

Fundamentos

1

Este capítulo se publica bajo licencia Creative Commons 2.5 Reconocimiento-NoComercial-SinObraDerivada 2.5 Spain



1.1. EL SOFTWARE LIBRE

En numerosas ocasiones se plantea la discusión entre partidarios y detractores del software libre, que frecuentemente se deriva si se refiere a discusión a cuestiones sociológicas y tecnológicas, incluso políticas, lejanas al usuario de a pie. No es el objetivo de este texto hacer una defensa encarnizada del software libre, que evidentemente tiene numerosas ventajas, como el lector podrá descubrir, ni un ataque furibundo contra el software propietario, en numerosas ocasiones inferior tecnológicamente al libre, pero más extendido y cercano al usuario final. Evidentemente, la decisión de utilizar un tipo u otro de herramienta dependerá de numerosos factores, de un análisis detallado del entorno de trabajo y de las posibilidades formativas y tecnológicas disponibles. Por ejemplo, en varios capítulos podrá ver cómo aplicaciones de software libre se instalan y administran sobre plataformas propietarias. No se pretende con ello más que reflejar una situación muy común en muchas unidades y servicios de información, como es la convivencia de ambos tipos de software.

Desde del punto de vista del usuario final, en estos momentos no hay ninguna razón que justifique la utilización de software propietario, excepto en casos puntuales y específicos. La mayor parte de las herramientas que necesita un usuario final para su trabajo ofimático, o de diseño gráfico, o de programación y desarrollo de aplicaciones existen dentro del ámbito de software libre, y en gran parte de las ocasiones son tan buenas o mejores que las herramientas propietarias que ofrecen similares prestaciones (baste pensar en *OpenOffice*, *Eclipse*, *Firefox*, *Thunderbird*, *Scribus*, *Gimp*, *Inkscape*, *MySQL*, *PostgreSQL*, *Samba* y un largo etcétera). Además, la mayoría de estas herramientas tienen versiones para sistemas operativos propietarios, como Windows o Mac OS X. Sí que es cierto, por contra, que este tipo de herramientas han llevado la etiqueta de complejas de instalar o de configurar, quizá por su vinculación original a plataformas GNU/Linux. Sin embargo, en el momento actual la mayoría de ellas ofrecen procesos de instalación automatizados similares a herramientas comerciales, que minimizan este problema.

Donde están mostrando todo su potencial las herramientas de software libre es en la creación de servicios de información para el web. Internet es un campo abonado en el que se implantan soluciones libres para dar soporte a las actividades y tareas que llevan a cabo los usuarios de todo tipo. Puede estar seguro de que el 70 o el 80 por ciento de los servidores de información que utiliza están soportados, total o parcialmente, por herramientas de software libre. Por ejemplo, en el campo de los servidores web, *Apache* es el más extendido, y continúa creciendo su implantación anualmente, a costa del descenso de servidores propietarios. *Apache*, que será ampliamente citado y usado en este libro, es un producto de software libre, desarrollado y soportado por la *Apache Foundation*. La afamada *Wikipedia*, la enciclopedia libre, está soportada por el software *Mediawiki*, desarrollado y mantenido por la propia *Wikipedia Foundation*, y también es software libre.

El software libre no es una moda actual. En realidad, hasta finales de la década de 1970, el software generalmente se distribuía libremente entre universidades, empresas u otras organizaciones, precisamente en virtud de la dificultad de desarrollo, ejecución y actualización del mismo, que hacían aconsejable el trabajo en colaboración entre los usuarios. En la década de 1980, la generalización de la disponibilidad de computadoras permitió que las empresas de software comenzasen a introducir licencias que restringían las posibilidades de copia, uso y modificación del mismo. A mediados de esta década, Richard Stallman comenzó el desarrollo del proyecto GNU, fundó la *Free Software Foundation* (Figura 1.1), y propuso la licencia GNU GPL, o *General Public License* para el software. En conjunto el software libre (expresión que ha sido traducida al español de la anglosajona *free software*) se caracteriza porque debe cumplir cuatro condiciones básicas:

- Libertad 0: poder ejecutar el programa con cualquier propósito.
- Libertad 1: poder estudiar y modificar el programa.
- Libertad 2: poder copiar el programa libremente, incluyendo su redistribución.
- Libertad 3: poder mejorar el programa, e incorporar las mejoras a nuevas versiones.

Evidentemente, estas libertades suponen que es necesario poder acceder, sin cortapisas, al código fuente del programa o aplicación. El lector debe tener en cuenta que un par de años después nace otro enfoque en el mundo del software libre, que desde entonces convive con la FSF y la filosofía GNU. Se trata del movimiento *Open Source Initiative*, que pone el énfasis en la libertad de acceso al código fuente, pero admite que un software pueda ser público, semipúblico o incluso propietario, dando prioridad a los aspectos del desarrollo técnico de las aplicaciones, antes que a los principios de libertad establecidos por la FSF. El lector debe tener en cuenta que, con este enfoque, software abierto no tiene necesariamente que ser sinónimo de software gratuito. Ambos enfoques conviven y colaboran en el desarrollo de proyectos y productos.

En consonancia con lo anterior, se han desarrollado gran cantidad de licencias, que establecen las formas en las cuales se distribuyen los programas y su código, y cuya revisión sobrepasa ampliamente los objetivos de esta obra. Cuando vaya a instalar alguna de las soluciones expuestas en este libro, debe leer y comprobar la licencia que regula su uso. Para el lector es interesante consultar los web de referencia indicados en la Tabla 1.1, que contiene los URL de las principales fuentes de información sobre la cuestión.

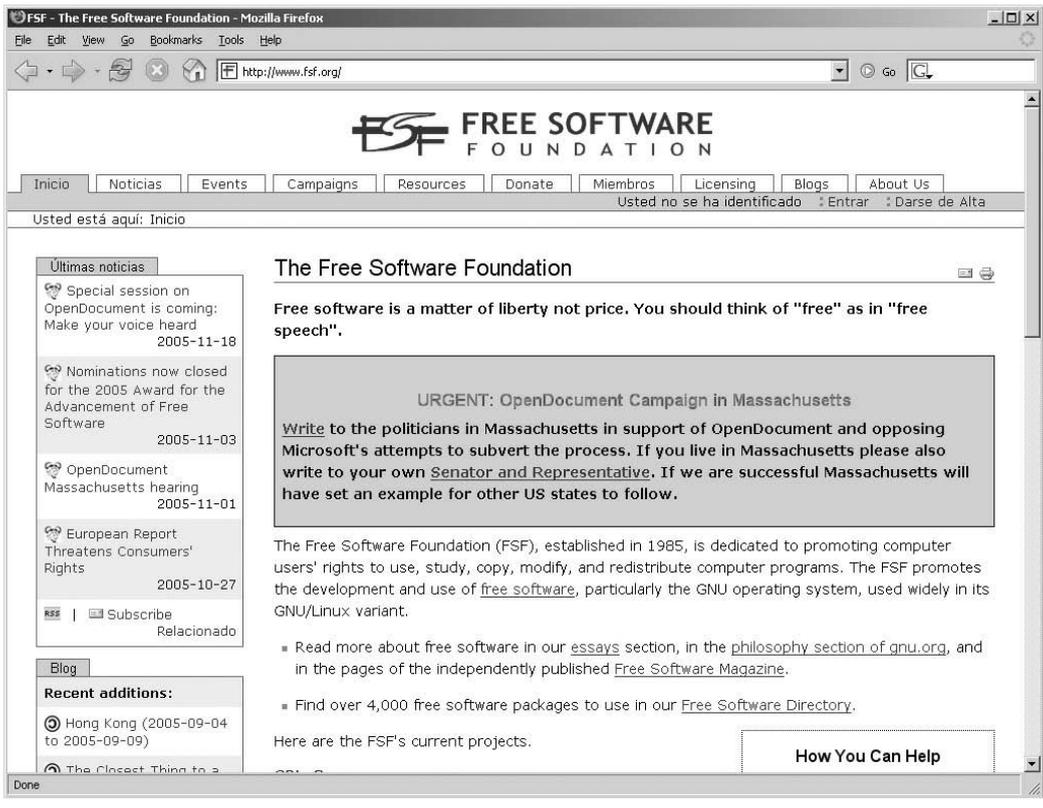


Figura 1.1. Free Software Foundation.

