

## Instalación y configuración

- [Consideraciones generales de instalación](#)

### Consideraciones generales de instalación

Antes de empezar con la instalación, primero necesita saber para qué quiere utilizar PHP. Existen tres campos principales donde se puede utilizar PHP tal y como se describe en la sección: [¿Qué se puede hacer con PHP?](#)

- Aplicaciones web y sitios web (scripting del lado del servidor)
- Scripting en la línea de comandos
- Aplicaciones de escritorio (GUI)

Para la primera forma mencionada, que es la más común, se necesitan tres cosas: PHP, un servidor web y un navegador web. Seguramente ya disponga del navegador web y, dependiendo de la configuración del sistema operativo, quizá ya tenga un servidor web (p.ej. Apache en Linux y MacOS X; IIS en Windows). También puede alquilar espacio web en una empresa. De esta forma, no se necesita instalar nada, solo tiene que escribir los scripts de PHP, subirlos al servidor que alquile y ver los resultados en su navegador.

En caso de configurar el servidor y PHP por su cuenta, existen dos opciones para el modo de conectar PHP con el servidor. Para muchos servidores, PHP tiene un módulo de interfaz directo (también llamado SAPI). Entre estos servidores se incluyen Apache, Microsoft Internet Information Server, Netscape y iPlanet. Muchos otros servidores tienen soporte para ISAPI, el módulo de interfaz de Microsoft (OmniHTTPd por ejemplo). Si PHP no tiene soporte para el módulo de su servidor web, siempre puede usarlo como procesador CGI o FastCGI. Esto significa configurar el servidor para usar el CGI ejecutable de PHP para procesar cada una de las peticiones a ficheros PHP en el servidor.

Si también está interesado en usar PHP bajo la línea de comandos (p.ej. escribir scripts que autogeneran imágenes de forma offline, o procesar ficheros de texto dependiendo de los argumentos que se les pasen), siempre necesitará el ejecutable de línea de comandos. Para más información, lea la sección sobre [escribir aplicaciones PHP desde la línea de comandos](#). En este caso, no se necesita ningún servidor o navegador.

Con PHP también se pueden escribir aplicaciones GUI de escritorio usando la extensión PHP-GTK. Este enfoque no tiene nada que ver con escribir páginas web, ya que no se muestra nada de HTML, pero gestiona ventanas y objetos dentro de ellas. Para más información acerca de PHP-GTK, por favor [» visite el sitio dedicado a esta extensión](#). PHP-GTK no está incluido en la distribución oficial de PHP.

De aquí en adelante, esta sección trata de la configuración de PHP para servidores web sobre Unix y Windows con interfaces de módulo de servidor y ejecutables CGI. También se puede encontrar información sobre ejecutables de línea de comandos en las siguientes secciones.

El código fuente de PHP y las distribuciones binarias para Windows pueden encontrarse en [» http://www.php.net/downloads.php](http://www.php.net/downloads.php). Recomendamos elegir un [» sitio alternativo](#) cercano para descargar las distribuciones.

### Instalación en sistemas Windows

## Tabla de contenidos

- [Windows Installer \(PHP 5.1.0 y anteriores\)](#)
- [Windows Installer \(PHP 5.2 y posterior\)](#)
- [Manual de Instalación](#)
- [Microsoft IIS](#)
- [Microsoft IIS 5.1 y IIS 6.0](#)
- [Microsoft IIS 7.0 y posteriores](#)
- [Apache 1.3.x en Microsoft Windows](#)
- [Apache 2.x en Microsoft Windows](#)
- [Servidores Sun, iPlanet y Netscape en Microsoft Windows](#)
- [Sambar Server en Microsoft Windows](#)
- [Xitami en Microsoft Windows](#)
- [Construcción de la fuente](#)
- [Instalación de extensiones en Windows](#)
- [Linea de Comandos PHP en Microsoft Windows](#)

Esta sección se aplica a Windows 98/Me y Windows NT/2000/XP/2003. PHP no funciona en plataformas de 16 bits, como Windows 3.1 y, a veces nos referimos a las plataformas soportadas por Windows como Win32.

### Nota:

Windows 98/Me/NT4 ya no se admite a partir de PHP 5.3.0.

### Nota:

Windows 95 ya no se admite a partir de PHP 4.3.0.

Hay dos maneras principales de instalar PHP para Windows: de forma [manual](#) o mediante el [instalador](#).

