

Puesta en marcha de un SGBD y un servidor web local

Piero Berni Millet
Dídac Gil de la Iglesia

PID_00155713



Universitat Oberta
de Catalunya

www.uoc.edu



Los textos e imágenes publicados en esta obra están sujetos –excepto que se indique lo contrario– a una licencia de Reconocimiento (BY) v.3.0 España de Creative Commons. Podéis copiarlos, distribuirlos y transmitirlos públicamente siempre que citéis el autor y la fuente (FUOC. Fundació para la Universitat Oberta de Catalunya). La licencia completa se puede consultar en <http://creativecommons.org/licenses/by/3.0/es/legalcode.es>

Índice

Introducción.....	5
1. Puesta en marcha de un SGBD y un servidor local.....	7
1.1. ¿Qué es Apache?	7
1.2. ¿Qué es PHP?	7
1.3. ¿Qué es MySQL?	9
1.4. El funcionamiento del sistema	10
2. Puesta en marcha de un SGBD para Windows (WAMP).....	12
2.1. Instalación de WampServer	12
2.2. Primeros pasos con WampServer	17
2.3. Consideraciones sobre la instalación sobre Windows 7	20
2.4. Consideraciones sobre los ficheros de configuración	22
2.4.1. Fichero de configuración de Apache	22
2.4.2. Fichero de configuración de PHP (PHP.ini)	24
2.4.3. Fichero de configuración de MySQL	25
3. Puesta en marcha de un SGBD para Ubuntu GNU/Linux.....	26
3.1. Requisitos del sistema para instalar la versión 8.10 Intrepid Ibex	27
3.2. Instalación del servidor LAMP desde el escritorio de Ubuntu	27
3.3. Instalación del servidor LAMP desde la consola de Ubuntu	30

Introducción

En este apartado aprenderéis a instalar un sistema de gestión de bases de datos local para Windows con el que tendréis ocasión de publicar datos en Internet de forma dinámica y en tiempo real.

El sistema estará compuesto, entonces, por los siguientes elementos:

- 1) Sistema operativo Windows 95/98/Me/NT/2000/XP/Vista.
- 2) Apache como servidor de web.
- 3) PHP como módulo de ampliación de Apache para acceder a la base de datos.
- 4) MySQL como servidor de base de datos.

En este apartado nos gustaría facilitar al lector la tarea de instalar el sistema completo desde el principio. Para ello, detallaremos los pasos que hay que seguir para la compilación e instalación de los diferentes paquetes, aunque la última palabra siempre la tendrá la documentación de cada paquete en particular. En algún caso, quizá la descripción que hacemos aquí no sea lo suficientemente completa, por lo que el estudiante tendrá que recurrir a las instrucciones de instalación de cada uno de los paquetes.

Una vez completados estos cuatro procesos de instalación, habremos conseguido el objetivo principal de esta guía, objetivo que culminará con la creación de una simple web de ejemplo conectada a una base de datos.

General public license

A excepción de Windows, podemos obtener todos los elementos del sistema sin ningún tipo de coste al tener licencia GPL (*general public license*). Las bases de esta licencia se encuentran especificadas en la dirección de Internet:
<http://www.gnu.org/copyleft/gpl.html>

1. Puesta en marcha de un SGBD y un servidor local

1.1. ¿Qué es Apache?

Apache es un servidor de web. Un servidor web es un software que responde a las solicitudes de los navegadores web. En estos momentos, Apache es uno de los servidores web más populares del mundo. Ello se debe, entre otras cosas, a que Apache es un software de alta calidad y de código abierto (*open source*), lo que significa que puede descargarse de forma gratuita desde Internet.

Apache es uno de los mayores éxitos del software libre y su aceptación entre los servidores web es tan grande que ha llegado hasta el punto de llegar a ser un serio competidor del servidor de web de Microsoft (IIS, *Internet information server*). Desde 1996, Apache es el servidor web más popular de Internet, hasta llegar a la actual cota de un 68% de los servidores web frente un 31% sobre IIS (*Fuente: <http://news.netcraft.com>*). Su desarrollo es continuo y su portabilidad le ha llevado a plataformas como Windows NT/2000/XP y Windows 95/98/Me, a los sistemas Unix y a plataformas como MacOS.

Una de las principales características de Apache es su extensibilidad basada en una gran capacidad de modulación de su código fuente, hecho que ha facilitado la aparición de módulos de extensión como PHP, que evitará el uso de cgi-bin por completo, facilitando así enormemente la programación de aplicaciones en el lado del servidor, en especial en el campo del acceso a bases de datos, así como su agilidad en servir las páginas solicitadas y su seguridad.

1.2. ¿Qué es PHP?

PHP corresponde a las iniciales de *personal home page tools* (herramientas para páginas iniciales personales). Es un lenguaje de programación tipo *script* para entornos web con unas funciones muy semejantes a las de ASP y JSP, utilizado, sobre todo, en servidores Linux para personalizar la información enviada a los usuarios que acceden a un sitio web. Desde un punto de vista técnico, es un lenguaje interpretado de alto nivel, similar en construcciones léxicas y sintácticas a C, C++, Java y Perl, por lo que a quienes ya conozcan estos lenguajes les resultará muy fácil comenzar a escribir código PHP.

PHP es un lenguaje incrustado (*embedded*) en páginas HTML, es decir, es un lenguaje de programación que se introduce dentro de las páginas HTML. El código PHP se interpreta en el lado del servidor de web, desde donde se genera la página HTML solicitada antes de llevar a cabo su transmisión al navegador. De esta forma, podemos programar aplicaciones asociadas al servidor de web, aumentando, así, la funcionalidad de dicho servidor y convirtiéndolo en un

El nuevo PHP

El nuevo PHP, cuya versión es la 5, se ha ampliado, respecto la versión 3, en aspectos tan importantes como conceptos de programación orientada a objetos, y su sintaxis es ahora mucho más cercana a la de C, por lo que cualquier programador que haya programado en C no tardará mucho tiempo en aprender a utilizar el lenguaje. Las construcciones sintácticas de PHP son más cercanas a Perl que a C, ya que en su diseño se buscó un lenguaje útil con el que la programación fuese rápida, es decir, que fuese un lenguaje muy productivo al más puro estilo de Perl.

sistema de desarrollo de aplicaciones cliente/servidor mucho más completo. Su principal objetivo es hacer que desarrolladores de aplicaciones basadas en la web puedan escribir páginas que se generan dinámicamente de una forma sencilla y rápida.

En cuanto a la tecnología del intérprete de PHP, la versión 3 ya era tan rápida como los intérpretes existentes de ASP. Con la versión 4 de PHP, su rendimiento y prestaciones mejoraron todavía más: el intérprete (Zend) era hasta 12 veces más rápido que el de la versión 3; se modularizó todo el diseño interno; se perfeccionó su integración con otros servidores HTTP como el IIS de Microsoft, y se encaró hacia la programación orientada a objetos (Programación OO). Con la versión 5, se ha rediseñado completamente el motor Zend, para crear un lenguaje completamente OO, agilizando más aún su funcionamiento, y extrayendo la compatibilidad con MySQL en un módulo externo (por cuestiones de licencia con MySQL, este SGBD ha dejado de ser "la base de datos" de PHP, para ser una más de las que PHP puede tratar).

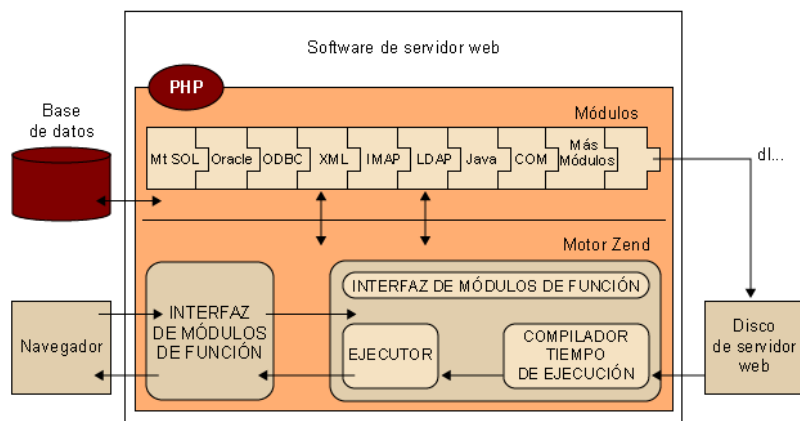
¿Qué es Zend? y ¿qué es PHP?

El nombre Zend se refiere al motor del lenguaje, es decir, el núcleo de PHP.

El término PHP se refiere al sistema completo tal y como aparece desde fuera.

Zend ocupa la parte de **intérprete** (analiza el código de entrada de un *script*, lo traduce y lo ejecuta), y también un poco de la parte de **funcionalidad** (implementa la funcionalidad del sistema). PHP ocupa la parte de funcionalidad y la de **interfaz** (habla con el servidor web, etc.). Juntos forman el paquete completo PHP.

Zend forma realmente el núcleo del lenguaje, mientras que PHP contiene todos los módulos externos (los cuales se pueden cargar en tiempo de ejecución) e incorporados (los que se compilan directamente con PHP) que crean las posibilidades destacadas del lenguaje.



Estructura interna de PHP

PHP proporciona, por tanto, una gran facilidad para acceder a diferentes tipos de bases de datos como Oracle, Sybase, MySQL, PostgreSQL, Adabas, etc. De hecho, es bastante sencillo portar una aplicación escrita con PHP para MySQL a cualquier otro servidor de base de datos, ya que las funciones de acceso que ofrece PHP son, en muchos casos, de sintaxis compartida.

1.3. ¿Qué es MySQL?

MySQL es un sistema de gestión de bases de datos (SGBD) SQL que en algunos aspectos es aproximadamente tan potente como Oracle (<http://www.oracle.com/>). Cabe mencionar que a mediados del año 2009, Oracle, ha adquirido MySQL.

Sus principales objetivos han sido la velocidad y la robustez. Es un SGBD sencillo y rápido que se adapta perfectamente a entornos en los que el volumen de datos sea del orden de megabytes (en la documentación se habla de su uso con bases de datos de 50 millones de registros). En la versión 5 de MySQL ha incluido el control de transacciones, procedimientos almacenados y *triggers*, por lo que ha rellenado el gran hueco que lo diferenciaba de grandes SGBD como Oracle. Si bien existe la posibilidad de comprar su soporte, Oracle sigue teniendo más aceptación en el mundo empresarial.