En relación con las licencias y con todo el universo del software libre, en numerosas ocasiones se plantea el problema de la responsabilidad del desarrollo, del mantenimiento del mismo, y del soporte a usuarios que el software libre ofrece. El estudio de los modelos de desarrollo del software libre es un tema sobre el que se ha escrito abundantemente, tanto a favor como en contra. La experiencia dice que, pese a todos los empeños e insistencia de las empresas de software propietario, los diferentes modelos de desarrollo de software libre que pueden encontrarse (grupos de desarrolladores independientes, fundaciones, empresas especializadas...) funcionan, y mantienen y documentan productos de muy alta calidad, como es el caso paradigmático de MySQL AB. Como contrapartida, es un ejercicio interesante recordar el elevado número de productos de software propietario que han desaparecido desde la década de 1980, dejando a sus usuarios sin evolución y sin soporte, y obligándoles a hacer nuevas inversiones para mantener su actividad (Figura 1.2).

Tabla 1.1. Movimientos del software libre

Free Software Foundation	http://www.fsf.org
Open Source Initiative	http://www.opensource.org

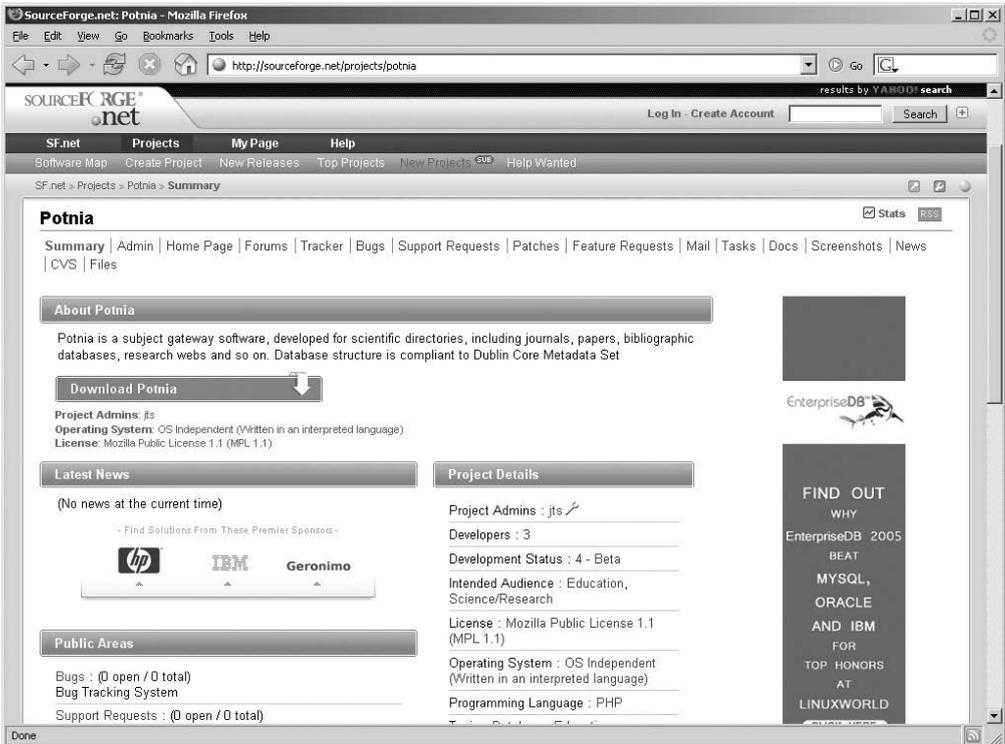


Figura 1.2. Un proyecto en Sourceforge.

Los programas de software libre se cuentan por miles. Para cada actividad que necesite llevar a cabo, existirá una herramienta libre, o los elementos necesarios para que pueda construir una nueva. Para poder localizar las herramientas, el usuario dispone de varios métodos de búsqueda de información, empezando por los motores de búsqueda clásicos. Sin embargo, lo más acertado es acudir a los servidores de referencia que reúnen los proyectos de software libre, como *Sourceforge* o *Freshmeat*. Éstos organizan el acceso a la información a través de *software maps*, o mapas de software, que ofrecen una clasificación temática (tipo de programa, área de aplicación, etc.), mediante la cual se pueden localizar los programas que pueden ajustarse a la necesidad planteada. La información de cada proyecto incluye la página web principal del mismo, el acceso al área de descarga (incluyendo las versiones anteriores), foros de usuarios, avisos, etc. La descarga de software libre desde *Sourceforge* es muy común cuando se trabaja con software libre.

Tabla 1.2. Servidores de proyectos de software libre

Sourceforge	http://sourceforge.net
Freshmeat	http://freshmeat.net/
BerliOS	http://www.berlios.de/index.php.es
Savannah	http://savannah.gnu.org http://savannah.nongnu.org

1.2. GNU/LINUX

El movimiento del software libre tiene como plataforma el sistema operativo abierto y libre GNU/Linux. Es ya muy conocida la historia de Linus Torvalds, y cómo creó un núcleo o kernel de sistema operativo, a comienzos de la década de 1990. La evolución acelerada de ese núcleo, gracias a la colaboración mundial de otros desarrolladores, que utilizaban, y continúan utilizando, hasta límites insospechados el potencial que les ofrece Internet, permitió ir completando progresivamente, en un tiempo récord, no un kernel, sino un sistema operativo completo, en el cual se han integrado la mayoría de las herramientas que han ido siendo desarrolladas por el proyecto GNU de la FSF. De ahí, que la forma correcta de denominarlo sea GNU/Linux, aunque la popularización del mismo haya hecho que comúnmente se use Linux como nombre genérico.

No debe obviarse la referencia a los sistemas de la familia BSD (*Berkeley Software Distribution*). En la década de 1980, un grupo de investigación en informática de la Universidad de Berkeley comenzó a desarrollar un sistema UNIX completamente nuevo, para evitar los problemas de licencias que estaba provocando la explotación comercial de UNIX. El nuevo código y las aplicaciones resultantes eran de muy alta calidad, incluso mejores que las comerciales. Las sucesivas evoluciones de la distribución han culminado con FreeBSD, que puede descargarse e instalarse libremente. BSD ha sido el núcleo sobre el cual se ha desarrollado Mac OS X.

En la actualidad, GNU/Linux es un entorno informático completo, que da soporte a todo lo que el usuario quiera hacer con una computadora. Cuando el usuario instala GNU/Linux, no instala sólo un sistema operativo: las instalaciones incorporan cientos de programas de utilidades del sistema operativos, edición de textos, tratamiento de imágenes y gráficos, Internet y comunicaciones... incluyendo varios entornos gráficos de usuarios basados en ventanas, para que el usuario pueda elegir libremente su entorno de trabajo, como son KDE y Gnome.

La riqueza de GNU/Linux, y la posibilidad de ajustarlo a muy diferentes necesidades y gustos, ha favorecido la aparición de las llamadas *distribuciones* (o distros). Cuando el usuario oye hablar sobre Debian, Ubuntu, Fedora, Red Hat, Suse, Mandriva, Gentoo, Slackware... se está haciendo referencia a diferentes distribuciones de GNU/Linux. Entre ellas, el usuario puede encontrar variaciones, atendiendo a la versión de kernel que utilizan, al entorno gráfico predefinido, a la selección de unas herramientas u otras, al sistema de instalación y mantenimiento de paquetes de software... En principio, la riqueza de distribuciones permite que cada usuario pueda encontrar la que más se ajuste a sus gustos. Además, para el usuario final no supone ningún problema cambiar de una a otra, ya que la compatibilidad de los ficheros y de su contenido está asegurada (lo que no se puede asegurar muy a menudo en sistemas propietarios). Esto es posible gracias a que todas las distribuciones respetan un estándar común que hace posible instalar las aplicaciones sin problemas en todas ellas, lo que asegura que OpenOffice, Gimp o Firefox, por ejemplo, funcionen en todas de la misma forma y sin complicaciones.