Si tiene un entorno de desarrollo como Microsoft Visual Studio, también puede [construir](#) PHP desde el código fuente original.

Una vez que haya instalado PHP en su sistema Windows, también puede querer [cargar varias extensiones](#) para mayor funcionalidad.

### Advertencia

Hay varios instaladores todo-en-uno a través de Internet, pero ninguno de esos son avalados por la PHP.net, ya que creemos que usando uno de los paquetes oficiales de Windows de » <http://www.php.net/downloads.php> es la mejor opción para que el sistema este seguro y optimizado.

## Windows Installer (PHP 5.2 y posterior)

Para crear el instalador de las últimas versiones de PHP en Windows se ha usado la tecnología MSI en conjunto con Wix Toolkit (» <http://wix.sourceforge.net/>). Esta aplicación se encarga de instalar y configurar tanto PHP, como las extensiones base y PECL. Además, también permite configurar muchos de los servidores web más extendidos, como IIS, Apache, o Xitami.

En primer lugar, se debe instalar el servidor HTTP (web) deseado, y asegurarse de que funciona. Una vez realizado, puede optarse por alguno de los siguientes métodos de instalación.

## Instalación Normal

Ejecute el instalador MSI y siga las instrucciones que le indica el asistente de instalación. En primer lugar, se le solicitará que seleccione el Servidor Web que desea configurar, junto con algunos detalles de configuración adicionales.

Posteriormente, se le pedirá que indique qué funcionalidades y extensiones desea instalar y habilitar. Podrá indicar si desea instalar o no cada elemento seleccionando "Instalar en el disco local" ("Will be installed on local hard drive") en el menú desplegable que acompaña a cada uno de ellos. Al marcar "Instalar todas las funcionalidades en el disco local" ("Entire feature will be installed on local hard drive"), podrá instalar todas las subcategorías del elemento seleccionado (por ejemplo, al marcar esta opción en "PDO", se instalarán todos los Drivers PDO).

### Advertencia

Se recomienda no instalar todas las extensiones de forma predeterminada, ya que muchas de ellas tienen dependencias con bibliotecas externas a PHP. En lugar de eso, utilice el Modo de Reparación de Instalación (Installation Repair Mode) desde el panel de control de "Agregar o quitar programas". Desde aquí podrá habilitar o deshabilitar a posteriori cualquier extensiones o componente.

Finalmente, el instalador inicializa tanto el fichero *php.ini* como al propio PHP para funcionar en Windows. También podrá configura algunos servidores web para que utilicen PHP. Actualmente es capaz de configurar IIS, Apache, Xitami y Sambar; si utiliza otro servidor web, deberá configurarlo a mano.

## Instalación Silenciosa

El instalador también permite ser ejecutado en modo silencioso, muy útil para Administradores de Sistemas que desean implantar PHP fácilmente. Para usar el modo silencioso:

```
msiexec.exe /i php-VERSION-win32-install.msi /q
```

Puede especificar el directorio de instalación como parámetro al instalador. Por ejemplo, para instalarlo en e:\php:

```
msiexec.exe /i php-VERSION-win32-install.msi /q INSTALLDIR=e:\php
```

Puede usar la misma sintaxis para especificar el Directorio de Configuración de Apache (APACHEDIR), el directorio del Servidor Sambar (SAMBARDIR), o el directorio del Servidor Xitami (XITAMIDIR).

Puede, del mismo modo, especificar qué componentes instalar. Por ejemplo, para instalar la extensión mysqli y el ejecutable CGI:

```
msiexec.exe /i php-VERSION-win32-install.msi /q ADDLOCAL=cgi,ext_php_mysqli
```

La lista actual de componentes para instalar es la siguiente:

```
MainExecutable - ejecutable php.exe ejecutable ( desde PHP 5.2.10/5.3.0 no está disponible; se incluye por omisión)
ScriptExecutable - ejecutable php-win.exe
```

```
ext_php_* - varias extensiones ( por ejemplo: ext_php_mysql para MySQL )
apache13 - módulo de Apache 1.3
apache20 - módulo de Apache 2.0
apache22 - módulo de Apache 2.2
apacheCGI - Ejecutable CGI de Apache
iis4ISAPI - módulo ISAPI de IIS
iis4CGI - ejecutable CGI de IIS
iis4FastCGI - ejecutable CGI de IIS
NSAPI - módulo de servidores Sun/iPlanet/Netscape
netserve - ejecutable CGI de Servidores Web NetServe
Xitami - ejecutable CGI de Xitami
Sambar - módulo ISAPI del Servidor Sambar
CGI - ejecutable php-cgi.exe
PEAR - instalador PEAR
Manual - manual de PHP en formato CHM
```

