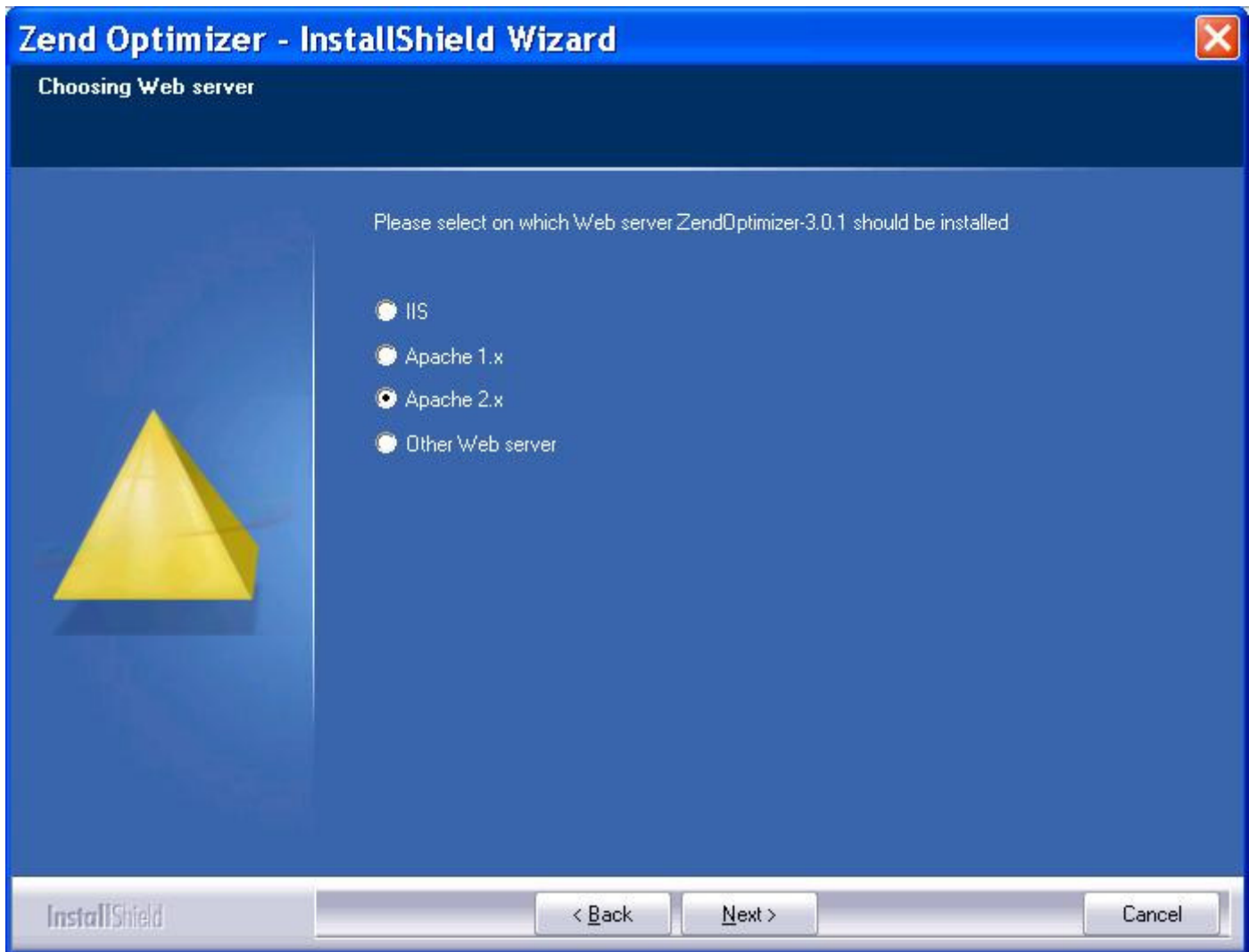
Next we must agree to Zend's License Agreement, simply click on Yes after the agreement has been read.