Las distribuciones GNU/Linux ofrecen al usuario la posibilidad de mantener el sistema completamente actualizado. Para ello, las aplicaciones y programas se organizan en lo que se llama “paquetes”. Las herramientas de instalación y actualización de las distribu-

Tabla 1.3. Distribuciones GNU/Linux

Debian	http://www.debian.org
Fedora	http://fedora.redhat.com
Ubuntu	http://www.ubuntulinux.org
Mandriva	http://www.mandriva.com
Suse	http://www.opensuse.org
Gentoo	http://www.gentoo.org
Slackware	http://www.slackware.com
MEPIX	http://www.mepis.org
Knoppix	http://www.knoppix.com
FreeBSD	http://www.freebsd.org
DistroWatch (información general sobre distribuciones)	http://www.distrowatch.com

ciones conectan con servidores, llamados “repositorios de paquetes”, en los que se encuentran los mismos, a través de Internet. Cuando detectan una nueva versión del paquete, ofrecen al usuario la posibilidad de actualizar el existente. De la misma forma, son capaces de buscar en los repositorios, paquetes no instalados previamente, para poder aumentar las capacidades del sistema. Los dos formatos de paquete más extendidos son deb (Debian, Ubuntu y otras distribuciones derivadas de Debian) y rpm (Fedora y Red Hat).

Por el contrario de lo que se cree normalmente, no es necesario instalar obligatoriamente GNU/Linux para poder disfrutar de las ventajas del software libre. Muchas de las aplicaciones y herramientas de software libre tienen también versiones plenamente operativas para sistemas propietarios, como por ejemplo OpenOffice. Varios de los capítulos de este libro muestran servicios de información instalados sobre plataformas Windows. Sin embargo, sí que es necesario que el usuario tenga conocimiento, claramente, de que GNU/Linux es muy superior, en prestaciones, capacidad y seguridad, a las máquinas equipadas con Windows, cuando tienen que actuar como servidor de información para Internet.

1.3. LOS SERVIDORES DE INFORMACIÓN

En el segundo capítulo se va a abordar de manera sistemática la planificación y desarrollo de un servicio de información digital. En las secciones anteriores se han delineado brevemente el software libre y GNU/Linux. Es el momento de plantear cuáles son los componentes software de un servidor de información en Internet, y cómo funcionan esos componentes. En primer lugar, es necesario hacer una distinción entre lo que se puede entender como un servidor bajo la responsabilidad del usuario, y lo que son los servicios de alojamiento o *hosting*:

1. Cuando el usuario dispone de una máquina o computadora con acceso completo, del tipo que podemos llamar administrador, con la posibilidad de gestionar cuentas de usuario, instalación y borrado de aplicaciones, ficheros de configuración, etc., estamos tratando con un servidor bajo su responsabilidad. Generalmente se dispone de acceso físico a la máquina, que puede estar instalada en su despacho, hogar u otro lugar de trabajo.
2. Cuando el usuario ha contratado con un proveedor externo un espacio de almacenamiento en un servidor dado, bajo un dominio genérico del propio proveedor, o comprado, y sólo puede acceder a algunas opciones parciales de administración, generalmente a través de un panel de control web, está haciendo uso de servicios de *hosting*. Otra de las características de estos servicios es que el envío de ficheros a la estructura de directorios del proveedor debe hacerse mediante un cliente FTP. Al tratarse de un servicio compartido (ya que el proveedor suele hospedar a numerosos usuarios en el mismo servidor), no se tiene acceso a los ficheros de configuración del servidor web o de los intérpretes de los lenguajes de script.

Por supuesto, puede darse el caso de que el usuario contrate un servicio de *hosting* con servidor dedicado, que ofrezca acceso completo de administrador. Las situaciones pueden ser muy diversas. Como no todos los usuarios pueden disponer de un servidor propio, los siguientes capítulos mostrarán los procesos de instalación en *hosting* y en servidores propios, e incluso en la mayoría de ellos no existe diferencia. Sin embargo, en otras ocasiones el lector podrá comprobar que es necesario un acceso completo para algunos. Lo más recomendable es que se ponga en contacto con el administrador del sistema y le explique la situación.

1.3.1. ARQUITECTURA TÉCNICA DEL SERVIDOR

Crear páginas y documentos web, y subirlos a un servidor para que todo el mundo pueda acceder a ellos no tiene hoy día mucho misterio. Sin embargo, la mayor parte de estas páginas, codificadas en HTML o en XHTML, son estáticas, es decir, no hacen uso de lenguajes de script. No tienen más requerimientos técnicos que la existencia de un software que haga de servidor web, que reciba las peticiones de los navegadores, y las responda enviándoles las páginas que piden. Pero cuando es necesario poner en marcha un servicio de información digital, se debe tener en cuenta que se hace un uso exhaustivo de contenido dinámico, y que hay que dar soporte a procesos de edición, modificación y búsqueda y consulta de contenidos. Esta necesidad está en el origen de lo que actualmente se han llamado sistemas de gestión de contenidos (*content management systems*, CMS). La forma más extendida y sencilla de un CMS es el blog o weblog, o cuaderno de bitácora, que ahora se encuentra en un proceso de expansión sin precedentes.

El funcionamiento básico de los servicios de información disponibles a través de Internet, y en particular del web, muestra un esquema sencillo. Un cliente, que generalmente es un navegador web, hace una petición de información a un servidor, que la recibe, procesa, y remite la respuesta oportuna. En su formulación más sencilla, los clientes web piden, y reciben, una página web estática, es decir, una página que sólo contiene código HTML. Cuando el navegador la recibe, sólo tiene que interpretar la página para mostrar correctamente el contenido (Figura 1.3).

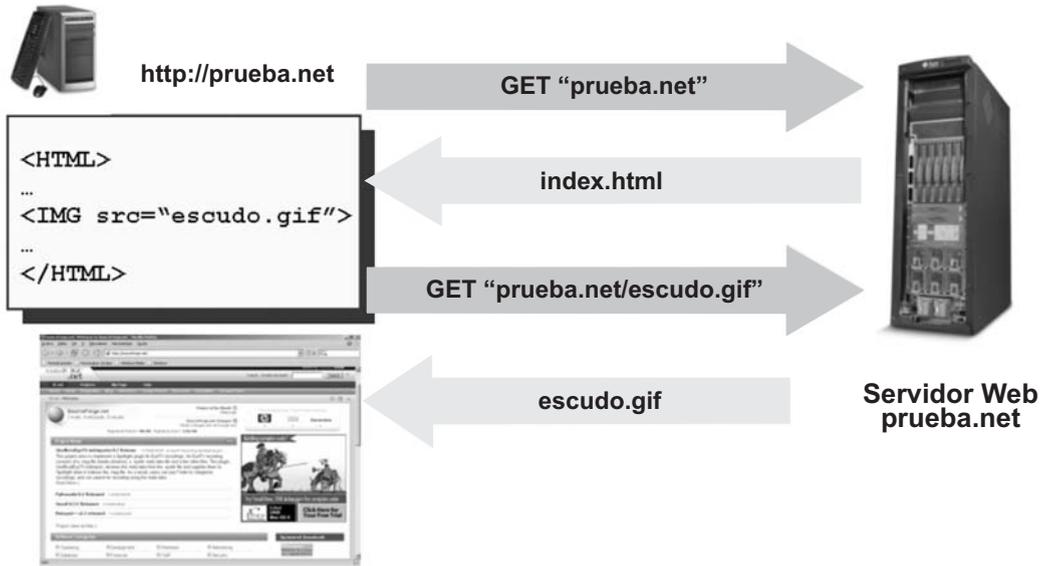


Figura 1.3. Servidor de páginas estáticas.

Las páginas estáticas no sirven para servicios de información que necesitan enviar al usuario contenidos que incluyen informaciones obtenidas de bases de datos. En este caso, el navegador envía una petición al servidor, incluyendo las instrucciones necesarias para ejecutar el código de un lenguaje de script. Este código puede estar incluido en la propia página web, o bien en ficheros existentes en el servidor. Cuando el servidor recibe la petición, pasa el código al intérprete del lenguaje que se trate (Perl, PHP...). Este intérprete lo ejecuta, conecta con el servidor de bases de datos, que le devuelve los datos necesarios, y realiza con ellos las operaciones oportunas. Por último, devuelve al servidor el código necesario para que remita al cliente una página web que éste pueda procesar y presentar al usuario. De esta manera, se forma lo que se han denominado páginas dinámicas (Figura 1.4).