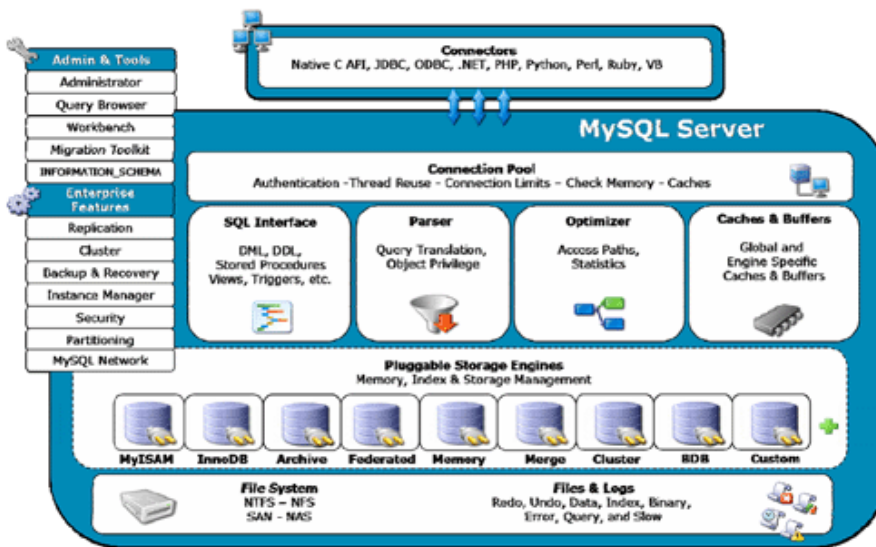
En el último *benchmark* realizado de SGBD, se ha visto un gran crecimiento en la potencia de MySQL, frente a sus competidores.

Sistema Operativo	SGBD	Lenguaje	Operaciones por minuto
Linux	MySQL 5.0	PHP 5	3664
Linux	Pico Lisp 2.2.1	Pico Lisp 2.2.1	2600
Linux	MonetDB 4.9	Java	1833
Windows	MySQL 5.0	Java	1798
Windows	MySQL 4.1	Java	1564
Windows	MySQL 4.1	PHP 5	1542
Linux	DB2 Express 8.2	Java	1537
Linux	Oracle 10g Express	Java	1412
Linux	Sieben Geisslein	Java	600
Linux	MySQL 5.0	Java	587
Windows	MySQL 5.0	PHP 5	444
Windows	MySQL 5.0	Python 2.4	137
Linux	PostgreSQL 8.1	PHP 5	120

En estos últimos años destacadas compañías de software han desarrollado aplicaciones SQL de uso libre y con código fuente (*open source*). En el mundo de GNU/Linux MySQL es, junto a Postgres (<http://www.postgresql.org/>), uno de los SGBD más populares.

MySQL también puede verse como un conjunto de aplicaciones o *pluggins* funcionando en conjunto.

Existen en su versión actual distintos motores de almacenamiento de datos, entre los que destacan MyISAM (permite índices por cadenas completas) y InnoDB (que permite el uso de transacciones) o la incorporación de *buffers* en memoria que permiten agilizar la respuesta de sus resultados.



MySQL se encuentra, igual que PHP, en fase de pleno desarrollo; se están publicando con regularidad nuevas versiones del sistema, así como herramientas que son básicas en cualquier SGBD actual:

- Dispositivo JDBC para acceder desde Java.
- Dispositivo ODBC para acceder utilizando la API ODBC.
- API de programación para C, Perl, C++, Python y TCL.
- Acceso desde PHP.
- Entornos visuales de gestión de la base de datos.
- Control de acceso basado en una base de datos de administración.

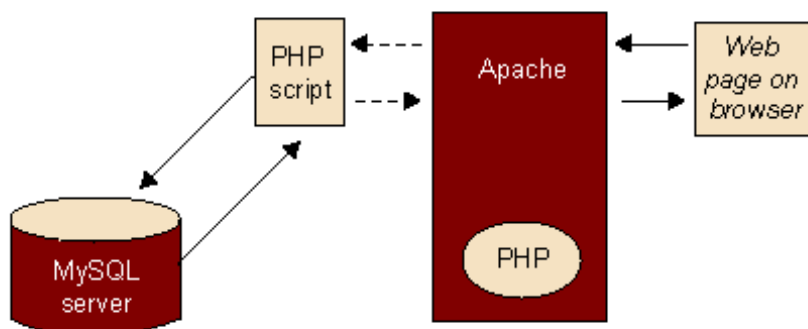
1.4. El funcionamiento del sistema

Como una motivación previa y para que el lector pueda tener una visión global desde el principio, presentamos en este apartado una visión global del sistema. En el siguiente gráfico aparecerán destacadas las partes fundamentales del mismo.

Los pasos que describen la interacción entre el usuario y la base de datos son:

1) El usuario carga una página HTML con un formulario, rellena los datos y se los envía al servidor web.

- 2) Por medio de la red TCP/IP los datos llegan a Apache.
- 3) El servidor detecta que el usuario solicita una página PHP, por lo que informa al módulo de PHP del programa que hay que ejecutar y le pasa los datos del formulario.
- 4) El módulo de PHP ejecuta el programa, el cual accede a MySQL utilizando, de nuevo, una comunicación TCP/IP.
- 5) MySQL procesa la petición del programa PHP y le envía los resultados de vuelta.
- 6) El módulo PHP recibe los resultados del servidor de base de datos, les da formato en una nueva página HTML y se los devuelve al cliente mediante el servidor Apache.
- 7) El cliente recibe la página HTML resultado de su petición por medio de la red TCP/IP.



En este primer esquema ya podemos ver que la interacción con la base de datos se hace, en su totalidad, por medio de PHP.

2. Puesta en marcha de un SGBD para Windows (WAMP)

WAMP es el acrónimo para instalaciones Apache, MySQL, PHP/Perl/Python sobre la plataforma Windows que permite la publicación de páginas dinámicas sobre la Web.

WampServer facilita la instalación de los módulos descritos anteriormente, ya que consiste en una sola aplicación que instalar que contiene los servicios para generar un servidor WAMP. Por su simplicidad de instalación, WampServer es una herramienta de gran utilidad a la hora de desarrollar páginas web dinámicas.

Existen en el mercado otras soluciones integradas para la creación de servicios web dinámicos, basados igualmente en MySQL, Apache y PHP/Perl/Python. Una solución bastante distribuida es XAMPP, y existen versiones para Windows, Linux, Mac y Solaris.

A continuación describiremos el proceso de instalación del paquete WampServer en la plataforma Windows XP y su posterior configuración. Seguidamente, presentamos algunos puntos que se deben considerar para instalaciones realizadas sobre Windows 7. Finalmente, daremos una descripción básica sobre los ficheros de configuración de Apache, PHP y MySQL que es conveniente saber para entender cómo solventar algunos problemas o limitaciones que un desarrollador puede encontrarse durante la creación de un portal web dinámico.

2.1. Instalación de WampServer

El proceso de instalación del servicio WAMP con WampServer es muy sencillo.

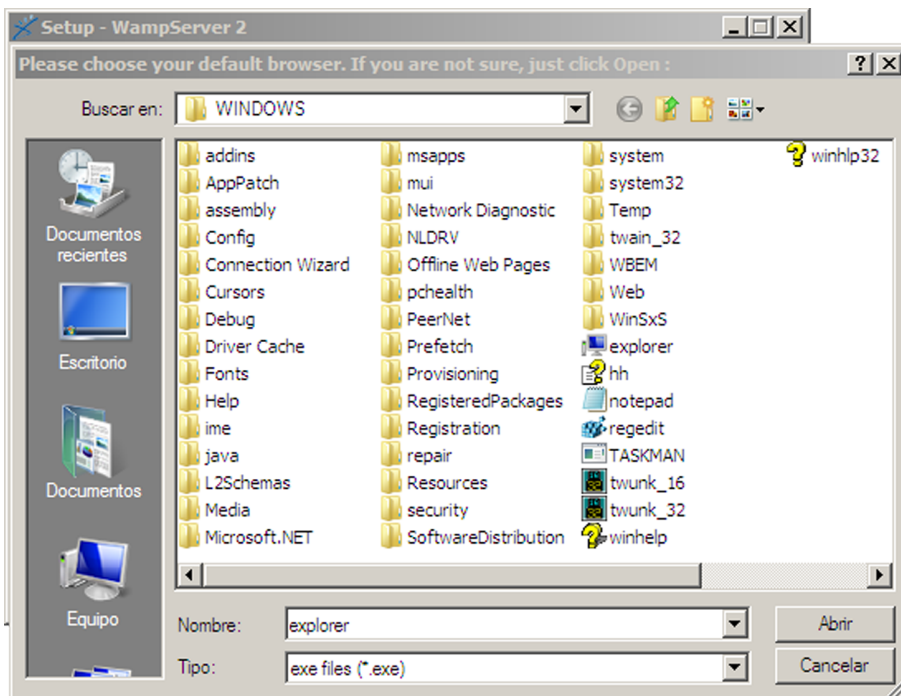
Tras la ejecución del instalador con nombre WampServer2.0i.exe aparece la pantalla de bienvenida al inicio de la instalación. Debemos simplemente presionar sobre el botón “Siguiente” para empezar con el proceso de instalación.

Descargarse la versión más reciente

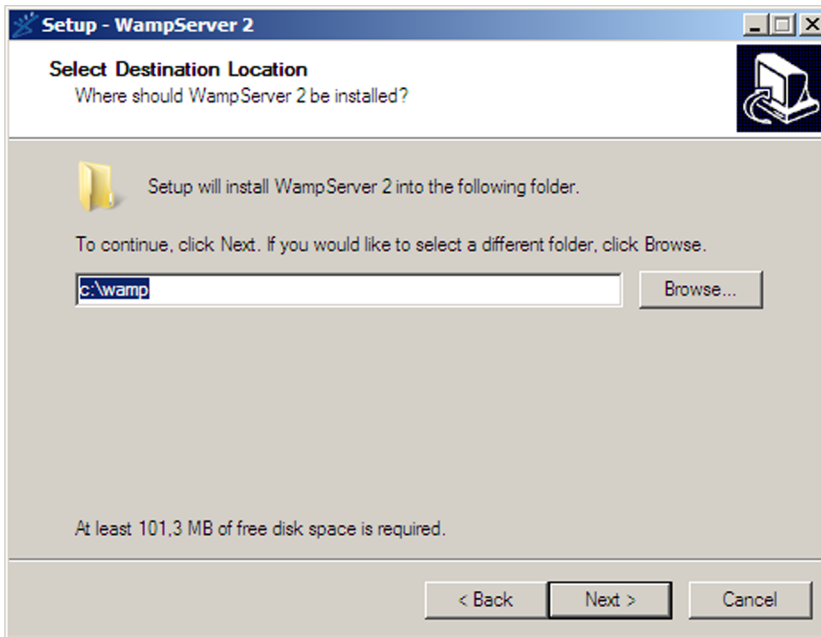
De la página web oficial del producto WampServer puede descargarse la versión más reciente del servidor. Las capturas de pantalla que se ofrecen en este capítulo se han realizado sobre la instalación del producto WampServer 2.0i con fecha de lanzamiento 11 de julio del 2009.