Si desea más información sobre instalaciones MSI desde la línea de comandos, consulte » <http://msdn.microsoft.com/en-us/library/aa367988.aspx>

## Actualizando PHP con el instalador

Para actualizar, debe ejecutar el instalador indistintamente desde la línea de comandos o del entorno gráfico. El instalador comprobará sus opciones de instalación actuales, eliminará la instalación antigua, y reinstalará PHP con las mismas opciones que ya tenía. Se recomienda utilizar este método para mantener PHP actualizado, en lugar de sobrescribir a mano cada fichero del directorio de instalación.

## Manual de Instalación

Esta sección contiene instrucciones para instalar y configurar manualmente PHP en Windows de Microsoft. Para las instrucciones sobre cómo utilizar el instalador de PHP para instalar y configurar PHP y un servidor web en Windows, consulte [Windows Installer \(PHP 5.2 y posteriores\)](#).

## Selección y descarga de el paquete de distribución PHP

Descarga la distribución binaria de PHP zip » [PHP para Windows: Binarios y Fuentes](#) . Hay varias versiones diferentes del paquete zip - Seleccione por la versión que sea adecuado para el servidor web utilizado:

- Si se utiliza PHP con IIS a continuación elija PHP 5.3 **VC9 Non Thread Safe** o PHP 5.2 **VC6 Non Thread Safe**;
- Si PHP se utiliza con IIS7 o superior y PHP 5.3+, a continuación los archivos binarios VC9 de PHP que se deben utilizar.
- Si se utiliza PHP con Apache 1 o Apache 2 a continuación elija PHP 5.3 **VC6** o PHP 5.2 **VC6**.

### Nota:

Versiones VC9 se compilan con el compilador de Visual Studio 2008 y tiene mejoras en el rendimiento y estabilidad. Las versiones VC9 requieren que tenga la » [Microsoft 2008 C++ Runtime \(x86\)](#) o la » [Microsoft 2008 C++ Runtime \(x64\)](#) instalado.

## La estructura del paquete PHP y el contenido

Descomprimir el contenido del archivo zip en un directorio de su elección, por ejemplo C:\PHP\. La estructura de directorios y archivos extraídos de del zip se verá de la siguiente manera:

### Ejemplo #1 PHP 5 paquete de estructura

```
c:\php
|
|--dev
|   |
|   |--php5ts.lib           -- php5.lib in non thread safe version
|
|--ext                     -- extension DLLs for PHP
|   |
|   |--php_bz2.dll
|   |--php_cpdf.dll
|   |--...
|
|--extras                 -- empty
|
|--pear                   -- initial copy of PEAR
|   |
|   |--go-pear.bat        -- PEAR setup script
|   |--...
|   |--php-cgi.exe       -- CGI executable
|   |--php-win.exe       -- executes scripts without an opened command prompt
|   |--php.exe           -- Command line PHP executable (CLI)
|   |--...
|   |--php.ini-development -- default php.ini settings
|   |--php.ini-production -- recommended php.ini settings
|   |--php5apache2_2.dll  -- does not exist in non thread safe version
|   |--php5apache2_2_filter.dll -- does not exist in non thread safe version
|   |--...
|   |--php5ts.dll        -- core PHP DLL ( php5.dll in non thread safe version)
|   |--...
```

A continuación se muestra la lista de los módulos y archivos ejecutables incluidos en la distribución del archivo zip de PHP:

- *go-pear.bat* - el script de configuración de PEAR. Consulte [» Instalación \(PEAR\)](#) para más detalles.
- *php-cgi.exe* - ejecutable CGI que se puede utilizar cuando se ejecuta PHP en IIS a través de CGI o FastCGI.

- *php-win.exe* - el ejecutable para la ejecución de scripts en PHP sin necesidad de utilizar una ventana de línea de comandos de PHP (Por ejemplo, las aplicaciones PHP que utilizan Windows GUI).
- *php.exe* - el ejecutable para la ejecución de scripts PHP dentro de una interfaz de línea de comandos (CLI) de PHP.
- *php5apache2\_2.dll* - Apache 2.2.X module.
- *php5apache2\_2\_filter.dll* - Apache 2.2.X filter.