La gestión dinámica de contenidos tiene lugar cuando no sólo se envían contenidos obtenidos de una base de datos, como respuesta a una petición del usuario. Los CMS dan soporte a procesos de introducción, edición y modificación, borrado y consulta de contenidos, así como a control de versiones, gestión de usuarios, formatos de presentación, etc. Esta gestión dinámica y activa de contenidos es posible gracias a la combinación de tres elementos clave, que vamos a ver repetidos en todos y cada uno de los capítulos de este libro:

1. **Servidor web:** el más extendido y recomendable es Apache, con licencia GPL, aunque existen otras opciones. Por seguridad, fiabilidad y robustez, debe ser la primera opción.
2. **Sistema de gestión de bases de datos:** la información se almacena en sistemas de base de datos. La más extendida es MySQL, pero también es una magnífica opción PostgreSQL. La mayoría de aplicaciones incluidas en este libro se integran directamente con MySQL.

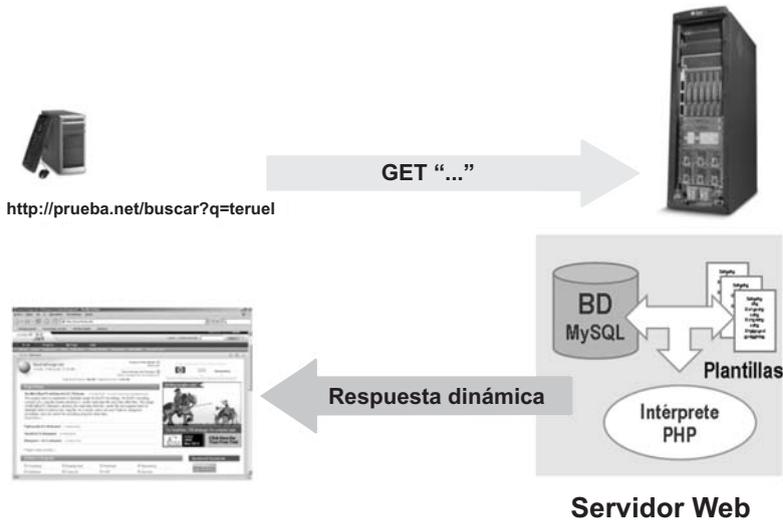


Figura 1.4. Servidor de páginas dinámicas.

Tabla 1.4. Servidores web libres

Apache	http://httpd.apache.org
AOL	Server http://www.aolserver.com
Xitami	http://www.xitami.com
Roxen	http://www.roxen.com
Jigsaw	http://www.w3c.org/Jigsaw

Tabla 1.5. Sistemas de gestión de bases de datos

MySQL	http://www.mysql.com
PostgreSQL	http://www.postgresql.org
Firebird	http://firebird.sourceforge.net
SQLite (librería)	http://www.sqlite.org

Un complemento necesario para los sistemas de gestión de bases de datos es un cliente que facilite la creación, manipulación y gestión de bases de datos, de tipo gráfico, que evite usar, en lo posible, las interfaces de línea de órdenes. El más extendido es **phpMyAdmin**, un cliente web para servidores MySQL. Para poder usarlo, debe tener instalado previamente Apache, PHP y MySQL. Puede instalarlo tras descargar el fichero disponible en <http://www.phpmyadmin.net>. En las distribuciones GNU/Linux suele venir como paquete, y la instalación del mismo ya hace todos los ajustes necesarios para que esté plenamente operativo. Tras la instalación, suele ser suficiente con poner en marcha su navegador, e introducir la URL <http://localhost/phpmyadmin>. phpMyAdmin utiliza los mismos usuarios y las

mismas claves que MySQL. En gran parte de las instalaciones que muestra este libro se usará phpMyAdmin para la creación de bases de datos (Figura 1.5).



Figura 1.5. phpMyAdmin.

3. **Lenguaje de script**, interpretado en el servidor: hay diferentes lenguajes interpretados que pueden integrarse en las propias páginas web, como PHP, o formar scripts independientes, que son llamados desde las páginas web, como Perl.

1.3.2. LENGUAJES DE SCRIPT INTERPRETADOS EN EL SERVIDOR

Los principales lenguajes de script que se usan en las aplicaciones web se recogen en el listado siguiente:

- PERL (*Practical Extraction and Report Language*): lenguaje de programación interpretado, desarrollado por Larry Wall en netlabs. Perl es un lenguaje de script de tipo BCPL (*Basic Combined Programming Language*). Ordenado, potente y muy fácil de adaptar a diferentes arquitecturas. No es ni un compilador ni un intérprete, sino que se encuentra en un punto intermedio. Muy utilizado para la elaboración de aplicaciones CGI, permite el desarrollo rápido de aplicaciones y herramientas especialmente orientadas al tratamiento de textos y archivos, aunque actualmente también se utiliza incluso para entornos gráficos, en combinación con sistemas como Perl/TK o GTK.
- PHP (*Hypertext Preprocessor*): lenguaje de programación interpretado con licencia open source. Su interpretación y ejecución se produce en el lado del servidor, en el que se

encuentra almacenada la página. El cliente sólo recibe el resultado de la ejecución. Permite la conexión a numerosos gestores de bases de datos de forma nativa y tiene la capacidad de ser ejecutado en la gran mayoría de los sistemas operativos.

- PYTHON: creado por Guido Van Rossum, Python es un lenguaje de programación con licencia open source, interpretado e interactivo, capaz de ejecutarse en una gran cantidad de plataformas. Administrado por la *Python Software Foundation*, la última versión estable es la 2.4.2, de septiembre de 2005.
- JAVA: lenguaje de programación orientado a objetos, multiplataforma, desarrollado por Sun Microsystems. Pretende ser un lenguaje totalmente portable entre distintas plataformas, gracias a que no se compila a código máquina, sino a un lenguaje intermedio que luego es interpretado por la “máquina virtual Java”, que sí es específica de cada plataforma.
- RUBY: Ruby es un lenguaje de scripting, interpretado, con una sintaxis inspirada en Eiffel y Ada, y orientado a objetos. Fué creado por Yukihiro Matsumoto en 1993. Se diferencia de otros lenguajes orientado a objetos en que en este lenguaje hasta los caracteres, números y literales son objetos, y por consiguiente disponen de unas clases con una serie de métodos. Es rápido y limpio de programar, y está alcanzado gran auge gracias a su integración con las tecnologías del Web 2.0, mediante el uso de la plataforma de desarrollo *Rails*, en lo que ya se conoce como *Ruby on Rails*.

Tabla 1.6. Principales lenguajes de desarrollo de aplicaciones para Internet

Perl	http://www.perl.com
PHP	http://www.php.net
Python	http://www.python.org
Java	http://java.sun.com
Ruby on Rails	http://www.rubyonrails.com/

I.4. INSTALACIÓN DE LOS COMPONENTES BÁSICOS

La combinación de los componentes servidor web, sistema de gestión de base de datos e intérprete de lenguaje, se identifica con el acrónimo LAMP, correspondiente a Linux/Apache/MySQL/PHP-Perl (en puridad, sólo hace referencia a PHP). Como el lector puede imaginar, WAMP se usará para aquellos en los cuales el sistema operativo es una versión de Windows (todavía no hemos visto un acrónimo MAMP para Macintosh OS X). La instalación de estos componentes puede hacerse de tres maneras principales:

1. El usuario obtiene los paquetes originales de cada uno de los elementos y procede a instalarlos en el sistema, así como a modificar la configuración del mismo donde sea necesario, para que todos ellos puedan conectar entre sí, e intercambiar la información oportuna. Este tipo de instalación suele ser la más larga y compleja, y suele requerir privilegios del administrador del sistema en cuestión, así como conocimientos avanzados.