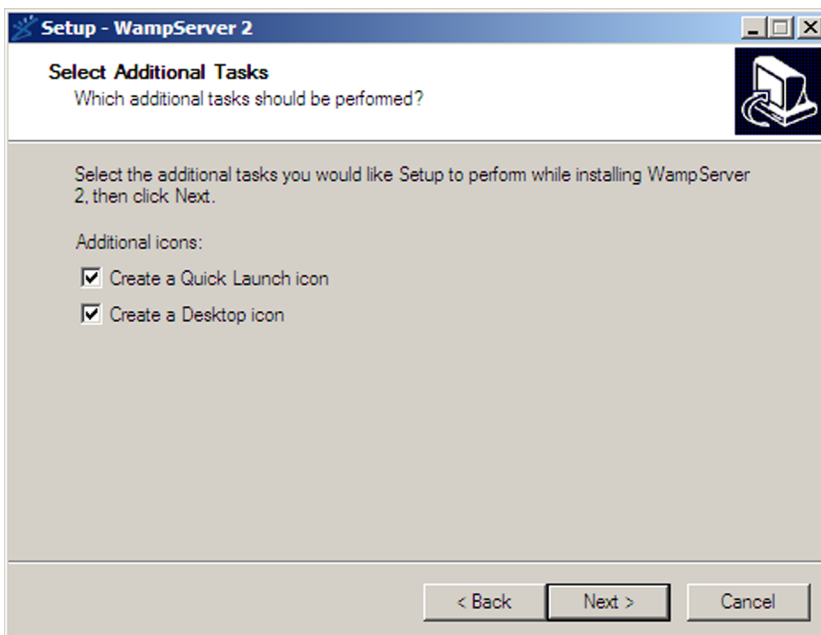
Algunos de los enlaces de WampServer, como por ejemplo para acceder a phpMyAdmin para la gestión de las bases de datos, se realizan a través del navegador web. El instalador nos preguntará en este punto cuál es el navegador web que desearemos usar. En la captura siguiente se muestra cómo se ha elegido el navegador Explorer.exe, es decir, el Internet Explorer existente en la plataforma Windows.

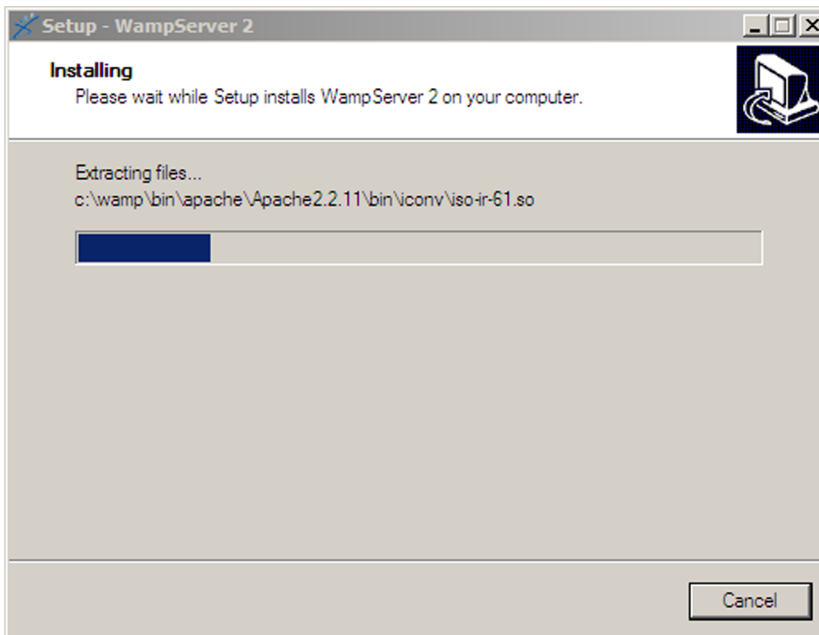
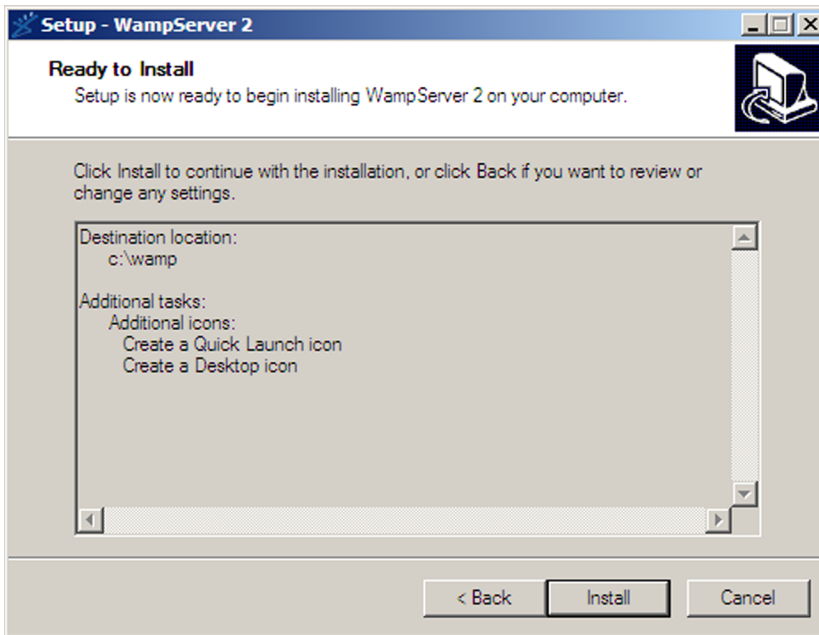


Después de aceptar la licencia GNU para la instalación del software WampServer, se solicita la dirección donde instalar la aplicación. Por defecto, la instalación se realizará en la carpeta C:\wamp.



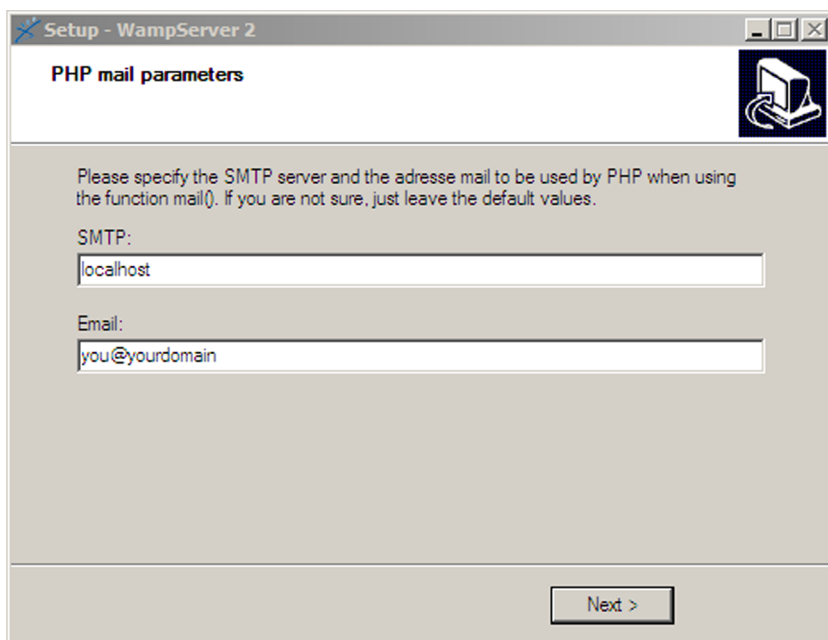
Una vez entrada la ruta de la instalación, presionaremos en el botón “Siguiente” e “Instalar”. Con ello tendremos la aplicación WAMP instalada en nuestro servidor local, pero será necesario un proceso de configuración del servidor web dinámico, muy sencillo también.



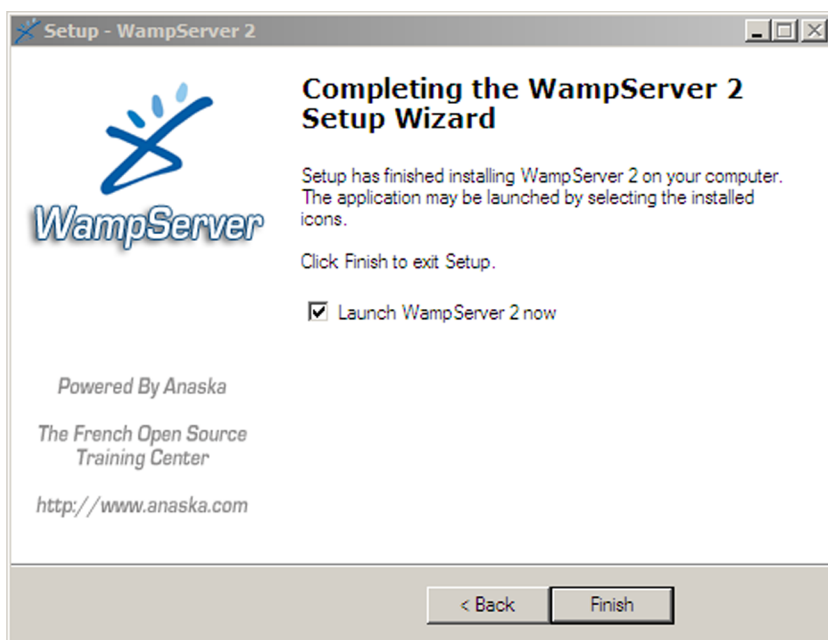


El servidor web WampServer, mediante librerías PHP, permite el envío de correos electrónicos. Un ejemplo cotidiano de envío de correos electrónicos es para el alta a servicios web, tales como registro en foros. El primer paso de configuración del servidor web dinámico es la entrada de los parámetros para usar un servidor de envío de correos electrónicos mediante protocolo SMTP. En caso de disponer de un servidor de correos, entraremos la dirección del servidor de correo electrónico y la dirección de correo electrónico desde donde se mandarán los correos.

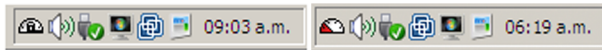
En caso de no disponer de servidor de correo electrónico, simplemente presionaremos en el botón "Siguiente".