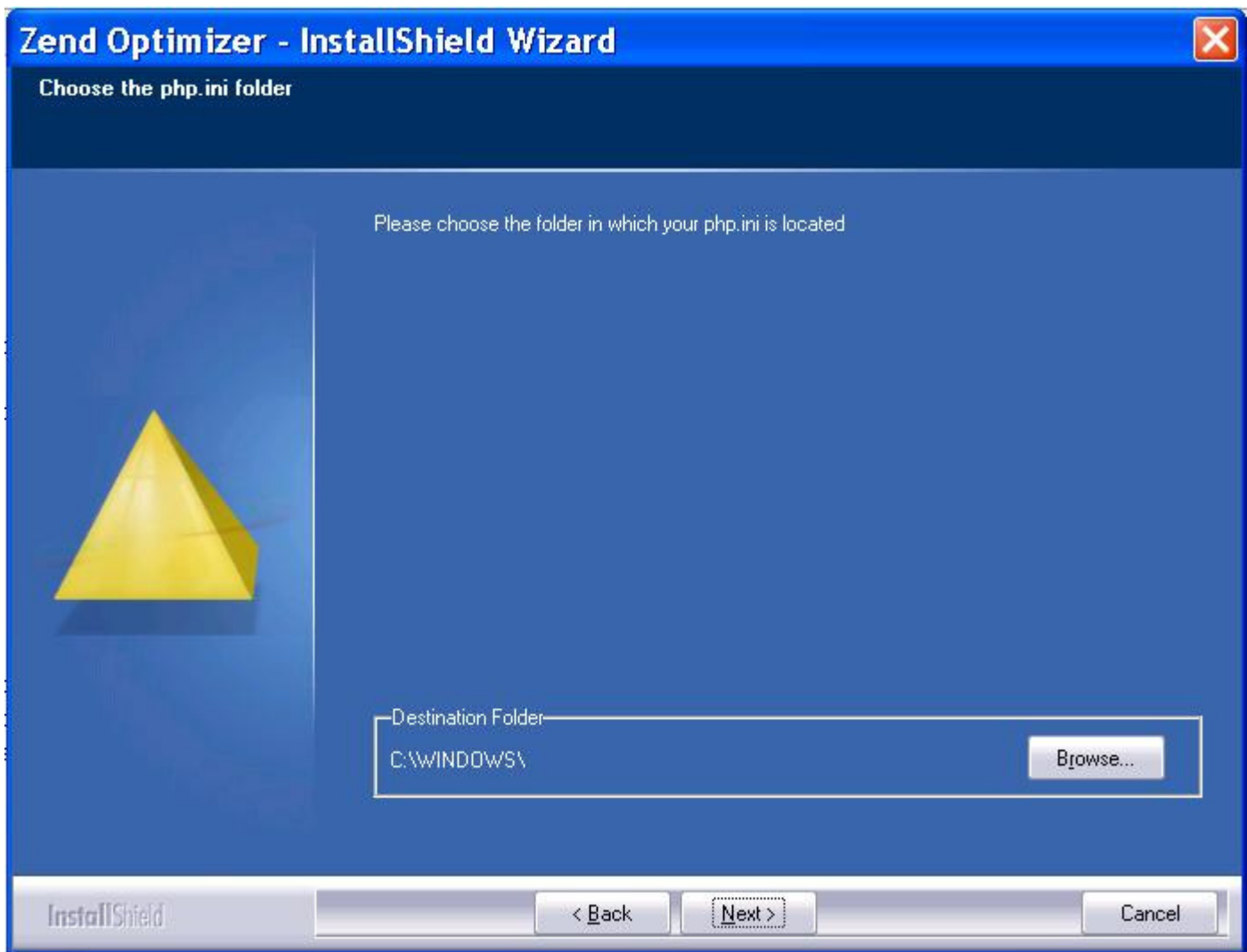
Next the application would like to know where to install Zend, simply click on browse.