2. El usuario puede buscar un *bundle*, instalador o paquete de instalación, que suele contener todos los componentes esenciales, y lanza el proceso de instalación. El instalador se encarga de realizar todos los ajustes y configuraciones oportunos, para que al final del proceso el usuario tenga un servidor LAMP/WAMP plenamente operativo. No suele ser necesario modificar la configuración realizada. Ésta es una opción muy común para servidores WAMP, ya que existe un número elevado de instaladores, que ofrecen diferentes opciones.
3. Los sistemas GNU/Linux disponen de gestores de paquetes (rpm y deb), que permiten instalar, en un entorno gráfico, los paquetes necesarios para poner en marcha un servidor. Además, estos gestores realizan instalaciones “inteligentes”, en cuanto se encargan de analizar qué otros paquetes pueden ser necesarios para una plena funcionalidad (a esta acción se le llama “resolver dependencias”). Para poder hacer uso de estas ventajas, puede usar RPM Package Manager o Synaptic. Basta buscar el programa, marcarlo y pedir su instalación. De esta forma podrá instalar Apache con los módulos necesarios, PHP y MySQL rápidamente. Con Synaptic, por ejemplo, los procesos de instalación ajustan automáticamente los ficheros de configuración para que todo funcione correctamente.

Si el usuario ha contratado un proveedor de *hosting*, resultará evidente que la instalación, configuración y actualización de los diferentes componentes será responsabilidad del proveedor. Es importante que, antes de instalar un servicio como los explicados en este libro, compruebe que el proveedor cumple los requerimientos necesarios. Muchos proveedores de *hosting* gratuito no incluyen PHP ni bases de datos en sus cuentas. Las prestaciones de las bases de datos MySQL, o de otro tipo, varían en los proveedores comerciales. La mayoría de los proveedores ofrecen el acceso a sus servidores de bases de datos MySQL a través de phpMyAdmin, por lo que es necesario que se familiarice con el mismo.

1.4.1. INSTALACIÓN EN UNA DISTRIBUCIÓN GNU/LINUX

La puesta en marcha de los tres componentes puede hacerse mediante la instalación individual de cada uno de ellos, lo que exige descargar, descompactar y ejecutar los scripts de instalación correspondientes, y terminar ajustando manualmente los ficheros de configuración. Sin embargo, es mucho más rápido y recomendable, para comenzar, que se instalen los paquetes que incorporan las diferentes distribuciones. Cuando se selecciona un servidor, los gestores de paquetes se encargan de instalar los programas complementarios necesarios, aunque el usuario no los haya seleccionado previamente. En la distribución que tenga instalada, busque en el entorno gráfico (KDE o Gnome) la aplicación encargada de gestionar los paquetes. Le será necesario disponer de una cuenta de administrador, ya que los servidores se instalan como servicios que arrancan automáticamente cuando se pone en marcha el sistema (Figura 1.6).

Como ejemplo, se va a instalar el servidor Apache en una distribución Ubuntu 5.10. En primer lugar, busque el menú **Sistema**, submenú **Administración**, aplicación **Gestor de paquetes Synaptic**.

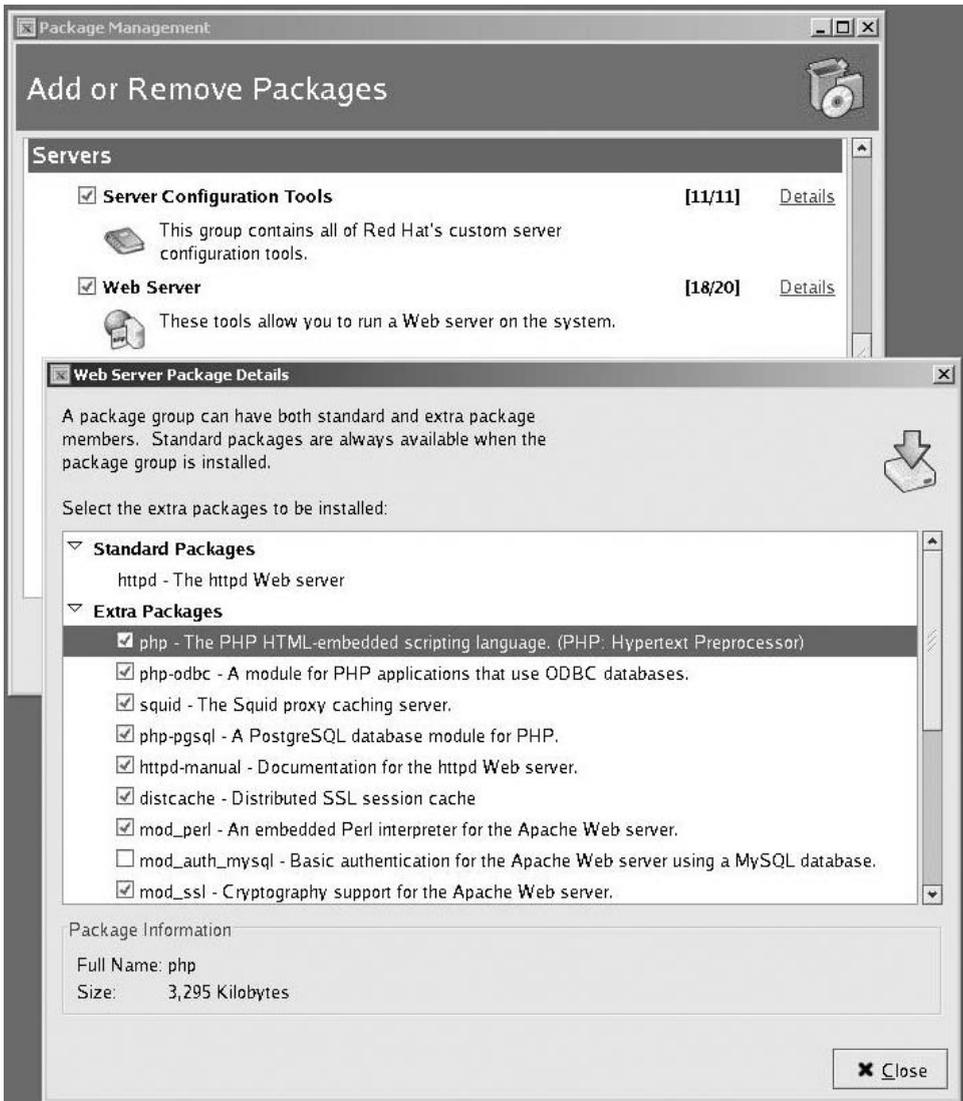


Figura 1.6. Instalación de Apache en KDE sobre RedHat/Fedora.

1. Introduzca la clave de administrador
2. Pulse sobre el botón **Buscar** de Synaptic.
3. En la caja de diálogo, introduzca “apache”. Pulse sobre **Buscar**.
4. Synaptic le mostrará todos los paquetes que contienen Apache en el nombre (en la ventana de Búsqueda también están disponibles otras opciones, véase Figura 1.7).
5. Pulse sobre Apache2. Se activa la instalación de Apache2. A continuación, Synaptic muestra una ventana de diálogo en la que le indica qué otros paquetes debe instalar para asegurar las dependencias, y que el servidor funcione correctamente. Pulse sobre **Marcar** (Figura 1.8).

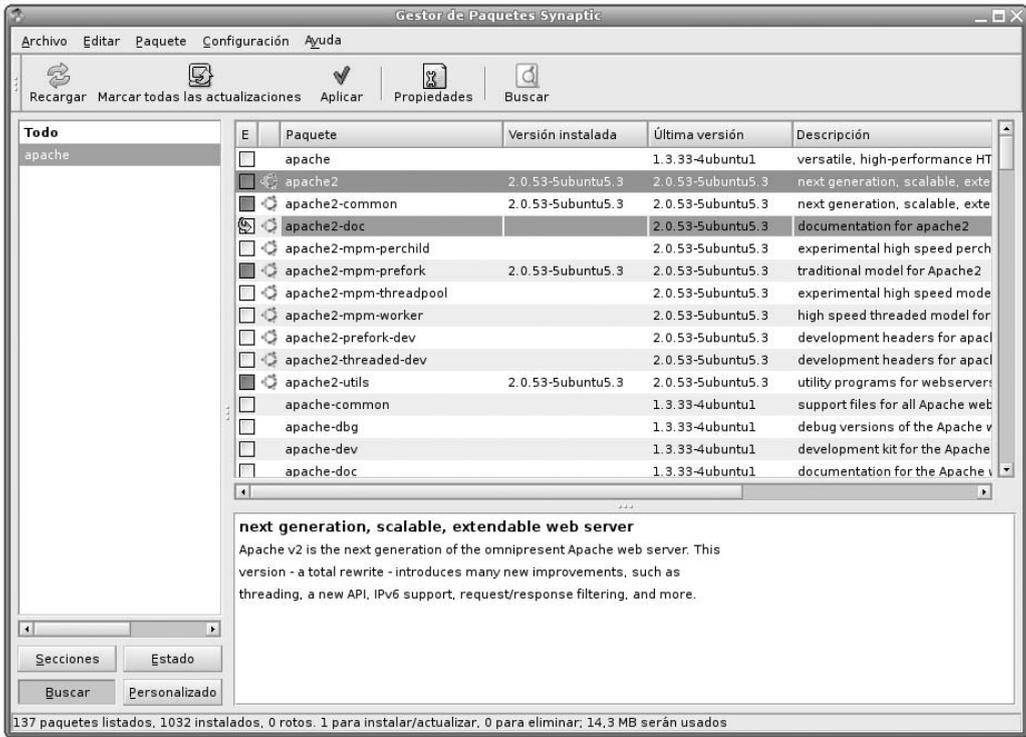


Figura 1.7. Listado de paquetes de Apache2.