## Cambio el fichero *php.ini*

Después de que el contenido del paquete de PHP se ha extraído, copie el *php.ini-production* en *php.ini* en la misma carpeta. Si es necesario, también es posible colocar el *php.ini* en cualquier otro lugar. Pero se requieren pasos adicionales de configuración como se describe en [Configuración de PHP](#).

El fichero *php.ini* le dice a PHP cómo se configura, y cómo trabajar con el ambiente que se ejecuta. Aquí hay una serie de ajustes para el fichero *php.ini* PHP que ayudan a trabajar mejor con Windows. Algunos de estos son opcionales. Hay muchas otras directivas que puedan ser relevantes para su ambiente - diríjase a [lista de directivas php.ini](#) para más información.

Las directivas:

- *extension\_dir* = **<path to extension directory>** - La *extension\_dir* necesidades para indicar el directorio donde las extensiones de PHP se almacenan los ficheros. La ruta puede ser absoluta (i.e. "C:\PHP\ext") o (i.e. ".\ext"). Extensiones que se enumeran más baja en el *php.ini* ficheros que se encuentra en el *extension\_dir* .
- *extension* = **xxxxx.dll** - Para cada extensión que desee habilitar, necesita un correspondiente "extension=" directiva que le dice a PHP que las extensiones en el *extension\_dir* para cargar en tiempo de inicio.
- *log\_errors* = **On** - PHP tiene un error de registro de instalación que se puede utilizar para enviar los errores a un fichero, o de un servicio (i.e. syslog) y trabaja en conjunto con el *error\_log* Directiva a continuación. Cuando se ejecuta bajo IIS, la *log\_errors* debe estar habilitado, con una validez *error\_log* .
- *error\_log* = **<path to the error log file>** - El *error\_log* tiene que especificar la ruta de acceso absoluta, o en relación con el archivo donde los errores de PHP debe estar registrado. Este archivo debe tener permisos de escritura para el servidor web. Los lugares más comunes para este archivo en varios directorios TEMP, por ejemplo "C:\inetpub\temp\php-errors.log".
- *cgi.force\_redirect* = **0** - La presente Directiva se requiere para ejecutar en IIS. Se trata de una protección de la instalación de la guía necesaria por muchos otros servidores web. Sin embargo, lo que permite que en IIS hacer que el motor de PHP a un error en Windows.
- *cgi.fix\_pathinfo* = **1** - Esto le permite acceder a PHP al real path después de la especificación CGI. IIS FastCGI implementa las necesidades de este conjunto.
- *fastcgi.impersonate* = **1** - FastCGI bajo IIS admite la posibilidad de hacerse pasar por tokens de seguridad del cliente que llama. Esto permite que IIS defina el contexto de seguridad que solicita ejecutar.
- *fastcgi.logging* = **0** - FastCGI debe estar deshabilitado en IIS. Si se deja habilitada, los mensajes de cualquier clase son tratados por FastCGI como las condiciones de error que hará que IIS genere una excepción HTTP 500.

Directivas opcionales

- *max\_execution\_time* = **##** - Esta directiva le dice a PHP, el importe máximo de tiempo que puede ejecutar cualquier script determinado. El defecto es 30 segundos. Aumentar el valor de esta directiva, si la aplicación PHP tomar mucho tiempo para ejecutarse.

- *memory\_limit* = **###M** - La cantidad de memoria disponible para el proceso de PHP, en Megabytes. El valor por defecto es 128, lo cual está bien para la mayoría de las aplicaciones PHP. Algunos de los más complejos puede ser que necesite más.
- *display\_errors* = **Off** - Esta directiva le dice a PHP si se incluyen los mensajes de error en el corriente que devuelve al servidor web. Si se establece en "On", entonces PHP enviará lo que las clases de errores que se define con la *error\_reporting* directiva de nuevo al servidor web como parte de la secuencia de error. Por razones de seguridad, se recomienda "Off" en los servidores de producción con el fin de no revelar ninguna información sensible de la seguridad que a menudo se incluyen en los mensajes de error.
- *open\_basedir* = **<paths a directorios, separados por semicolon>**, por ejemplo, *openbasedir*="C:\inetpub\wwwroot;C:\inetpub\temp". Esta directiva especifica las rutas de directorio en donde PHP se le permite realizar las operaciones del sistema de archivos. Cualquier operación de fichero fuera de las rutas especificadas se producirá un error. Esta directiva es especialmente útil para bloquear la instalación de PHP en entornos de alojamiento compartido para evitar que scripts PHP de acceso a los archivos fuera del directorio raíz del sitio web.
- *upload\_max\_filesize* = **###M** and *post\_max\_size* = **###M** - El tamaño máximo permitido de un archivo subido y los datos posteriores, respectivamente. Los valores de estas directivas serán aumenta si las aplicaciones PHP necesita realizar grandes cargas, como por ejemplo fotos o archivos de vídeo.