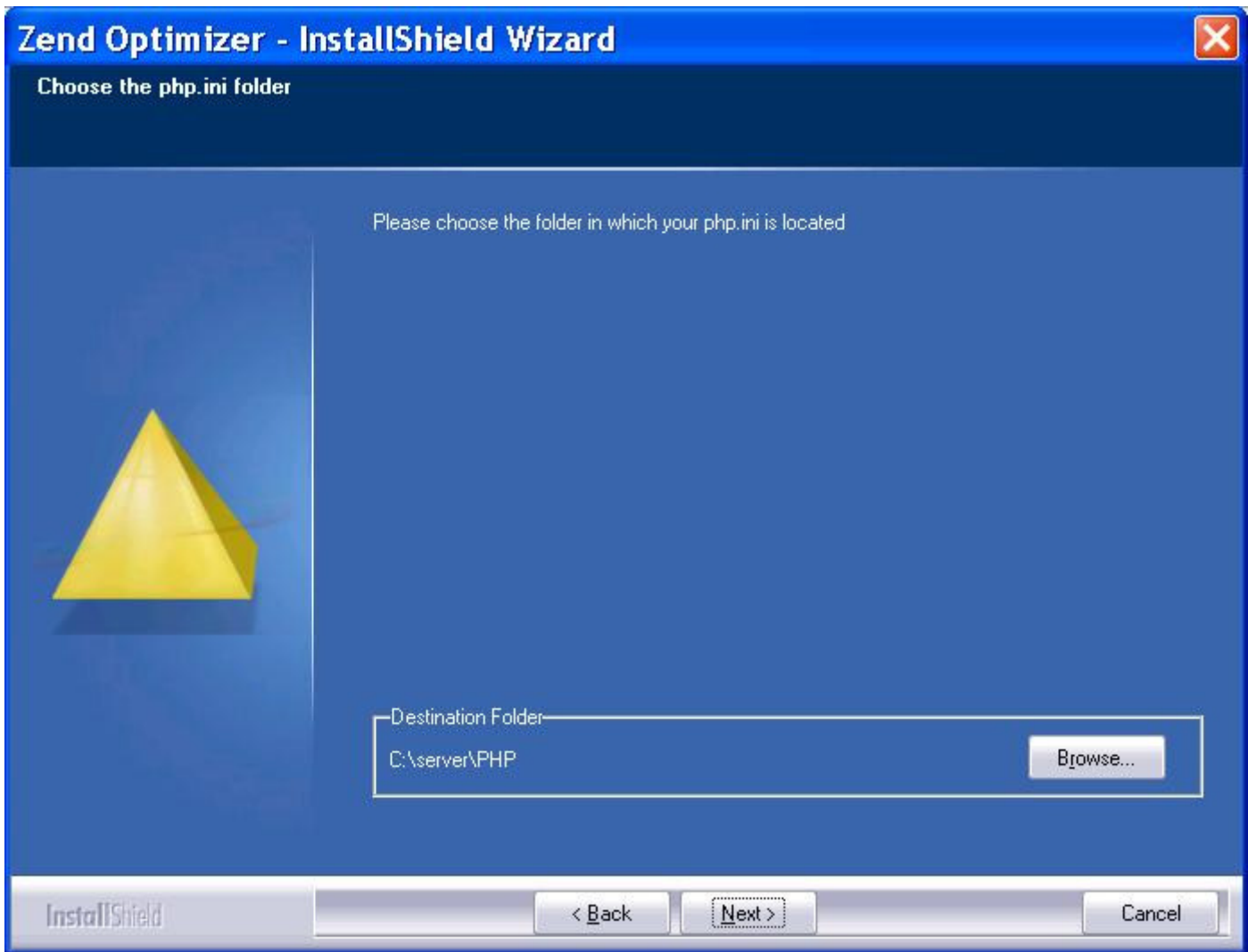
Find the new location, or type it in the URL bar. In this case we're using "C:\server\Zend" as we'd like to keep everything in its own folder. Click OK then Next