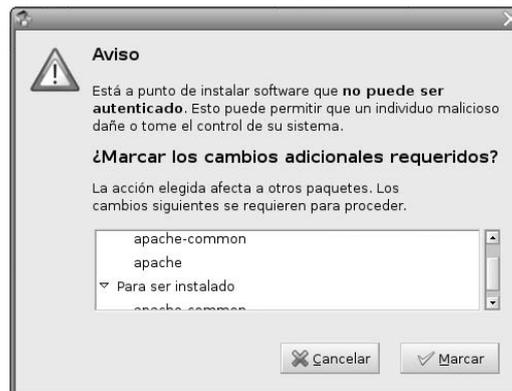


Figura 1.8. Resolución de dependencias.

- De nuevo en la ventana principal de Synaptic, pulse sobre **Aplicar**. Synaptic le pedirá que confirme que desea instalar el conjunto de paquetes seleccionados. Pulse el botón **Aplicar**. Synaptic buscará los paquetes en el CD de instalación de Ubuntu (si está disponible en su ordenador); en caso contrario, intentará descargarlos de un repositorio a través de Internet (Figura 1.9).



Figura 1.9. Ventana de confirmación de instalación de paquetes.

- Una ventana le informa del proceso de descarga e instalación de los diferentes paquetes, y le indicará si se produce algún problema (Figura 1.10).

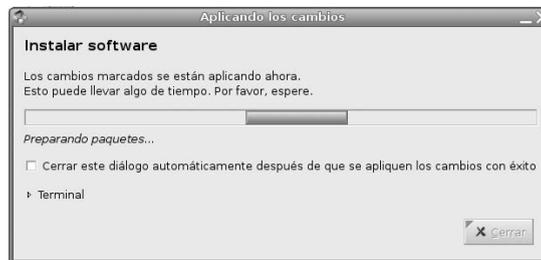


Figura 1.10. Ventana del proceso de instalación.

- Por último, Synaptic le muestra la ventana de confirmación de instalación con éxito. Los paquetes seleccionados de Apache2 ya están instalados, y ya dispone de un servidor web operativo en su máquina (Figura 1.11).

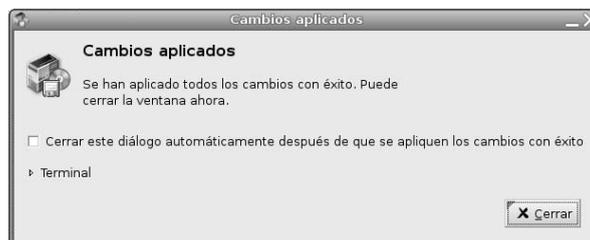


Figura 1.11. Ventana de confirmación del fin de la instalación.

1.4.2. INSTALACIÓN EN UNA PLATAFORMA WINDOWS

XAMPP es un instalador que automatiza el proceso de poner en marcha un servidor con Apache, PHP, Perl, MySQL y phpMyAdmin. XAMPP ofrece versiones del instalador para sistemas GNU/Linux, Solaris, Windows y Mac OS X, así que no tendrá problema en encontrar la suya. En primer lugar, use su navegador para ir al web de referencia, cuya URL es <http://www.apachefriends.org>. También sería más interesante usar el URL <http://www.apachefriends.org/en/xampp.html>, ya que le lleva directamente a las diferentes versiones de XAMPP. Descargue la que corresponda a su sistema. Observe que el conjunto de paquetes que incorpora XAMPP incluyen servidor de comunicaciones seguro (OpenSSL), SQLite, servidor FTP, servidor de correo electrónico, módulos de Apache, librerías para tratamiento de imágenes y de tipos de letra, etc. Los pasos siguientes van a mostrar cómo hacer una instalación en Windows.

1. Para simplificar aún más la instalación, descargue la versión **Installer** del web de referencia. En el momento de redactar estas líneas, la última versión es la 1.5.1. Observe que el enlace le lleva a *Sourceforge*. Elija el *mirror* más cercano a su localización (los mirror europeos suelen ir bastante rápidos, tenga en cuenta que va a descargar 28 megas).
2. Ejecute el fichero que ha descargado en su máquina. En primer lugar, XAMPP le pedirá que indique el lenguaje de instalación, ofreciéndole “Español”. Pulse **Ok**. En la ventana de Bienvenida, pulse **Siguiente** para continuar la instalación (Figura 1.12).



Figura 1.12. Comienzo de la instalación.

3. Introduzca el directorio en el que instalará XAMPP. Observe que no es necesario (y en muchas ocasiones no es recomendable) que lo sitúe dentro del directorio “Archivos de programa”. En este caso, hemos optado por crear un directorio XAMPP. Pulse sobre **Instalar** (Figura 1.13).

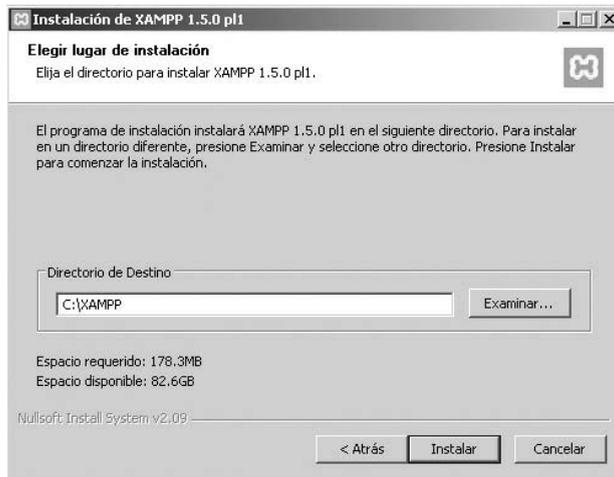


Figura 1.13. Definición del directorio para XAMPP.

4. Espere mientras XAMPP descomprime e instala todos los componentes. Se le muestra una ventana que le informa del desarrollo del proceso. Una vez terminado, una ventana DOS le indica el éxito definitivo del proceso.
5. Por último, y una vez terminado y configurado el servidor, pulse sobre **Terminar** en la ventana de finalización. Preste atención ahora: XAMPP le preguntará si desea instalar los servidores como servicio de Windows. En principio, la respuesta deberá ser afirmativa. De esta forma, el servidor estará siempre operativo, aunque los usuarios no inicien sesión. En caso contrario, el servidor sólo se pondrá en marcha a petición del usuario. Si deseamos disponer de un servidor operativo las 24 horas, pulse sobre **Sí** (Figura 1.14).

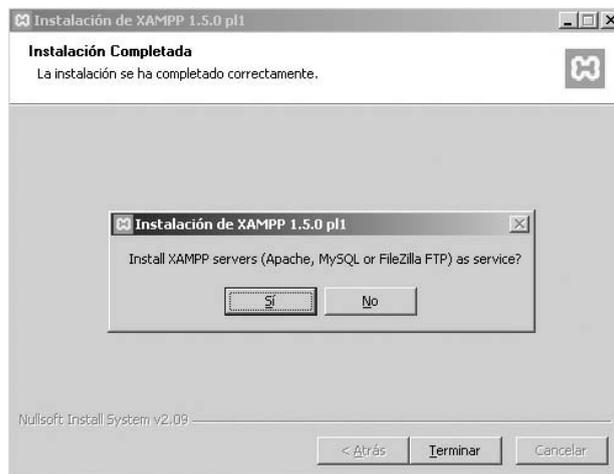


Figura 1.14. Instalación de los servidores como servicio.