PHP esta ahora configurado en el sistema. El siguiente paso es elegir un servidor web, a fin de que pueda ejecutar PHP. Elegir un servidor web de la tabla de contenido.

Además de ejecutar PHP a través de un servidor web, PHP puede ejecutar desde la línea de comandos como un script **.BAT**. Ver [PHP línea de comandos de Microsoft Windows](#) para más detalles.

## Microsoft IIS 7.0 y posteriores

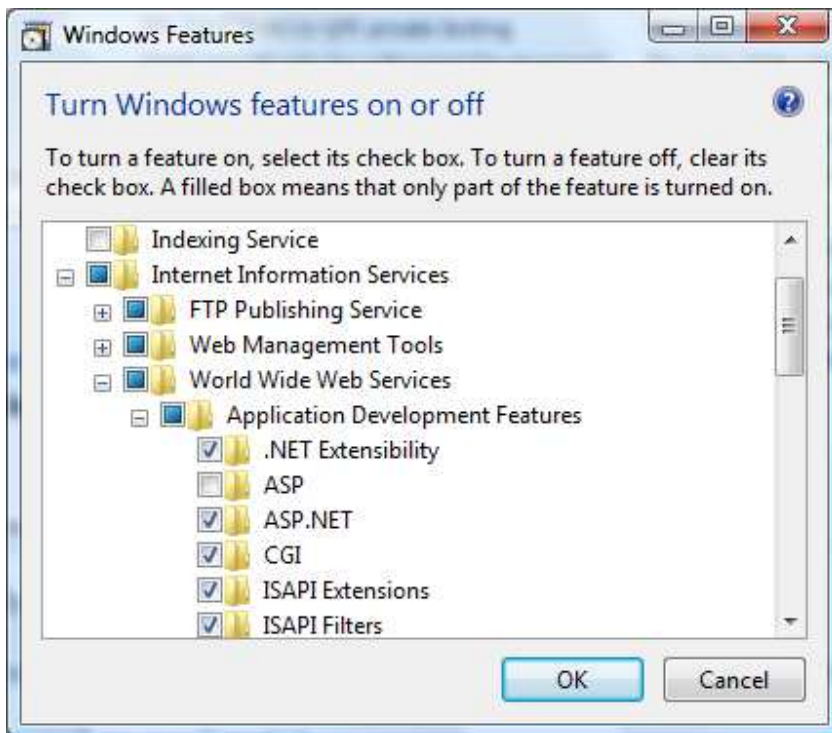
Esta sección contiene instrucciones para configurar en forma manual Internet Information Services (IIS) 7.0 y posteriores para trabajar con PHP en Microsoft Windows Vista SP1, Windows 7, Windows Server 2008 and Windows Server 2008 R2. Para obtener instrucciones sobre la configuración de IIS 5.1 and IIS 6.0 on Windows XP and Windows Server 2003 refiérase a [Microsoft IIS 5.1 and IIS 6.0](#).

### Habilitación de apoyo FastCGI en IIS

Módulo FastCGI está deshabilitado en la instalación por defecto de IIS. Los pasos para que pueda variar según la versión de Windows se está utilizando.