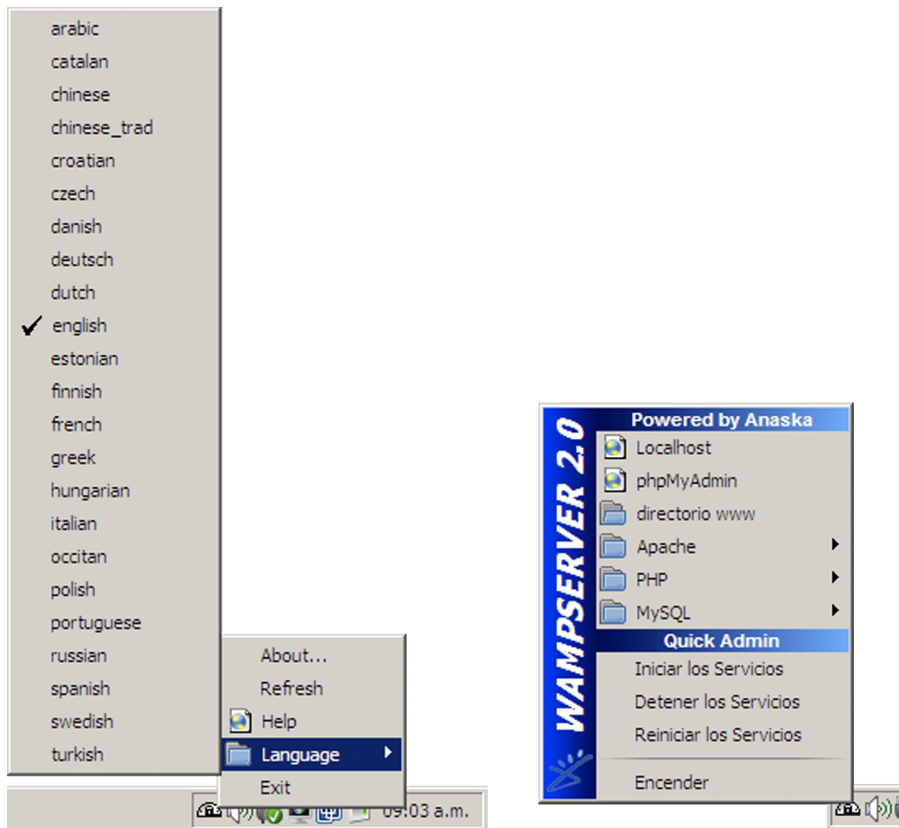
Para finalizar, comprobaremos que la casilla “Iniciar WampServer 2 ahora” está activada y presionaremos en el botón “Finalizar” para finalizar con el proceso de configuración del servidor web dinámico.



Observando la barra de tareas de Windows, podemos detectar que WampServer está en ejecución (el icono en blanco mostrará que el servicio está encendido, mientras que el icono en rojo determina que está apagado). Pinchando sobre el icono de WampServer podemos modificar ciertos parámetros de configuración, como el idioma de los menús de configuración, encender o apagar los servicios web o de bases de datos, activar módulos de Apache, activar extensiones para PHP, etc.

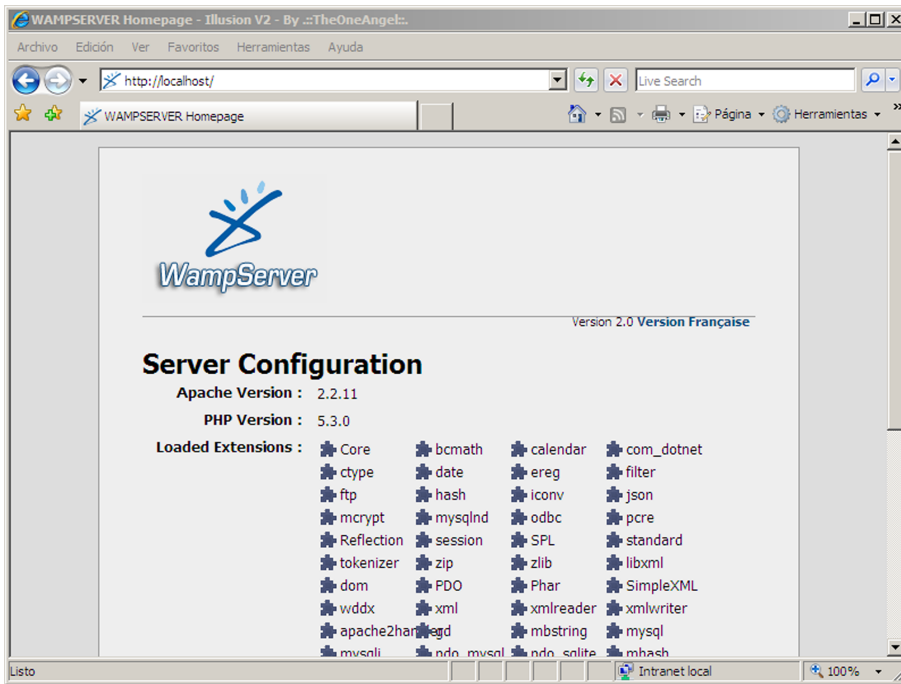


2.2. Primeros pasos con WampServer



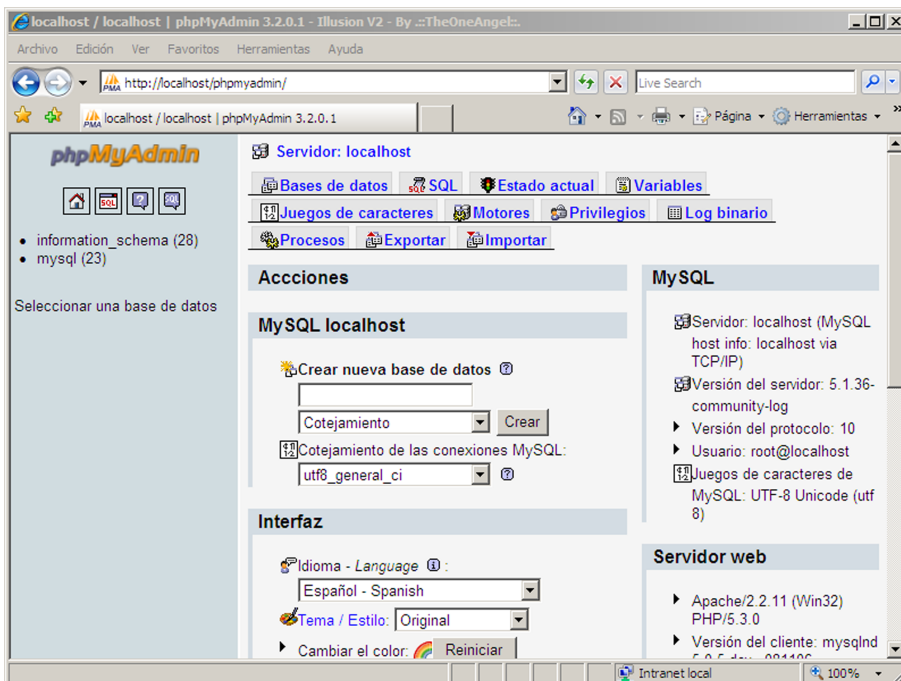
Una vez terminada la instalación y configuración del servidor web dinámico con WampServer, comprobaremos que su ejecución es correcta. Por defecto, cuando llamamos a la raíz del árbol web, se nos mostrará una pantalla de bienvenida con la configuración actual que está usando el servidor web, así como las extensiones de Apache que tiene activadas en dicha ejecución. Encontraremos también un enlace a `phpinfo()` para ver los módulos activados en el servidor web dinámico a través de PHP.

Para poder ver esta pantalla de bienvenida, abriremos un navegador y escribiremos la dirección “`http://localhost`”, o pincharemos sobre el enlace “Localhost” que se muestra en el menú de WampServer.

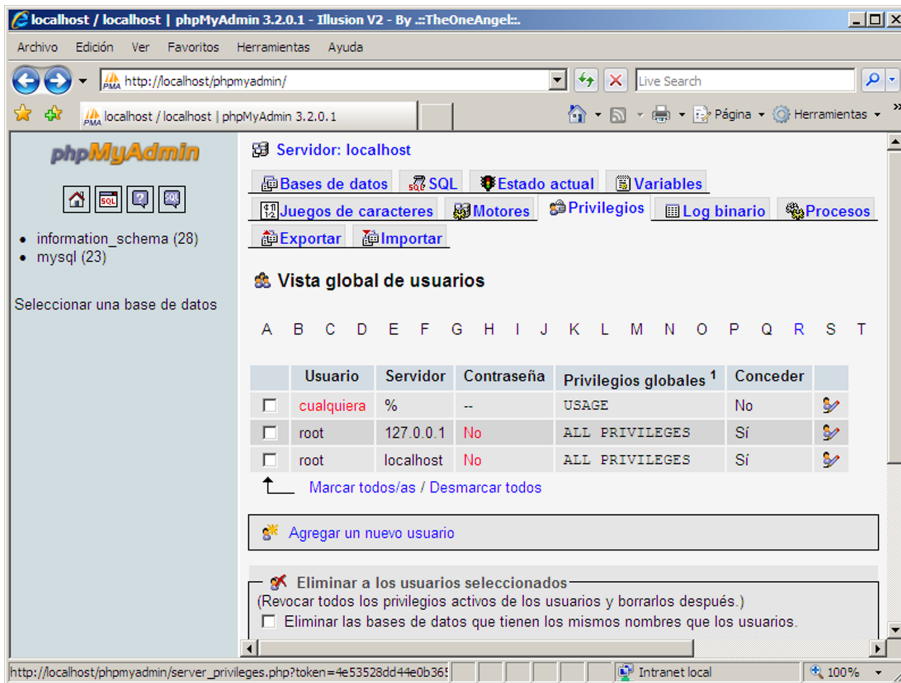



La gestión de las bases de datos podremos realizarla con paquetes gráficos como MySQL GUI Tools, que podemos descargar de MySQL.

Por otro lado, WampServer contiene la herramienta web phpMyAdmin, que está disponible a través de phpMyAdmin, o por medio de uno de los enlaces ofrecidos en la página de bienvenida de WampServer.



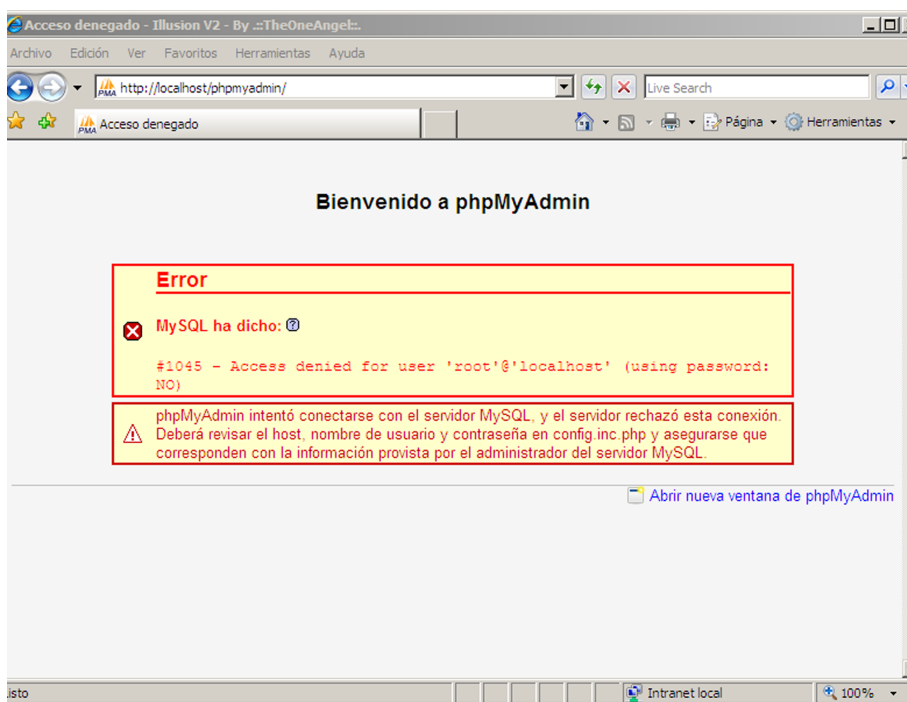
Como podremos comprobar, por defecto el usuario administrador (root) de MySQL está configurado sin contraseña, por lo que la seguridad de acceso a MySQL no es la recomendada. De todos modos, modificar la clave de acceso del administrador a MySQL no es complicado. Para hacerlo, realizaremos los siguientes pasos:



1) Una vez dentro de phpMyAdmin, entraremos en la sección de “Privilegios” e editaremos el usuario root del servidor 127.0.0.1 pinchando sobre el botón de edición .