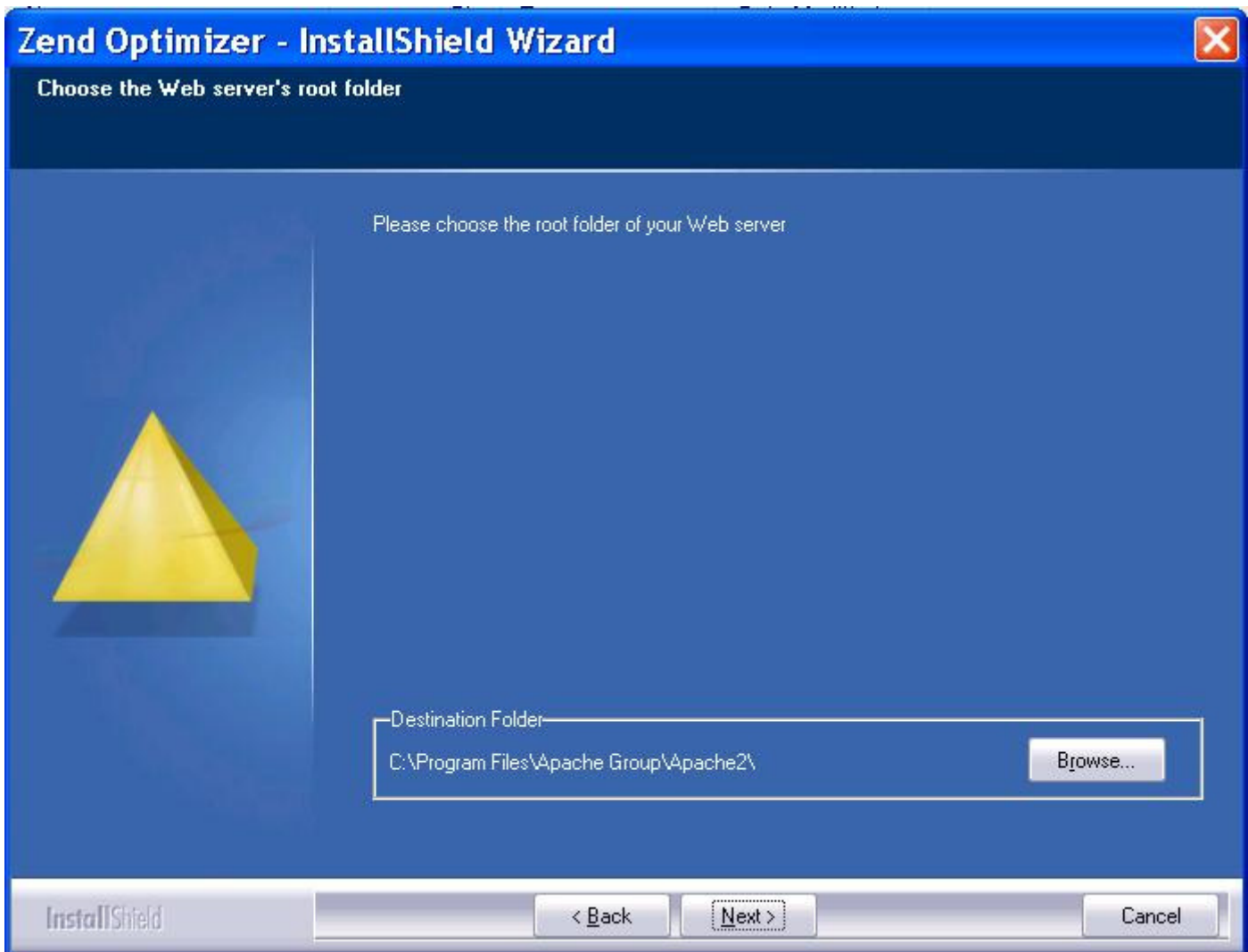
Next we'll need to select the type of web server we're using. In this case we used Apache 2.x, so simply select Apache 2.x from the list and click on Next.