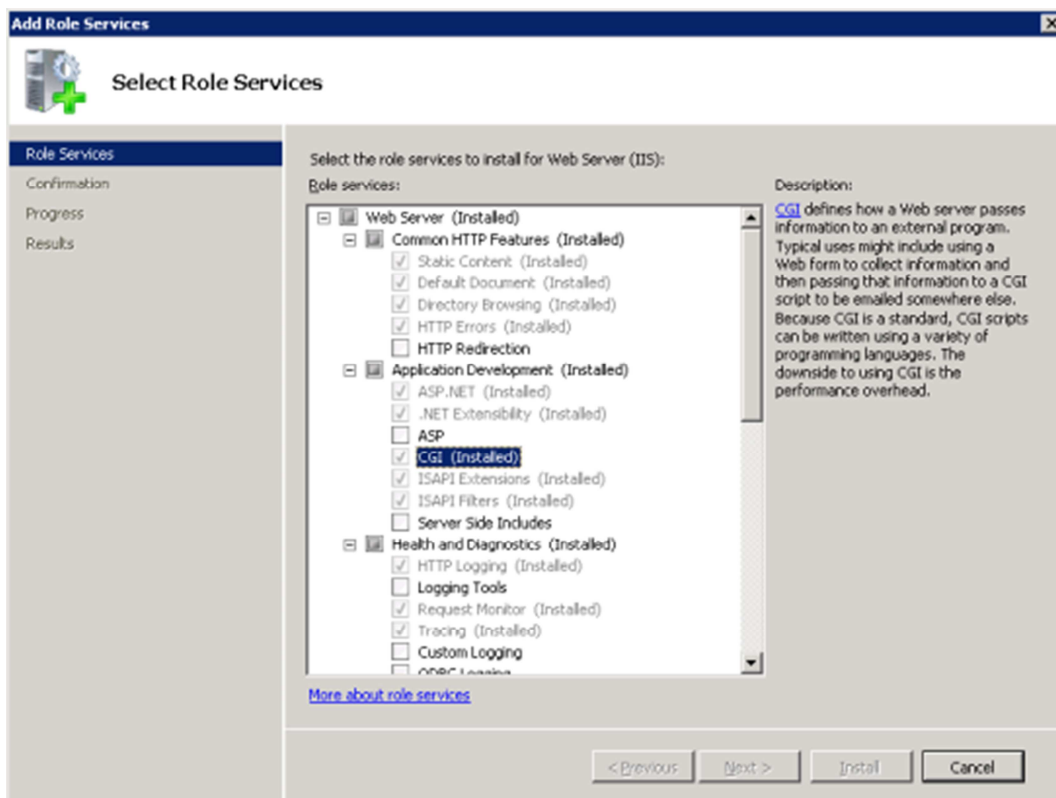
Para activar el soporte FastCGI en Windows Vista SP1 y Windows 7:

1. En el menu de inicio de Windows seleccione "Run:", teclee "optionalfeatures.exe" y click "Ok";
2. En el "Windows Features" cuadro de diálogo "Internet Information Services", "World Wide Web Services", " Características Desarrollo de aplicaciones " y luego permitir que el "CGI" checkbox;
3. Haga clic en Aceptar y espere hasta que se complete la instalación



Para activar el soporte FastCGI en Windows Server 2008 y Windows Server 2008 R2:

1. En el menú Inicio de Windows elegir "Run:", teclee "CompMgmtLauncher" y click "Ok";
2. En el "Web Server (IIS)" El rol no está presente en el marco del "Roles" nodo, a continuación, añadir haciendo clic en "Adicionar Roles";
3. Si la "Web Server (IIS)" role esta presente, entonces haga click "Add Role Services" y luego permitir que el "CGI" casilla de verificación en grupo "Desarrollo de Aplicaciones" ;
4. Click "Next" y luego "Install" ay espere a que la instalación se complete.





## Configurar IIS para procesar las solicitudes de PHP

Descargue e instale PHP de acuerdo a las instrucciones descritas en [manual installation steps](#)

### Nota:

No construir thread-safe de PHP se recomienda cuando IIS utilizando. El no-thread-safe crea están disponibles en [» PHP para Windows: binarios y fuentes.](#)

Configure el CGI y la configuración específica de FastCGI *php.ini* fichero como se muestra a continuación:

### Ejemplo #1 CGI y FastCGI en la configuración *php.ini*

```
fastcgi.impersonate = 1
fastcgi.logging = 0
cgi.fix_pathinfo=1
cgi.force_redirect = 0
```

Configurar la asignación de IIS controlador para PHP utilizando la interfaz de usuario Administrador de IIS o una herramienta de línea de comandos.