2) En la sección “Cambio de contraseña”, estableceremos la clave de acceso que queremos usar para el usuario *root* para las conexiones a las bases de datos y presionaremos sobre el botón “Continuar” para almacenar los cambios. Desde los Estudios de Graduado Multimedia recomendamos que se establezca como clave “BBDDGM”, acrónimo de “Bases de Datos–Graduado Multimedia”.

3) Repetiremos el proceso para cambiar la clave del usuario root en localhost, estableciendo la misma contraseña usada en el usuario root en 127.0.0.1.



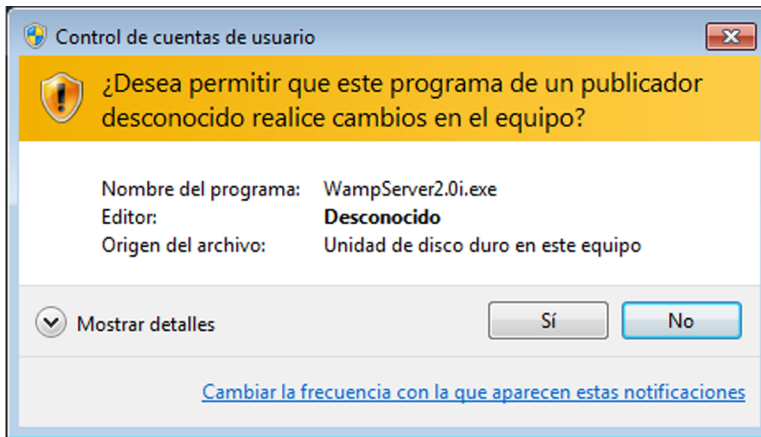
4) Posterior a estos cambios, phpMyAdmin no podrá acceder a las bases de datos para su gestión, dado que está configurado para usar el usuario root sin contraseña. Para permitir a phpMyAdmin que pueda volver a gestionar las bases de datos existentes en MySQL, deberemos modificar el fichero config.inc.php existente dentro de la carpeta C:\wamp\apps\phpmyadmin3.2.0.1 para que la contraseña corresponda con la que hayamos establecido. Hay que tener en cuenta que el directorio “wamp” podrá cambiar en caso de haber realizado la instalación en otro directorio.

```
$cfg['Servers'][$i]['user'] = 'root';  
$cfg['Servers'][$i]['password'] = 'BBDDGM';
```

Una vez actualizado el fichero de configuración de phpMyAdmin, el acceso a phpMyAdmin volverá a ser funcional, por lo que sólo será necesario refrescar el navegador web para acceder a <http://localhost/phpmyadmin>.

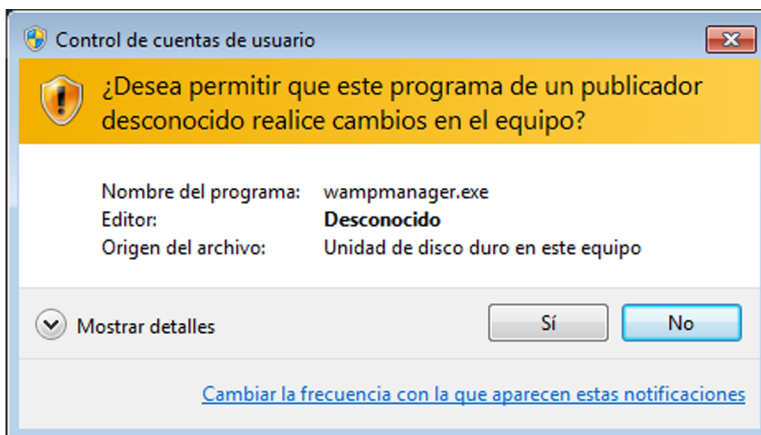
2.3. Consideraciones sobre la instalación sobre Windows 7

El proceso de instalación en Windows 7 sólo se diferencia al de la plataforma XP en lo que a controles de seguridad se refiere. Al ejecutar el instalador, Windows nos prevendrá de que el ejecutable quiere modificar el registro de Windows.

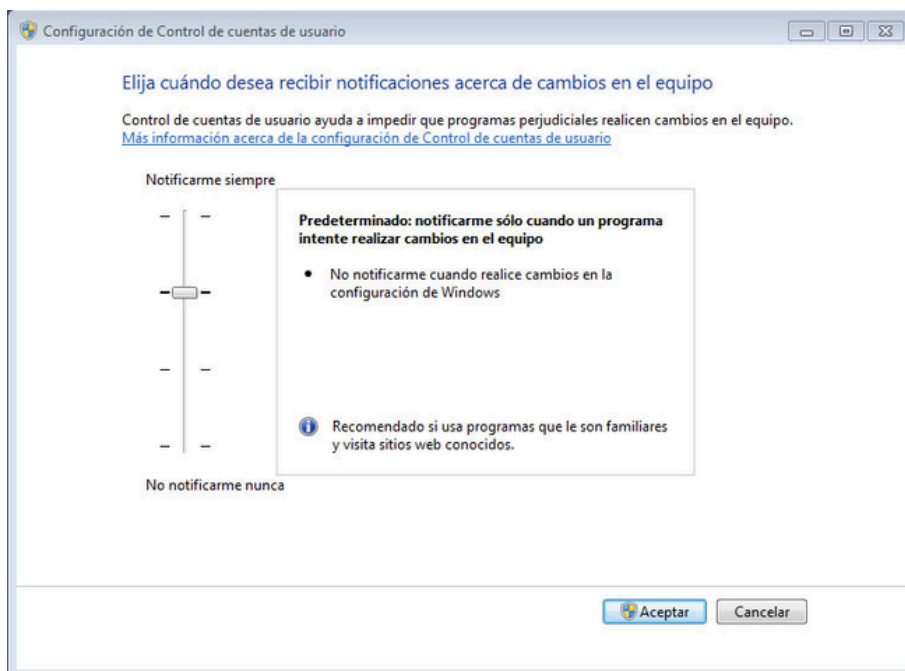


Para poder realizar la instalación, deberemos aceptar que Wamp abra puertos para que nuestros servicios puedan ser accedidos desde otros ordenadores. Por ello, Windows 7 nos notifica sobre el riesgo de que una aplicación esté permitiendo conexiones remotas para acceder a recursos de nuestro ordenador.

Wampmanager.exe es el fichero binario que Windows detecta como posible amenaza de seguridad para nuestro ordenador. Deberemos aceptar la ejecución del servicio presionando el botón "Sí".



Hasta este punto, dada la configuración de seguridad de Windows 7, sería necesario aceptar el mensaje de aviso de amenaza de seguridad cada vez que queramos activar los servicios ofrecidos por WampServer. En Windows 7, es posible regular el nivel de avisos del sistema operativo. Para evitarlo, podemos bajar el nivel de seguridad al mínimo en la Configuración de Control de cuentas de usuario. Dicha modificación puede realizarse desde el Menú de Inicio → Panel de control → Cuentas de usuario → Cambiar configuración de Control de cuentas de usuario.



2.4. Consideraciones sobre los ficheros de configuración

El uso de los servicios Apache, PHP y MySQL mediante paquetes como Wamp-Server provoca que la instalación tenga gran sencillez, pero, por otro lado, pasan a ignorarse muchos de los factores que pueden alterarse para la configuración de los servicios.

A continuación presentamos algunos parámetros de los ficheros de configuración que se deben tener en cuenta durante el desarrollo de portales web dinámicos.

2.4.1. Fichero de configuración de Apache

El fichero de configuración de Apache (httpd.conf) contiene los parámetros para el inicio del servicio Apache.

Como ya hemos mencionado, Apache es el servicio que permite publicar sitios web en nuestro servidor. Para que un sitio web publicado en Apache pueda consultarse desde otros ordenadores, es necesario que el servicio escuche de un puerto para atender a peticiones HTTP que recibe. Las peticiones http, que son las realizadas a servidores web, suelen enlazarse al puerto 80. Por defecto, Apache está configurado para arrancar el servicio http en el puerto 80. Existen otros servicios que se asocian también al puerto 80, tales como el servicio web IIS o Skype. Por ello, en ocasiones es una buena solución enlazar el servicio web Apache a un puerto alternativo, comúnmente el 8080. El parámetro Listen está

Ejemplo

Por ejemplo:
`http://localhost:8080/info.php`
en caso de que el puerto por defecto se haya cambiado por el puerto 8080.

orientado a este efecto. Deberemos tener en cuenta que en caso de cambiar el puerto de Apache, las peticiones que hagamos en el navegador web deberán indicar el puerto al que deben lanzarse.

```
Listen 127.0.0.0:8080
```

En ocasiones, pueden darse errores en el servidor web o éste puede incluso dejar de funcionar. En estos casos, suele ofrecerse una dirección de correo electrónico para contactar con el administrador del portal web. En el fichero `httpd.conf` existe un parámetro para definir dicha dirección de correo electrónico:

```
ServerAdmin admin@localhost
```

Los ficheros publicados en un servidor web se encuentran a partir de una carpeta inicial, llamada Raíz del servidor web. La carpeta raíz también es un parámetro configurable dentro de Apache. Por defecto, durante la instalación de Apache, se establece `c:/wamp/www` como carpeta para la ubicación de los documentos web, pero se puede cambiar con la siguiente directiva:

```
DocumentRoot "c:/wamp/www/"
```

Normalmente, las peticiones web solicitan un documento en concreto, como por ejemplo `http://servidor.web.edu/servicio1/documento1.php`.

Pero no siempre se solicita específicamente un documento de la web, como por ejemplo en `http://servidor.web.edu/servicio1/`. En dicho caso, el usuario quiere obtener el documento de índice que se encuentra en el directorio "servicio1". Para indicar qué nombre y extensión pueden tener los posibles ficheros de índice y su prioridad, se utiliza la directiva `DirectoryIndex`.