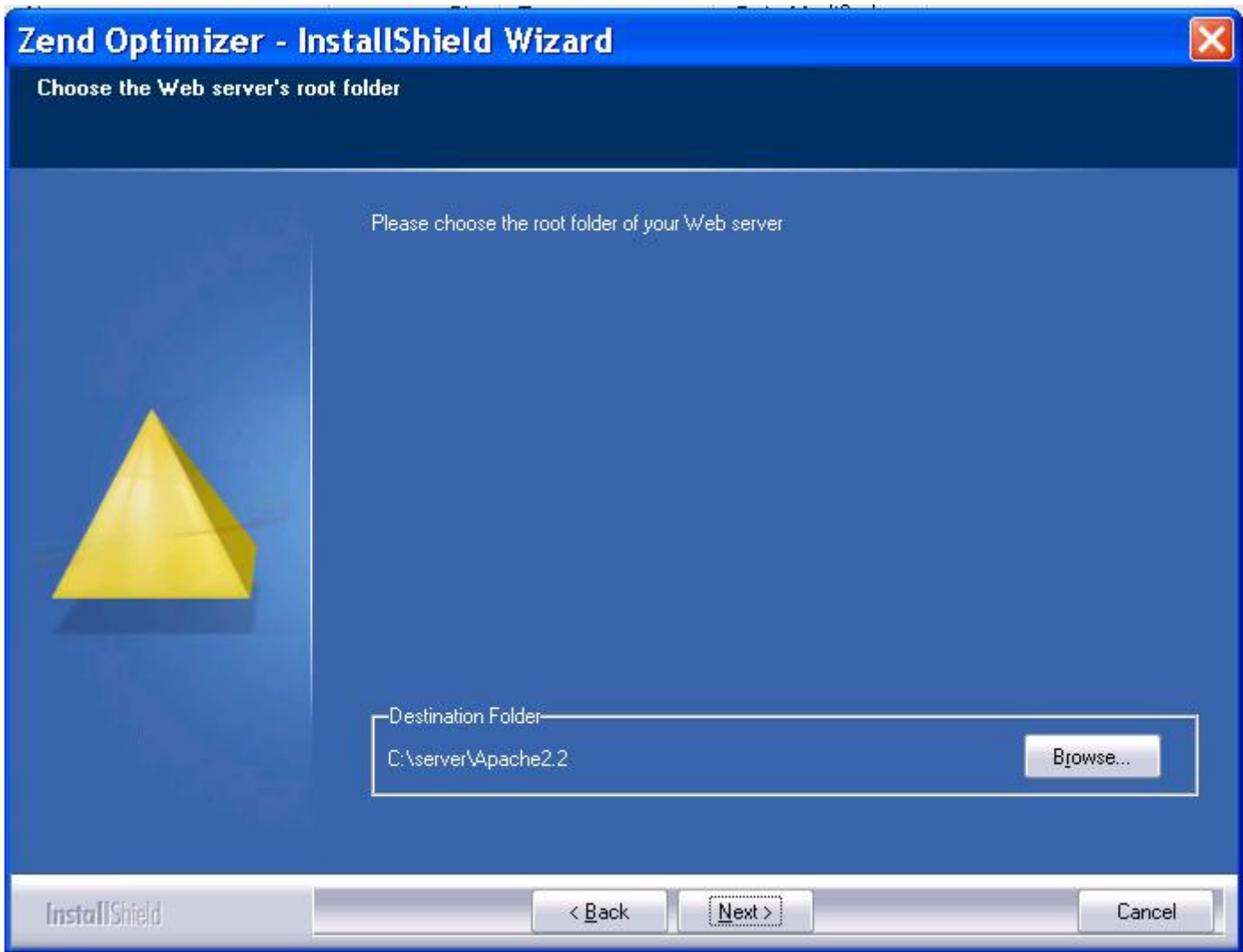
Next the installer would like to know where our “php.ini” file is installed. In this case it’s installed in the “c:\server\PHP” folder. So, click Browse button, and point it to c:\server\PHP.



