

ATDB 2004

PHP Handout

Vol. 2 : PHP Installation

SUMMARY

INSTALLING PHP	3
WHERE TO FIND PHP	3
INSTALLING PHP FOR LINUX AND APACHE WEB SERVER	3
SOME CONFIGURE OPTIONS	5
--with-gdbm	6
--with-gd	6
--with-freetype	6
--with-mysql	6
INSTALLING PHP FOR WINDOWS	7

Have a good time for discovering PHP with us...

Written & Composed by Philippe Kislin (Jury 2004)
Helped by Laurent Weinhard & Jean-Charles Lamirel
And Directed by Amos David.

Installing PHP

PHP is truly cross-platform. It runs on the Windows operating system; most versions of Unix, including Linux; and Macintosh OS X. Support is provided for a range of Web servers including Apache (itself open-source and cross-platform), Microsoft Internet Information Server, WebSite Pro, the iPlanet Web Server, and Microsoft's Personal Web Server.

PHP is designed to integrate easily with databases. This feature is one of the factors that makes the language such a good choice for building sophisticated Web applications. PHP supports almost every database currently available, either directly or via Open Database Connectivity (ODBC).

In ATDB, we will use a combination of windows, Apache, and MySQL. All these are free to download and use and can be installed relatively easily on a PC.

Where to find PHP

You can find PHP at <http://www.php.net/>. PHP is open-source software. The PHP Web site is an excellent resource for PHP coders. The entire manual can be read online at <http://www.php.net/manual/>, complete with helpful annotations from other PHP coders. You can also download the manual in several formats.

The current version is the 4.3.2 (the 5 version is still in beta)

You can find out more about getting Linux for your computer at :

<http://www.linux.org/help/beginner/distributions.html>.

If you want to run Linux on a Power PC at :

<http://www.yellowdoglinux.com/>.

If you are running OS X, you can find installation information at :

<http://www.php.net/manual/en/install.macosx.php>.

If you want to run PHP with Windows, you can find complete installation instructions at <http://www.php.net/manual/en/install.windows.php>.

MySQL, one of the databases we will use in ATDB, can be downloaded from <http://www.mysql.com>. Versions are available for many operating systems, including Unix, Windows, and OS/2.

We'll install in the computer science department for the TP(s), the All-in-One TechPlat from Nusphere. With this package, you can install all you need in 5 clicks or less in the windows platform.

Installing PHP for Linux and Apache web server

Here is one way of installing PHP with Apache on Linux. Compiling PHP, though, gives you greater control over the features built in to your binary.

Before you install, you should ensure that you are logged in to your system as the root user. If you are not allowed access to your system's root account, you might need to ask your system administrator to install PHP for you.

You can either recompile Apache, statically linking PHP into it, or you can compile PHP as a dynamic shared object (DSO). If your version of Apache was compiled with DSO support, it is capable of supporting new modules without the need for recompiling the server. This method is the easiest way to get PHP up and running, and it is the one we cover in this section.

To test that Apache supports DSOs, you should launch the Apache binary (`httpd`) with the `-l` argument, like so:

```
/usr/local/apache/bin/httpd -l
```

Where Is Apache Web Server ?

`httpd`, the Apache application, can be installed in different places on a system. One standard location is `/usr/local/apache/bin/httpd`, but you may find that it is somewhere else on your server. If it has been placed in your path, you may not even have to use the full path in order to invoke the application. You would then be able call apache like this:

```
httpd -l
```

You should see a list of modules. If you see

```
mod_so.c
```

among them, you should be able to proceed; otherwise, you might need to recompile Apache. The Apache distribution contains full instructions for this.