En el siguiente ejemplo, primero se comprobará la existencia del fichero `http://servidor.web.edu/servicio1/index.php`, posteriormente `index.php3`, y así sucesivamente.

```
<IfModule dir_module>
    DirectoryIndex index.php index.php3 index.html index.htm
</IfModule>
```

Durante el desarrollo de sitios web, a menudo podemos encontrarnos con que el servidor no responde a las peticiones de la manera esperada. Los registros del servidor pueden servir de mucha utilidad para detectar los motivos por los que un servicio web actúa incorrectamente. La directiva `LogLevel` nos permite establecer el nivel de *log* que queremos registrar.

```
LogLevel warn
```

Las líneas de configuración que empiezan con el comando `AddType` detallan a Apache cómo debe interpretar diferentes tipos de ficheros a partir de su extensión. Por ejemplo, ficheros con extensión `.tgz` deberán ser abiertos por descompresores GZip, o ficheros con extensión `.php`, deberán ser interpretados por la aplicación `httpd-php` o, en otras palabras, el motor para el lenguaje PHP.

```
AddType application/x-compress .Z
AddType application/x-gzip .gz .tgz
AddType application/x-httpd-php .php
```

Dada la cantidad de parámetros y opciones que pueden configurarse en Apache, existe también la posibilidad de modularizar los ficheros de configuración, creando un fichero de configuración principal `httpd.conf`, y varios ficheros de configuración complementarios. Los ficheros complementarios deben ser declarados en el fichero de configuración principal (`httpd.conf`) para que, al arrancar el servicio, sus parámetros sean incluidos. Esto se realiza mediante la directiva `Include`.

```
Include conf/extra/httpd-ssl.conf
```

2.4.2. Fichero de configuración de PHP (PHP.ini)

Dentro del fichero de configuración `php.ini` se describen, entre otras cosas, las extensiones de PHP que se deben activar, las rutas para la subida de ficheros mediante protocolo POST, los tiempos de ejecución de ficheros PHP y el uso de la memoria RAM.

Los documentos con extensión PHP pueden combinar código HTML con código PHP. El código PHP debe diferenciarse del código HTML, ya que el código PHP debe ser interpretado. Normalmente, el código PHP se delimita mediante las llaves `<?php y ?>`, pero es posible hacerlo también mediante las llaves `<? y ?>`. Por defecto, las llaves cortas (`short_open_tag`) no están habilitadas en WampServer, por lo que deberá modificarse el valor del parámetro `short_open_tag` (existe la opción de activar las claves cortas mediante el menú de WampServer).

```
short_open_tag = Off
```

Los documentos ASP usan las llaves `<% y %>` para identificar el código que debe ser interpretado. Si deseamos que nuestras aplicaciones PHP usen las llaves `<% y %>`, deberemos activar el parámetro `asp_tags`.

```
asp_tags = On
```


La complejidad de documento PHP puede ser elevada, lo que en algunos casos puede implicar tiempos de ejecución largos y gran cantidad de recursos consumidos. El tiempo de ejecución de un fichero PHP, así como la memoria que su ejecución requiere también son parámetros que pueden definirse dentro del fichero `php.ini`.

```
max_execution_time = 30
max_input_time = 60
memory_limit = 128M
```

Del mismo modo, puede declararse el tamaño máximo permitido para la subida de ficheros.

```
post_max_size = 8M
```

Así como hemos visto con Apache, PHP también genera trazas de log que son de gran ayuda para la interpretación de errores durante el desarrollo de aplicaciones dinámicas en PHP. En PHP, el nivel de registro de errores puede definirse de la siguiente manera:

```
error_reporting = E_ALL
```

Dentro del propio fichero `php.ini` se describe información sobre el efecto de cada uno de los parámetros que contiene, por lo que recomendamos realizar al menos una lectura del documento de configuración para familiarizarse con las posibilidades existentes.

2.4.3. Fichero de configuración de MySQL

La configuración de sistema gestor de bases de datos MySQL vía la edición de su fichero de configuración (`MY.ini` o `my.cnf` en instalaciones independientes de MySQL) no suele hacerse con tanta frecuencia como en Apache y PHP. No obstante, contiene parámetros de configuración que pueden ser de gran interés.

Normalmente, las comunicaciones hacia MySQL se realizan por protocolo TCP/IP sobre el puerto 3306. Éste es el puerto por defecto en el que MySQL se registra para aceptar peticiones SQL. Este puerto es configurable y puede ser cambiado por un puerto alternativo. El siguiente comando permite cambiar el puerto de MySQL por defecto por el 3333.

```
port = 3333
```

3. Puesta en marcha de un SGBD para Ubuntu GNU/Linux

Linux es un término genérico para referirse a sistemas operativos similares a Unix basados en el núcleo de Linux. Las variantes de estos sistemas se denominan distribuciones de Linux y cada distribución está enfocada para satisfacer las necesidades de un grupo específico de usuarios.

El objetivo de la denominación GNU/Linux (GNU es un acrónimo recursivo que significa '*GNU is Not Unix*') es crear un sistema operativo completamente libre. La base de cada distribución incluye el núcleo Linux, con las bibliotecas y herramientas del proyecto GNU y de muchos otros proyectos/grupos de software.

Ubuntu es una distribución GNU/Linux que ofrece un sistema operativo enfocado a computadores personales con soporte para servidores. Está basado en Debian GNU/Linux (distribución Linux sin ánimo de lucro más popular del mundo) y soporta, principalmente, dos arquitecturas de hardware: Intel x86 y AMD64. Esta distribución ha sido traducida a numerosos idiomas y concentra sus objetivos en la facilidad y libertad de uso. Sus versiones estables se liberan cada 6 meses y se mantienen actualizadas en materia de seguridad hasta 18 meses después de su lanzamiento.

Todos los lanzamientos de Ubuntu se proporcionan sin coste alguno con varias alternativas sobre el tipo de instalación.

1) El **CD de escritorio** (comúnmente llamado live CD), permite probar Ubuntu sin hacer cambios en el equipo y agregar una opción para instalarlo más tarde.

2) El **CD de instalación de servidor** instala Ubuntu permanentemente en una computadora usada como servidor (no se instalará una interfaz gráfica de usuario). Se pueden descargar desde los sitios oficiales.

Ubuntu posee una gran colección de aplicaciones sencillas de instalar y configurar desde su entorno de escritorio oficial Gnome. UTF-8 es la codificación de caracteres en forma predeterminada desde la versión 5.04. Ubuntu es la distribución GNU/Linux elegida por la UOC porque es un sistema operativo libre con un escritorio potente y amigable.

3.1. Requisitos del sistema para instalar la versión 8.10 Intrepid Ibex

Se recomienda:

- Procesador: 1 GHz
- Memoria RAM: 512 MB (1 GB mínimo)
- Disco duro: 10 GB mínimo

Instalación a partir de live CD (recomendado) o a partir de Windows con Wubi (para inexpertos):

- Procesador: 600 MHz
- Memoria RAM: 256 MB (1 GB mínimo)
- Disco duro: 10 GB mínimo

Enlaces recomendados

Enlaces externos:

Sitio web oficial de Ubuntu (en inglés)
Comunidad hispana de Ubuntu (en castellano)

Wubi

Wubi es un instalador de Ubuntu para sistemas operativos Windows que permite a usuarios de Windows, no acostumbrados a Linux, poder probar Ubuntu sin el riesgo de perder información durante un formateo o particionado, además de desinstalar Ubuntu desde Windows.

3.2. Instalación del servidor LAMP desde el escritorio de Ubuntu

La arquitectura LAMP se refiere a un conjunto de tecnologías de software necesarias para alcanzar una solución global, en este caso para configurar sitios web o servidores dinámicos. El acrónimo LAMP combina los siguientes programas: Linux (sistema operativo), Apache (servidor web), MySQL (gestor de bases de datos), Perl, PHP o Python (lenguajes de programación).

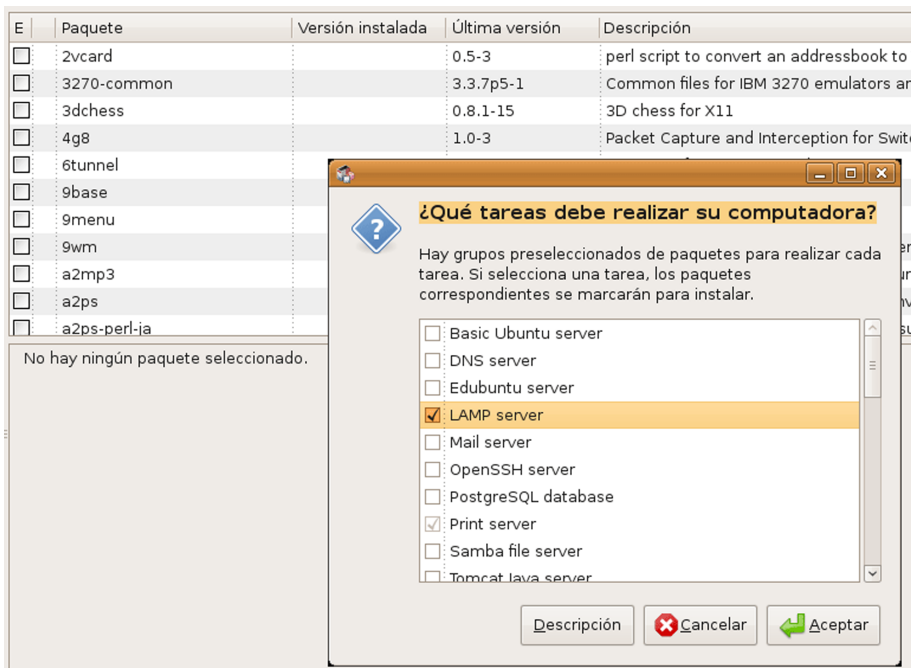
En Ubuntu existen varios modos de añadir, eliminar o actualizar aplicaciones del sistema. Podemos instalar los paquetes desde el CD de instalación o, alternativamente, desde los sitios oficiales de Internet (será necesario tener abierta la conexión a Internet). El gestor de paquetes Synaptic nos permitirá instalar, reinstalar y eliminar paquetes de una manera gráfica muy sencilla.

El proceso de instalación de Apache + Mysql + PHP es rápido y sencillo.