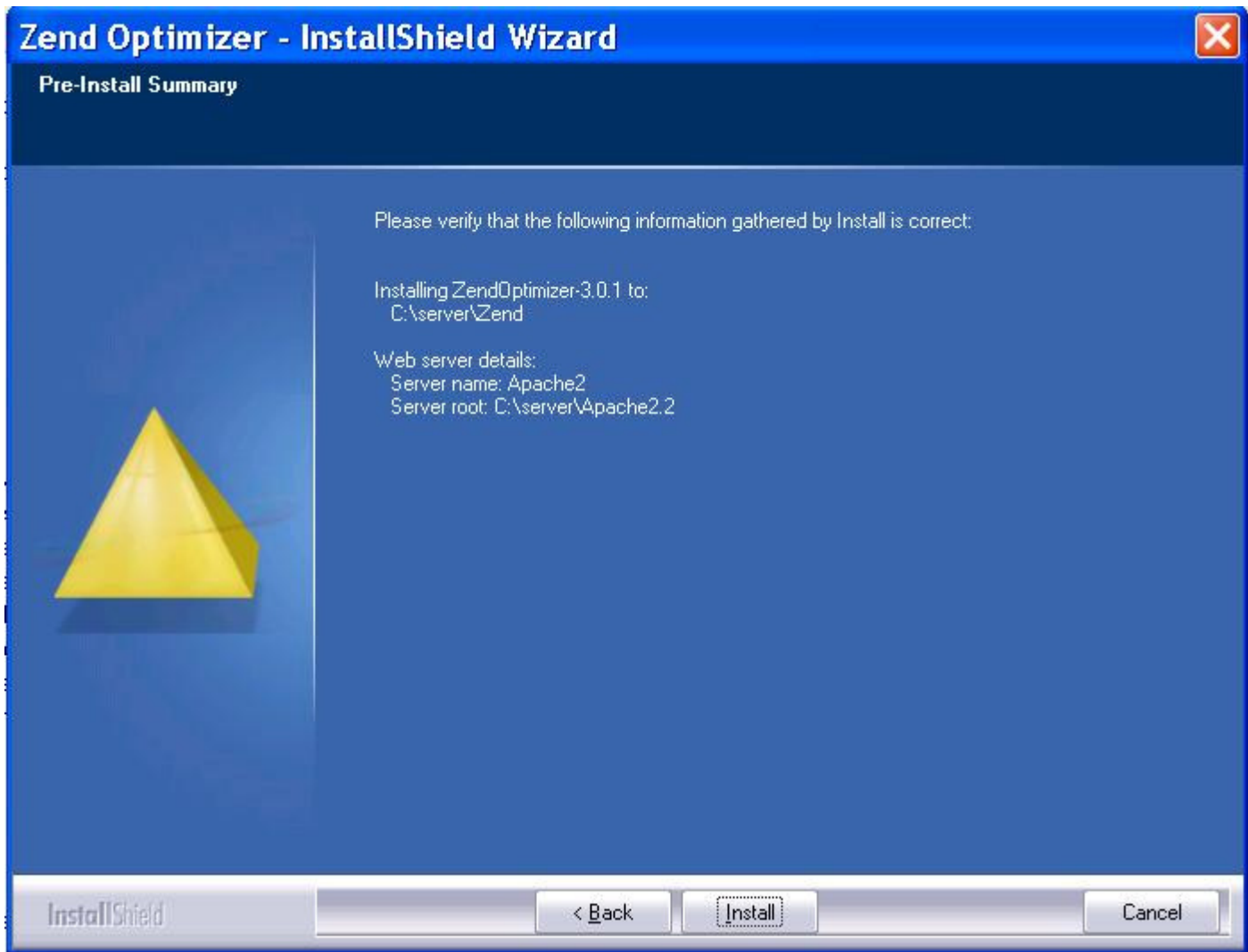
This will look like after we have select the php.ini location



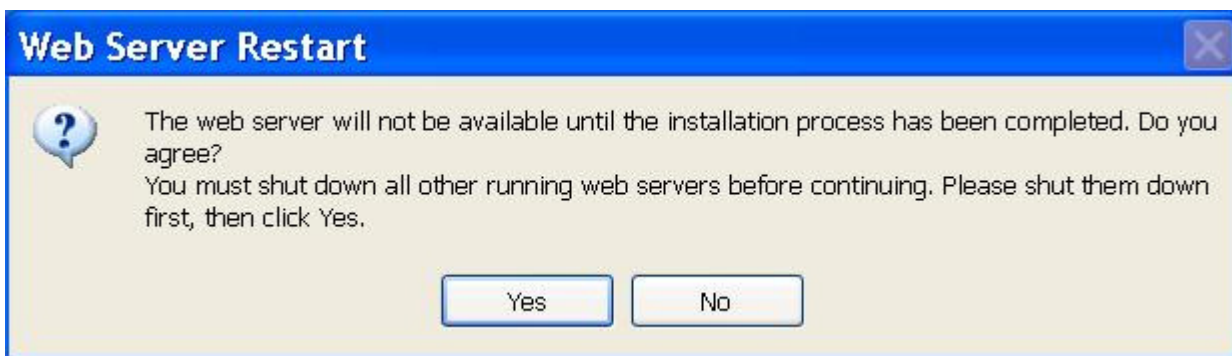
We'll need to specify the location of Apache. In this case the location of Apache was "C:\server\Apache2.2". After this has been confirmed or input correctly, click on Next.