6. Vaya respondiendo afirmativamente a todas las preguntas sobre la instalación de Apache y MySQL (debe recordar que para poder instalar los servicios debe disponer de una cuenta del grupo de administradores). La posibilidad de activar el servidor FTP Filezilla dependerá de los servicios que quiera ofrecer al usuario. Una vez terminado el proceso, XAMPP le preguntará si desea poner en marcha el **Panel de Control de XAMPP**. Responda afirmativamente (Figura 1.15).

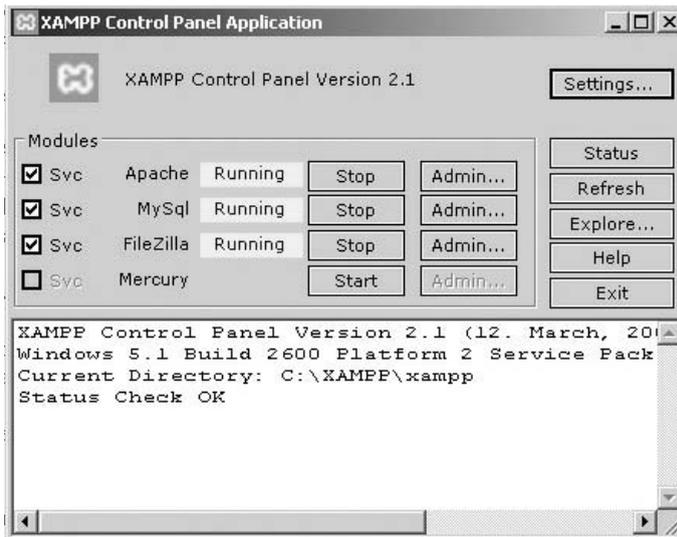


Figura 1.15. Panel de control de XAMPP.

7. El panel de control le indicará qué servicios están funcionando, y le ofrecerá la posibilidad de detenerlos o activarlos. Sin embargo, la instalación, tal y como está ahora, no ofrece los niveles mínimos de seguridad deseables. Para comprobarlo, pulse sobre el botón **Admin**, correspondiente al servicio Apache.
8. Su navegador le mostrará la página principal de su servidor, que le va a ofrecer gran cantidad de información sobre todo lo instalado y su configuración (Figura 1.16).
9. Si pulsa sobre el enlace **Estado**, del menú lateral izquierdo, XAMPP le informará de qué servicios están operativos. Si pulsa sobre **phpinfo()**, XAMP le mostrará la configuración actual de PHP en su ordenador. **Componentes** le indicará qué programas están instalados, y **Documentación** le ofrecerá un listado de URL que le remiten a los web de referencia de los programas instalados.
10. Pulse sobre **Chequeo de seguridad**. XAMPP le informará de todos los problemas de seguridad que ofrece su servidor (podrá comprobar que son bastantes). Para solucionar estos problemas, la propia página le ofrece un enlace al script **xamppsecurity.php** (Figura 1.17).
11. Pulse sobre este enlace. XAMPP le mostrará una nueva página web en la que puede establecer una clave para el usuario root de MySQL (lo cual es imprescindible).

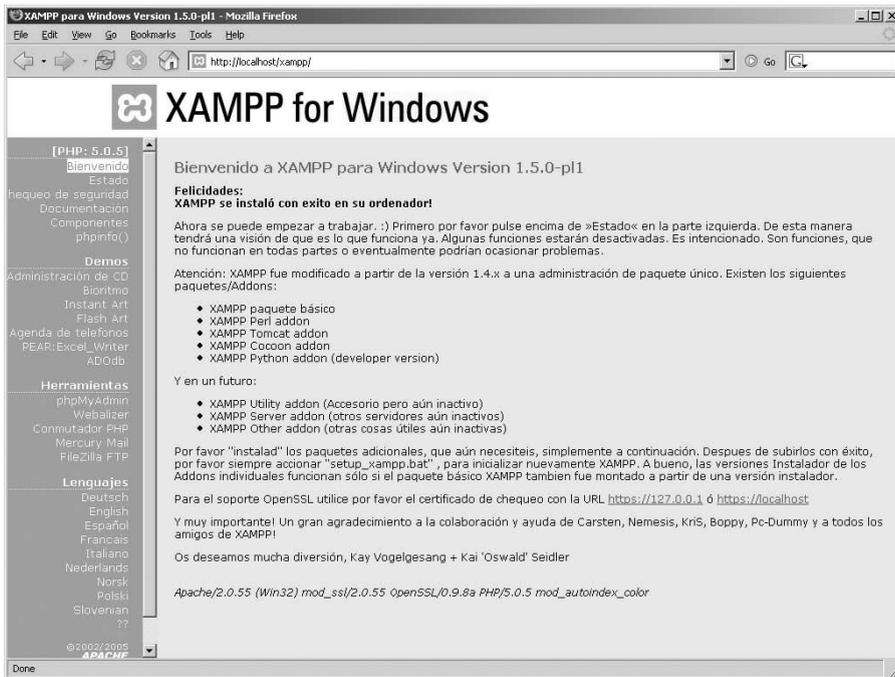


Figura 1.16. Página principal de la instalación de XAMPP.

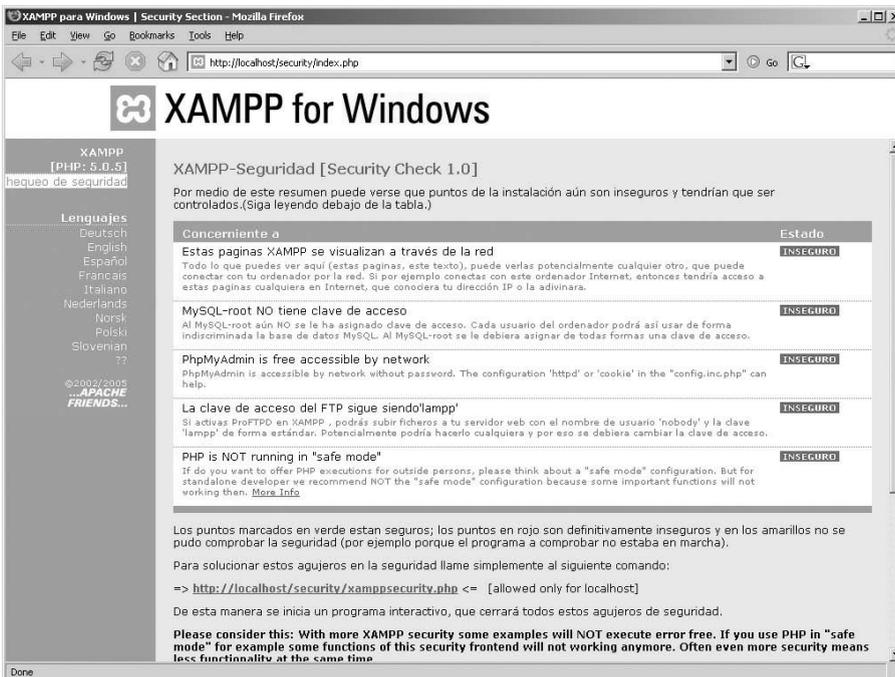


Figura 1.17. Comprobación de seguridad.

ble). Pulse sobre **Password Changing**. Deje sin clave la protección de directorios, ya que ello impediría a los usuario acceder al contenido de todo el servidores, a menos que éstos conozcan la clave de acceso (Figura 18).