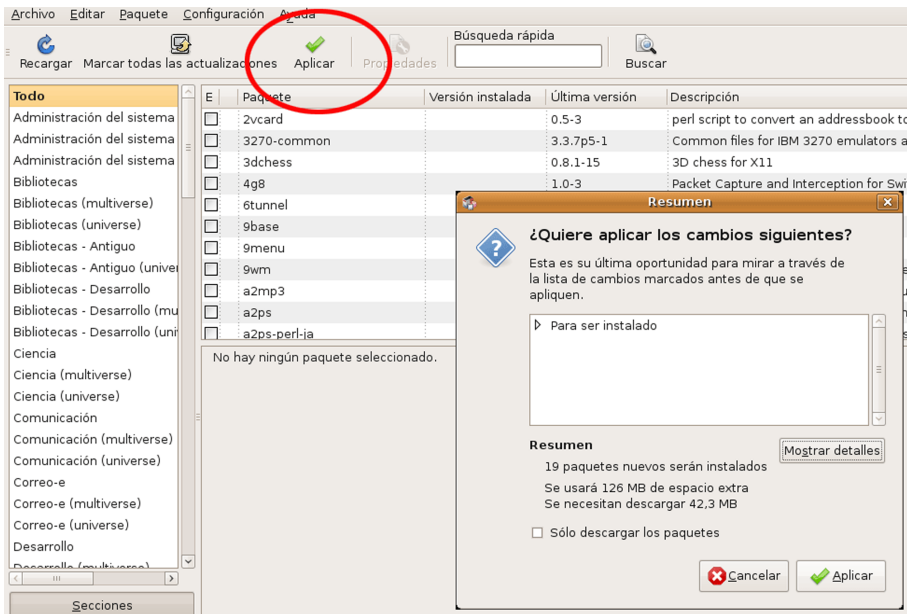
1) Abrimos el Gestor de Paquetes Synaptic desde Menú de Ubuntu: Sistema → Administración → Gestor de Paquetes Synaptic



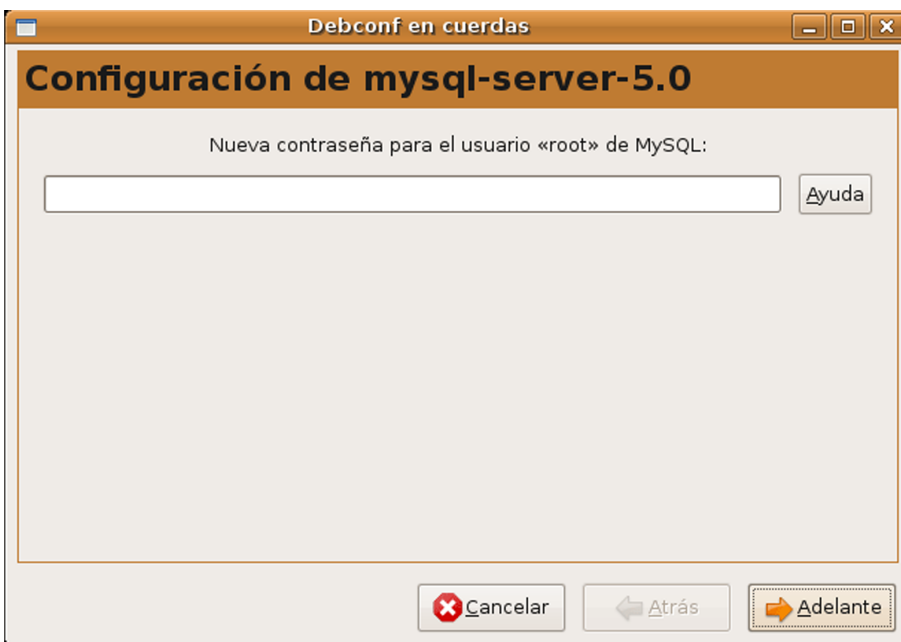
2) En Gestor de Paquetes Synaptic abrimos el menú “Editar” y ejecutamos la opción “Marcar paquetes por tarea”. Se abrirá una ventana que nos pregunta “¿Qué tarea debe realizar su computadora?”; marcamos la casilla “Lamp server”.



3) Tras darle al botón “Aceptar”, se nos informa de todos los paquetes necesarios para instalar el servidor LAMP. Confirmamos la instalación pulsando el botón “Marcar”. Después hacemos clic en el botón “Aplicar” ubicado en el menú para que comience la instalación.



4) Una vez terminado el proceso de descarga de los paquetes comienza la instalación. En la configuración de MySQL 5.0 se nos pregunta qué contraseña queremos asignarle al administrador (root) del servidor de la base de datos. Tras este paso, concluye la instalación del Servidor Lamp con Synaptic.

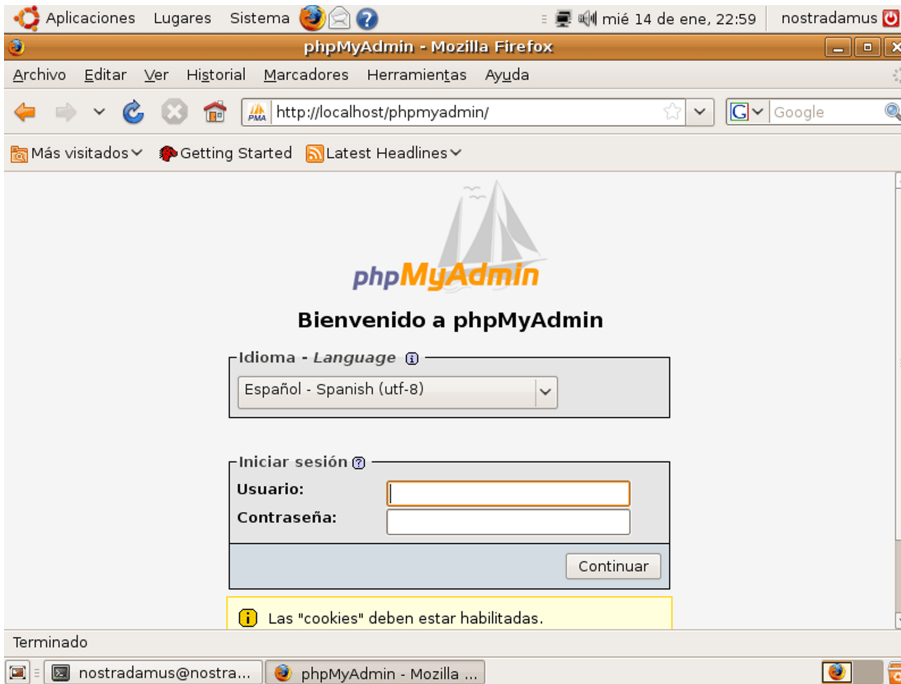


5) Abrimos Firefox y escribimos `http://localhost` (alternativamente, `http://127.0.0.1`) y nos deberá aparecer la página de bienvenida de Apache. Recordad que el directorio raíz del servidor HTTP se encuentra en `/var/www`, y allí es donde se deberán colocar todos los archivos del sitio web.

También es recomendable instalar con Synaptic el paquete `phpmyadmin` para administrar MySQL y sus bases de datos por medio de la interfaz web de esta aplicación. Tras marcar el paquete e instalarlo, bastará con escribir en el navegador:

```
http://localhost/phpmyadmin/
```

La página de inicio de phpMyAdmin nos pedirá un nombre de usuario (`root`) y la contraseña que definimos en su momento al instalar MySQL (paso 4).



3.3. Instalación del servidor LAMP desde la consola de Ubuntu

La consola de Ubuntu (también conocida como intérprete de comandos, terminal o consola) es un programa informático que recibe y procesa órdenes de texto y tiene como función actuar como interfaz entre el usuario y el sistema operativo.

En este apartado vamos a ver cómo instalar el servidor LAMP desde la terminal de Ubuntu. Para labores de administración en terminal, Ubuntu incluye un comando llamado `sudo` con el que se evita el uso del usuario `root` (administrador de Linux). Esta instalación sirve tanto para Ubuntu como para Debian (los usuarios de Debian deberán identificarse como `root` y no utilizar el `sudo` en los comandos).

Los paquetes de software son accesibles a partir de la lista de repositorios ubicada normalmente en `/etc/apt/source.list`. Si quisiéramos actualizar el índice de paquetes sincronizándolo con las últimas versiones disponibles de cada uno, escribiríamos:

```
> sudo apt-get update
```

A continuación vamos a instalar los paquetes necesarios para montar un servidor LAMP desde la terminal de Ubuntu. Abrimos la terminal (Aplicaciones -> Accesorios -> Terminal).

1) Para instalar Apache 2.0 escribimos el siguiente comando y pulsamos “Enter”:

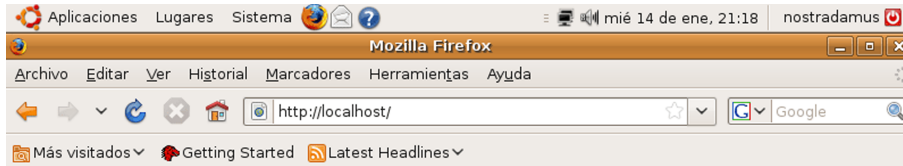
```
> sudo apt-get install apache2
```

La terminal nos pedirá la contraseña de root (del sistema operativo); la introducimos y pulsamos de nuevo “Enter”.

Los archivos de configuración de apache están ubicados en `/etc/apache2/apache2.conf`.

La carpeta raíz de la web se encuentra en `/var/www/`.

Para comprobar que el servidor web está funcionando, abrimos nuestro navegador web (Firefox en Ubuntu) y escribimos `http://localhost/`; aparecerá la página de bienvenida de Apache.



It works!

2) A continuación instalaremos los paquetes PHP5 y el módulo php5 de Apache 2.0. Abrimos de nuevo la terminal de Ubuntu y escribimos:

```
> sudo apt-get install php5 libapache2-mod-php5
```

Tras la instalación de PHP, reiniciamos Apache con el comando:

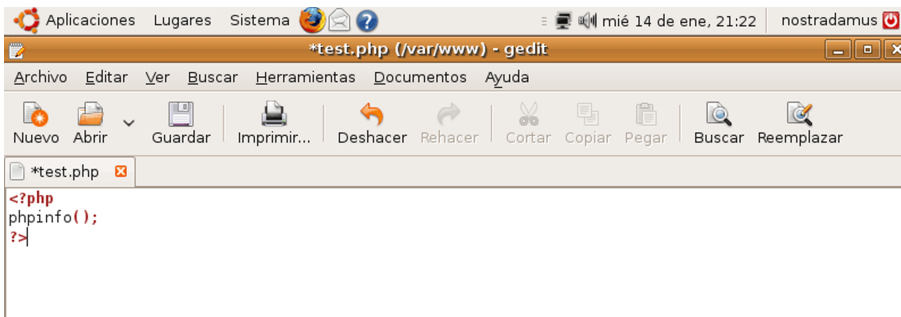
```
> sudo /etc/init.d/apache2 restart
```

Para comprobar que PHP funciona correctamente como módulo de Apache, creamos un archivo de texto y dentro de ese archivo escribimos `<?php phpinfo(); ?>`

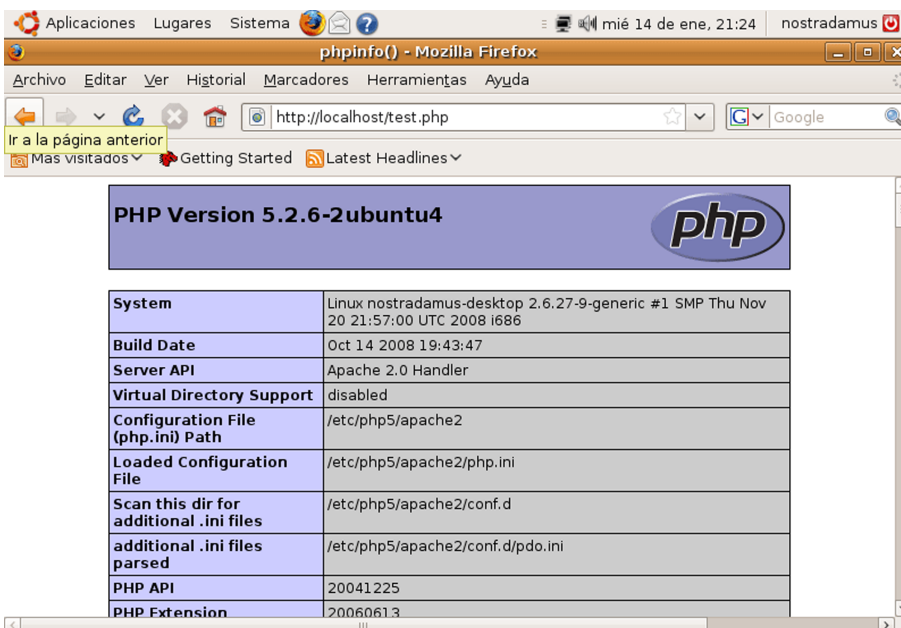
```
> sudo nano /var/www/test.php
```

o alternativamente

```
> sudo gedit /var/www/test.php
```



Luego abrimos el navegador web y escribimos `http://localhost/test.php`; deberá aparecer la pantalla con toda la configuración de PHP.