Compile Apache with DSO Support

If you do install Apache, remember to ensure that you compile DSO support in. You can do this by passing `--enable-module=so` to the configure script, like this:

```
./configure --enable-module=so
```

If you have not already done so, you need to download the latest distribution of PHP (PHP 5.0.0b1 at the time of writing or the last stable version 4.3.2). Your distribution will be archived as a tar file and compressed with gzip, so you will need to unpack it:

```
tar -xvzf php-5.0.0b1.tar.gz
```

After your distribution is unpacked, you should move to the PHP distribution directory:

```
cd php-5.0.0b1
```

Within your distribution directory you will find a script called `configure`. This accepts additional information that should be provided when the `configure` script is run from the command line. These command-line arguments control the features that PHP supports. For this example, we will include some useful command-line arguments, although you might want to specify arguments of your own.

```
./configure \
--prefix=/home/usr/local/php5/ \
--with-mysql \
--with-apxs=/usr/local/apache/bin/apxs \
--with-xsl \
--with-gdbm \
--with-gd \
--with-freetype=/usr/include/freetype/ \
--with-zlib-dir=/usr/include \
--with-ttf \
--with-jpeg-dir=/usr/lib
```

Installing PHP with Apache 2

It is currently recommended that you run PHP with Apache 1.3 rather than Apache 2. However, you can find full instructions for installing PHP with Apache 2 at <http://www.php.net/manual/en/install.apache2.php>. The main installation difference lies in a flag to the configure script. You should use `--with-apxs2` rather than `--with-apxs`.

The directives chosen in this example are designed to support the features discussed in this book. Most of them require that your system has certain libraries installed before you can compile PHP. The configure script will complain if the relevant libraries cannot be located.

Of these configure options, the one that is absolutely essential is `--with-apxs` because it associates PHP with your server. The argument you use depends on the location of Apache on your server. If you are running Linux and are not sure where to find Apache, try running the `locate` command at the command line, like so:

```
locate apxs
```

It lists all the paths on your system that contain the string `apxs`.

After the `configure` script has run, you can run the `make` program. You need a C compiler on your system to run this command successfully:

```
make
make install
```

These commands should end the process of PHP compilation and installation. You should now be able to configure and run Apache.

Some `configure` Options

When we ran the `configure` script, we included some command-line arguments that determined the features the PHP engine will include. The `configure` script itself gives you a list of available options. From the PHP distribution directory, type the following:

```
./configure --help
```

The list produced is long, so you might want to add it to a file for reading at leisure:

```
./configure --help > configoptions.txt
```

--with-gdbm

The `--with-gdbm` option includes support for the Gnu Database Manager. If your DBM library is in a nonstandard location, you might need to specify a path, as shown here:

```
--with-gdbm=/path/to/dir
```

--with-gd

`--with-gd` enables support for the GD library, which, if installed on your system, allows you to create dynamic GIF or PNG images from your scripts.

```
--with-gd=/path/to/dir
```

If your compile fails, you should try explicitly setting the path when using this option.

Successful option combinations for compiling with GD support are subject to occasional change, so you should check the manual at <http://www.php.net/gd> for the latest information.

At the time of writing, it is also necessary to specify a path to the `zlib` compression library (<http://www.gzip.org/zlib/>) to install GD successfully. We specified the path for the standard location like so:

```
--with-zlib-dir=/usr/include
```

We also want to use the GD library to work with JPEG files, so we have compiled in support for this as well:

```
--with-jpeg-dir=/usr/lib
```

This path points to the standard install directory for the `jpeg-6b` library that can be downloaded from <ftp://ftp.uu.net/graphics/jpeg/>.

--with-freetype

`--with-freetype` provides support for the FreeType 1 library that enables you to include fonts in any dynamic image you create. To enable this option, you must have the FreeType 1 library installed. You can find out more about FreeType at <http://www.freetype.org>. As with many other directives, if you run into problems you should try specifying a path:

```
--with-freetype=/path/to/dir
```

--with-mysql

`--with-mysql` enables support for the MySQL database:

```
--with-mysql=/path/to/dir
```

Installing PHP for Windows systems

With the Techplat 3.4 from Nuphere technology, it's very easy. Just a double click on the installer and your Apache, PHP and Perl are ready to use. We bring this free software with us and let it in the pub directory of the web server of the computer science department.

See also for a non automatic install :

<http://www.thesitewizard.com/archive/php4install.shtml>

http://www.onlamp.com/pub/a/php/2000/12/14/php_admin.html

<http://www.devarticles.com/c/a/Apache/Installing-PHP-under-Windows/>