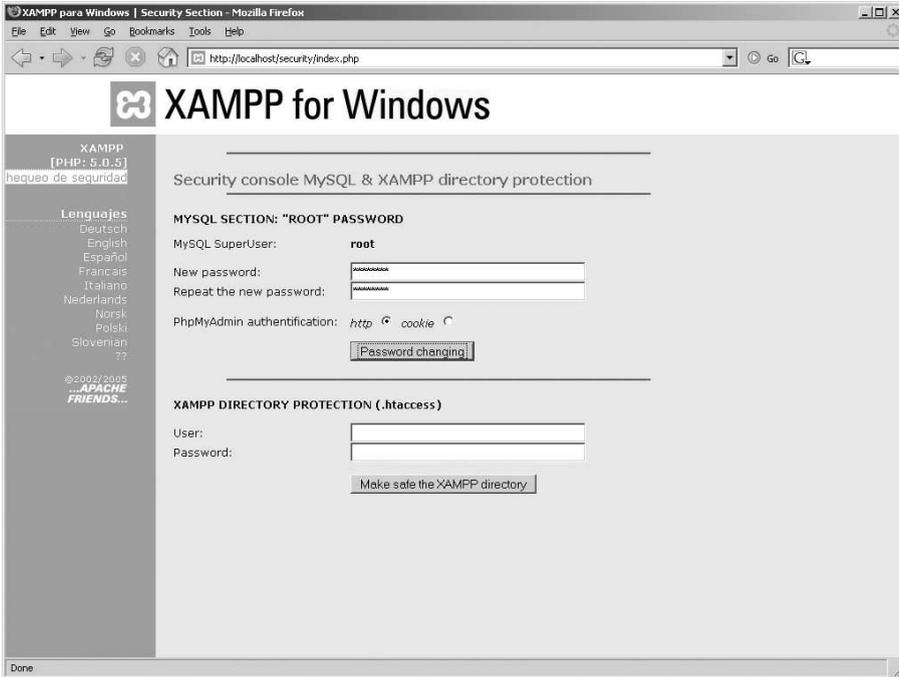


Figura 1.18. Contraseñas para el acceso a MySQL y phpMyAdmin.

12. Reinicie MySQL desde el Panel de control. Recargue la página principal de su servidor XAMPP. Si realiza ahora el **Chequeo de seguridad**, podrá ver que ya no hay problema con MySQL ni con phpMyAdmin, lo que hace a su servidor de bases de datos seguro, ya que sólo se puede acceder al mismo usando el usuario y la clave correspondiente. Si ha activado el servidor FTP, la definición de usuarios y de claves de acceso debe hacerse directamente desde la propia aplicación (en el caso de Windows, pulse el botón **Admin** correspondiente a Filezilla que se ofrece en el Panel de control).
13. De vuelta a la página principal de su instalación XAMPP, podrá ver que el menú lateral le ofrece una serie de enlaces bajo el título **Herramientas**. Las que más le van a interesar son:
 - a) phpMyAdmin: da acceso al phpMyAdmin instalado, que le permitirá gestionar las bases de datos de MySQL. Recuerde que antes se ha asegurado la instalación dando una clave de acceso al usuario root (Figura 1.19).
 - b) Webalizer: uno de los mejores programas de estadísticas de uso para servidores web. Podrá controlar el número de visitas, las páginas servidas, por día, mes, año, etc. (Figura 1.20).

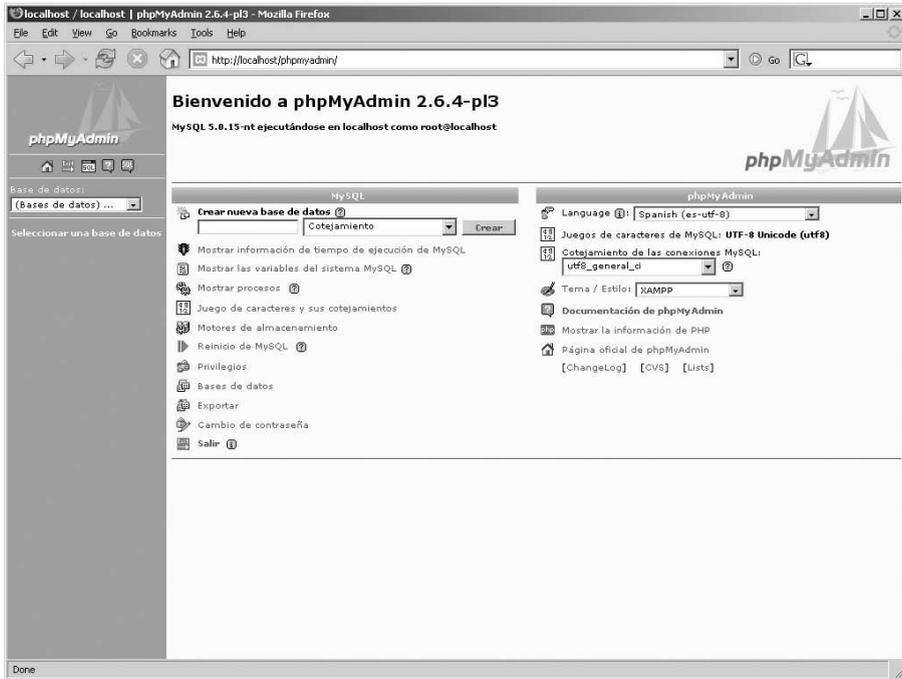


Figura 1.19. PhpMyAdmin instalado por XAMPP.

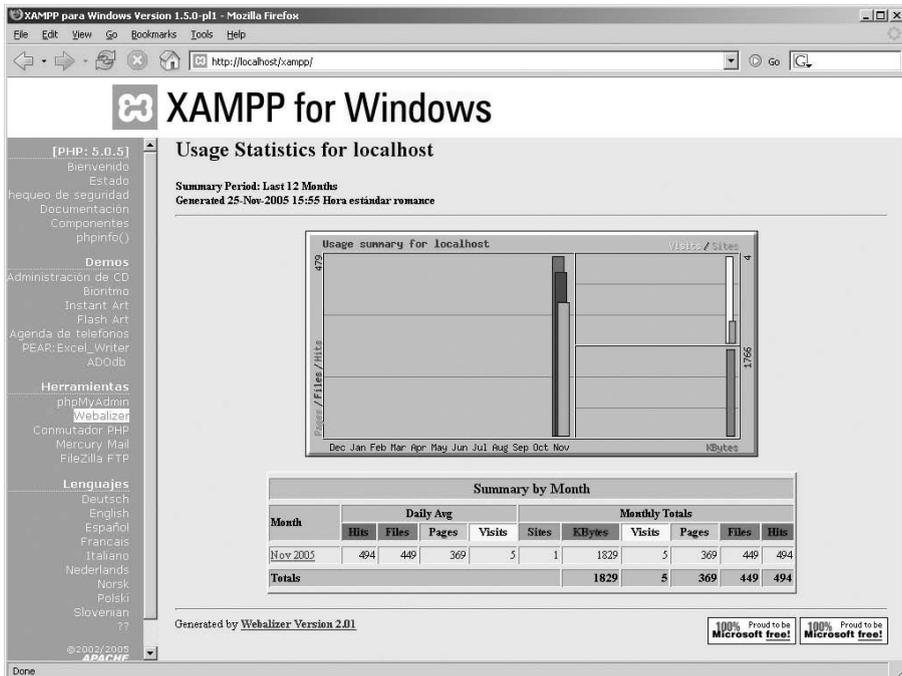


Figura 1.20. Webalizer.

- c) **Commutador PHP:** la versión instalada de PHP es la 5.x. Sin embargo, existen aplicaciones que sólo funcionan correctamente con la versión 4.x de PHP. Para evitar este problema, XAMPP incorpora un script, llamado **php-switch.bat**, que se encuentra en el directorio `ruta/xampp/`. Lea detenidamente la información que le muestra la página web. Para cambiar de versión de PHP, detenga Apache, ejecute el fichero, y reinicie Apache. El fichero **php-switch.bat** detecta en cada caso la versión de PHP activa, y cambia a la otra. Puede ejecutarlo tantas veces como sea necesario.
14. En principio, ya está lista su instalación XAMPP. Recuerde que el contenido que vaya generando (páginas web, instalación de foros, wikis o CMS, etc.) deberá hacerlo dentro del directorio `htdocs`. Dada la variedad de problemas que pueden aparecer, es recomendable que consulte la FAQ para Windows disponible en la URL <http://www.apachefriends.org/en/faq-xampp-windows.html>.

Como puede imaginar, XAMPP no es el único instalador disponible para facilitar la puesta en marcha de un servidor de información. La siguiente tabla le indica los más utilizados.

Tabla 1.7. Instaladores de Apache, PHP, Perl y MySQL para Windows

AppServ	http://www.appservnetwork.com/
Apache2Triad	http://apache2triad.net/
WAMP5	http://www.en.wampserver.com/
EasyWebServer	http://e.w.s.free.fr/site/