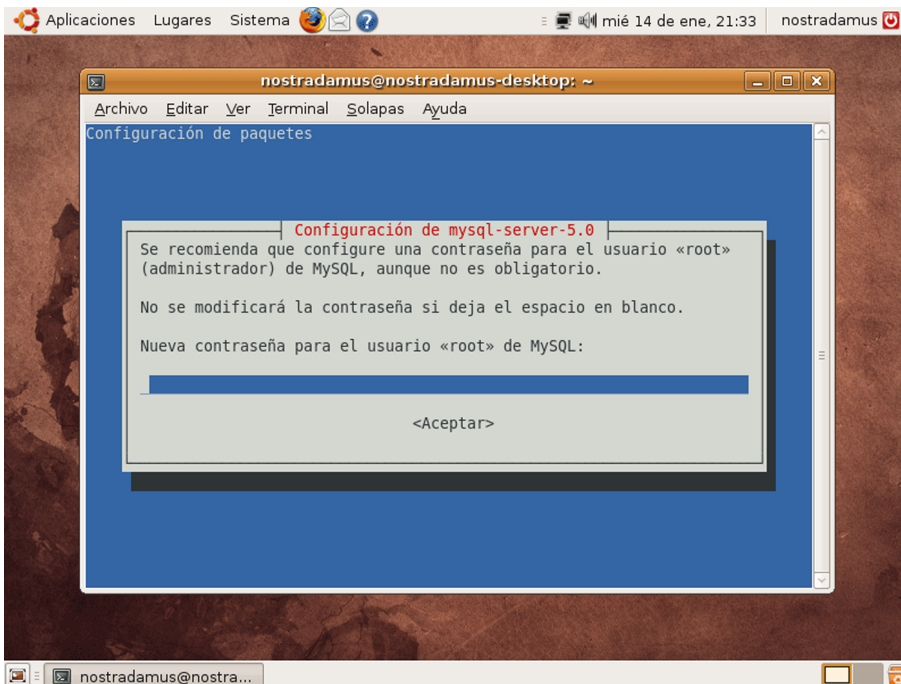


El archivo de configuración de PHP se encuentra en `/etc/php5/apache2/php.ini` (cada vez que lo modifiquemos habrá que reiniciar el servidor http para que los cambios tengan efecto).

3) Instalaremos ahora el servidor MySQL 5, el programa cliente `mysql5`, y el módulo `mysql` para `php5`. Los tres paquetes que se deben instalar se indican con el siguiente orden:

```
> apt-get install mysql-server mysql-client php5-mysql
```


4) Al finalizar el proceso de instalación se nos pedirá introducir la contraseña de root del administrador del servidor MySQL (introducimos la nueva contraseña y la confirmamos).



Si no hemos introducido la contraseña de root, lo podremos hacer más adelante, cuando lo creamos necesario. Por defecto, MySQL crea un usuario root sin ninguna contraseña. Podemos añadir la nueva contraseña de administrador conectándonos al servidor de base de datos con el programa cliente mysql, y, una vez dentro de la terminal de mysql, la cambiamos.

```
> mysql -u root
```

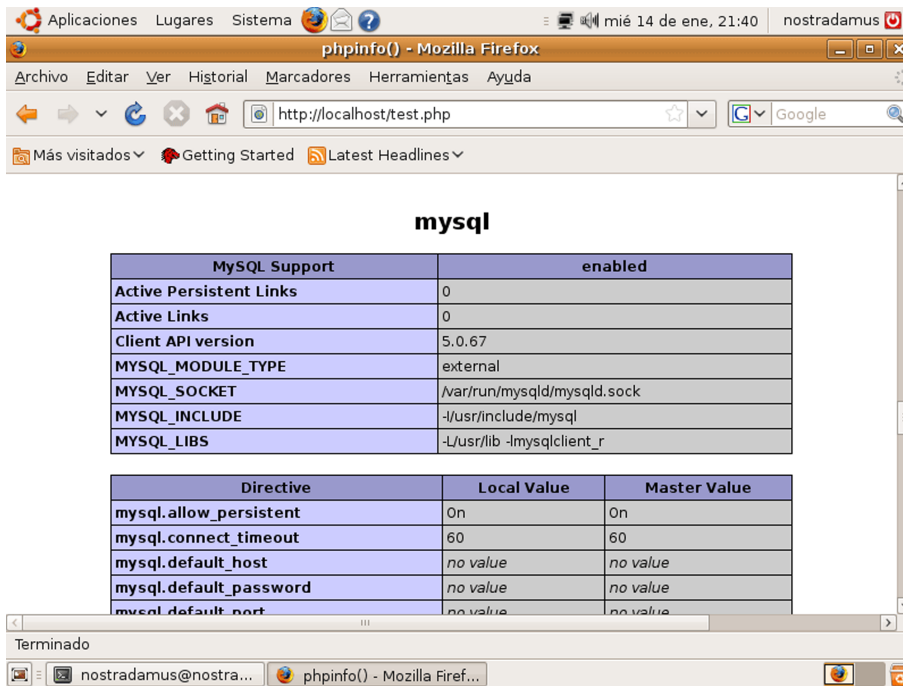
En la terminal de mysql escribimos:

```
mysql> USE mysql;
mysql> UPDATE user SET Password=PASSWORD('nuevo-password')
WHERE user='root';
mysql> FLUSH PRIVILEGES;
```

5) La instalación del paquete php5-mysql (paso 3) hace accesible MySQL desde Apache con la API de PHP, pero será necesario reiniciar Apache para que este cambio tenga efecto.

```
> sudo /etc/init.d/apache2 restart
```

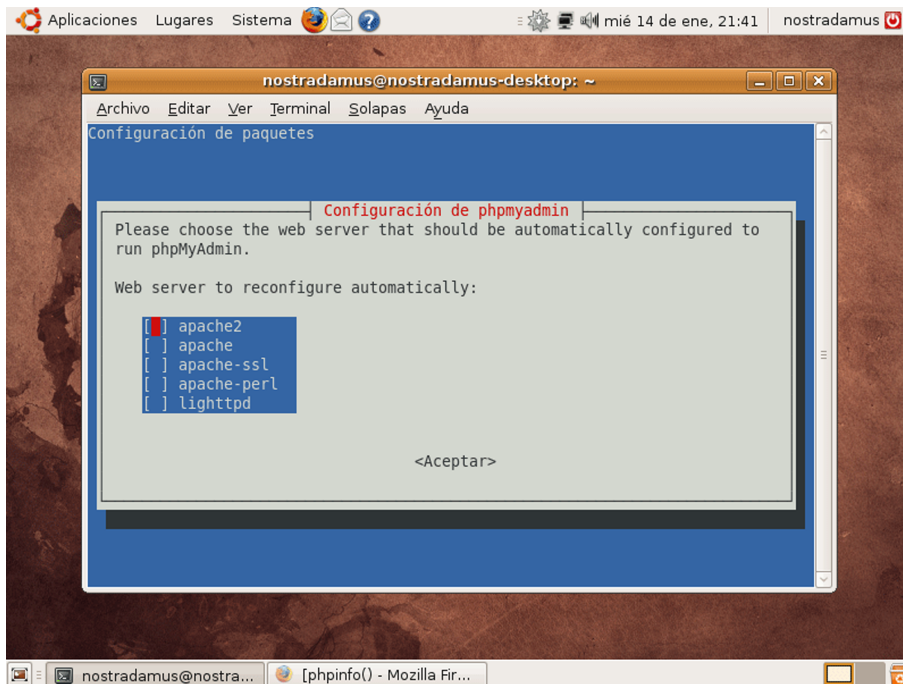
Volvemos a abrir la página de configuración de PHP con el navegador y comprobamos que esté disponible el módulo Mysql de PHP en Apache.



6) Opcionalmente, vamos a instalar el paquete de PhpMyAdmin desde la consola. Para ello, escribimos:

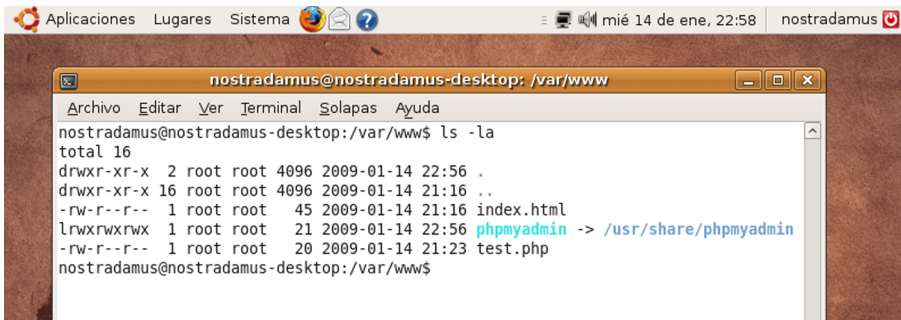
```
> sudo apt-get install phpmyadmin
```

Al final de la instalación marcamos la casilla apache2 (Apache 2.0).



Después de instalar PhpMyAdmin desde los repositorios, hay que hacer un enlace simbólico desde `/usr/share/phpmyadmin` hacia `/var/www/phpmyadmin` para tener acceso a la aplicación web.

```
> sudo ln -s /usr/share/phpmyadmin /var/www
```



```
nostradamus@nostradamus-desktop: /var/www
Archivo Editar Ver Terminal Solapas Ayuda
nostradamus@nostradamus-desktop:/var/www$ ls -la
total 16
drwxr-xr-x  2 root root 4096 2009-01-14 22:56 .
drwxr-xr-x 16 root root 4096 2009-01-14 21:16 ..
-rw-r--r--  1 root root  45 2009-01-14 21:16 index.html
lrwxrwxrwx  1 root root  21 2009-01-14 22:56 phpmyadmin -> /usr/share/phpmyadmin
-rw-r--r--  1 root root  20 2009-01-14 21:23 test.php
nostradamus@nostradamus-desktop:/var/www$
```

El archivo de configuración de PhpMyAdmin se encuentra en el directorio:
`/etc/phpmyadmin`

Abrimos PhpMyAdmin con Firefox e introducimos las claves del administrador de MySQL para acceder a la aplicación.