This will look like after we have select the Apache location. Click Next



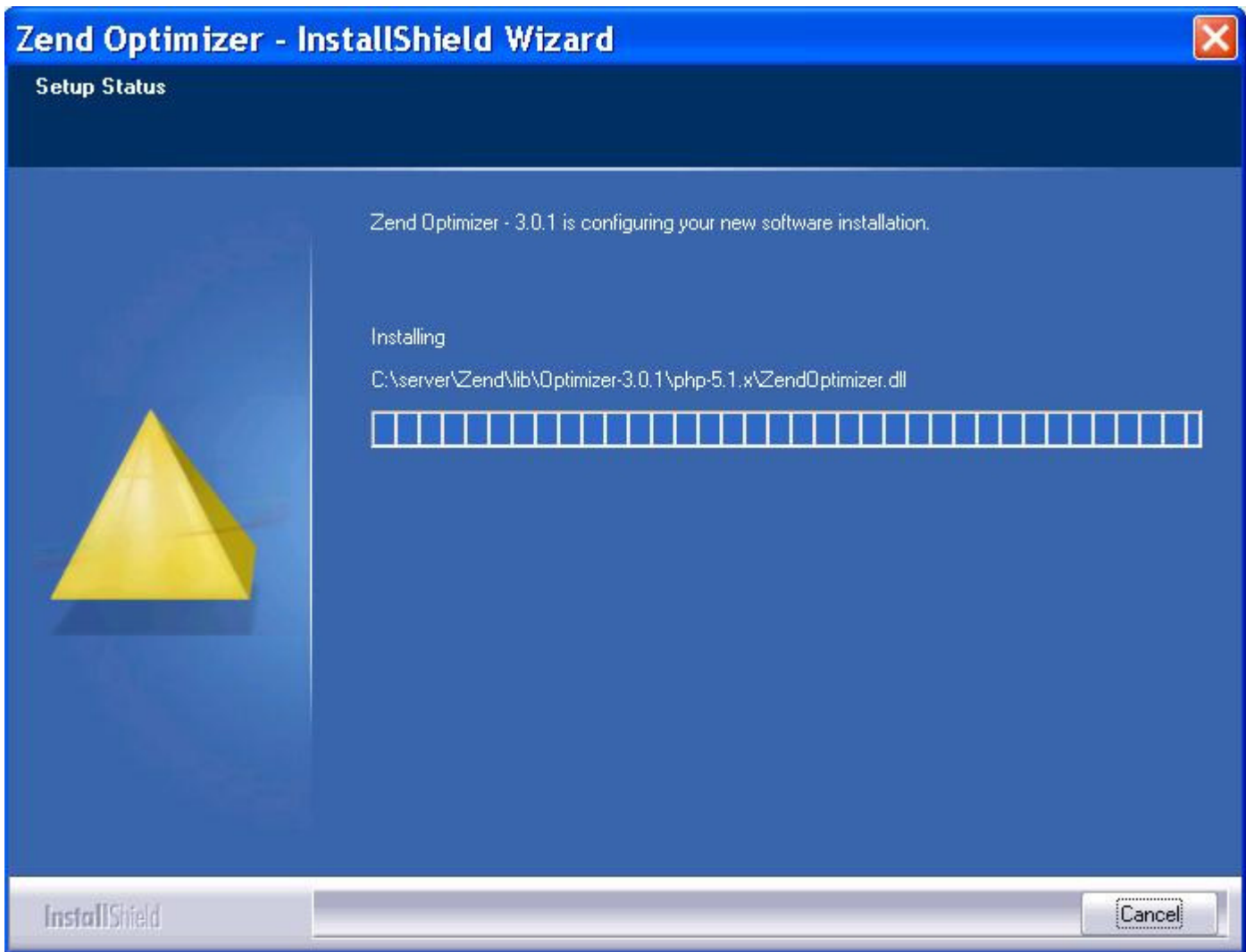
After that, the application will bring us to ready to install window. Simple click on Install and the installer will begin the installation of Zend Optimizer using our custom configurations.



This warning box telling us to stop the Apache web server before continuing the installation. Stop the Apache web server and click Yes to continue the installation



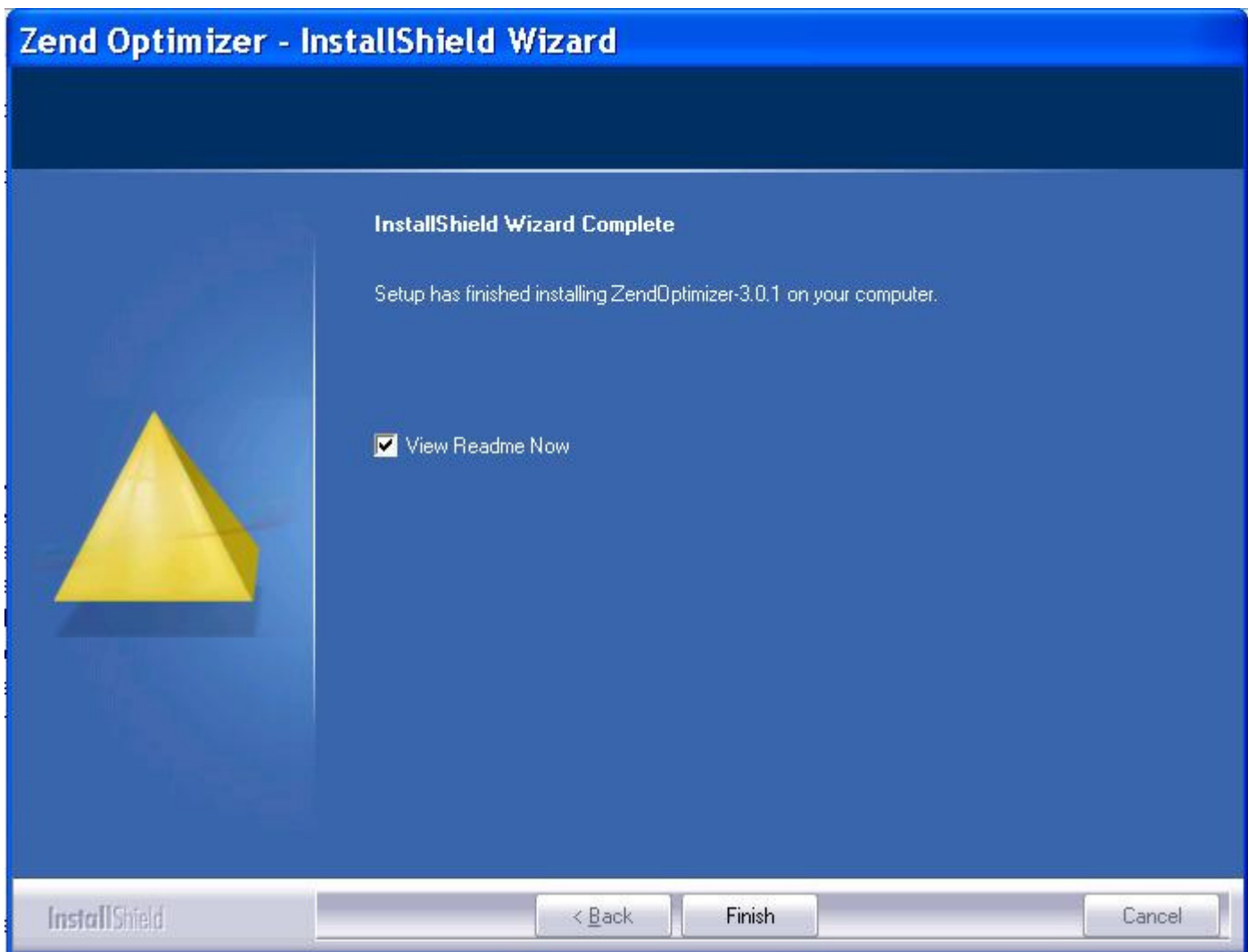
This warning box tells us that the installer failed to find the Apache services and told us to manually stop the Apache web server before continuing the installation. Since we've done already, just click OK to continue the installation.



When it's installing, we'll see a window similar to this and will disappear automatically when finished.



This warning box tells us that the installer failed to find the Apache services and told us to manually start the Apache web server before continuing the installation. Start the Apache web server and click OK to continue the installation.



Once everything is finished, we'll get a window similar to this. Simply click on Finish and we're done. Zend Optimizer is now installed in the location we provided and is configured using the settings we specified during installation.

Now we can test to see if Zend Optimizer is installed and running correctly. Let's open our favorite browser and run the phpinfo() application, just like we did at PHP installation. Simply type in `http://localhost/phpinfo.php` as the URL and hit enter.

This program makes use of the Zend Scripting Language Engine:
Zend Engine v2.1.0, Copyright (c) 1998-2006 Zend Technologies
with Zend Extension Manager v1.0.10, Copyright (c) 2003-
2006, by Zend Technologies
with **Zend Optimizer** v3.0.1, Copyright (c) 1998-2006, by Zend Technologies



If you get a page similar to this (look closely at the highlighted word, the Zend Optimizer word), then everything went just fine and Zend Optimizer is now installed and working correctly.