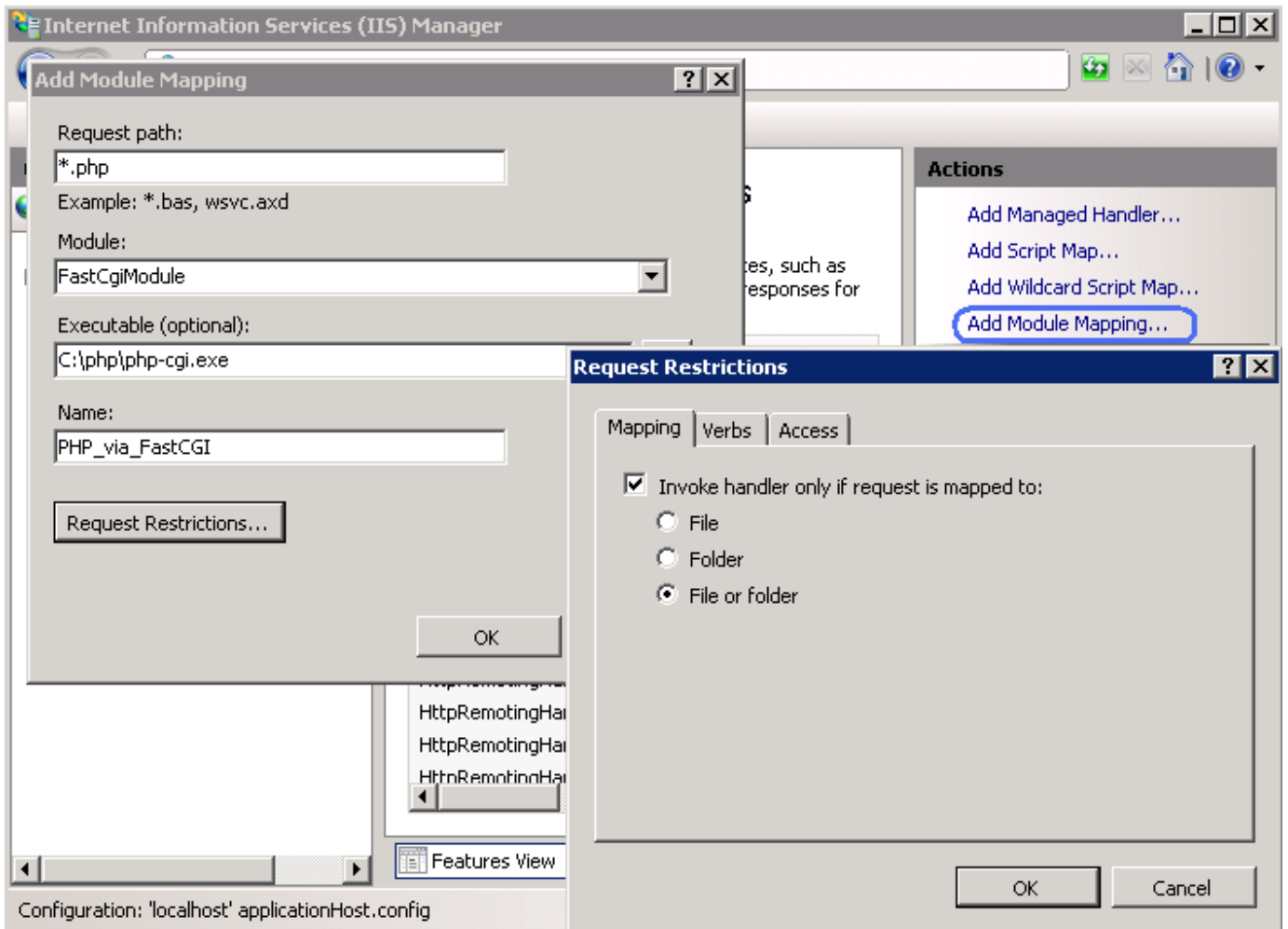
### ***El uso de IIS interfaz de usuario de Manager para crear una asignación de controlador para PHP***

Siga estos pasos para crear una asignación de controlador de IIS para PHP en la interfaz de usuario Administrador de IIS:

1. En el menú Inicio de Windows elegir "Run:", escriba "inetmgr" y click "Ok";
2. En la interfaz de usuario de IIS Manager, seleccione el nodo de servidor en el "Connections" vista de árbol;
3. En el "Características" Página de abrir el "Handler Mappings" feature;



4. En el "Actions" haga click en panel "Add Module Mapping...";
5. En el "Add Module Mapping" diálogo entre lo siguiente:
  - Solicitud de ruta: \*.php
  - Modulo: FastCgiModule
  - Ejecutable: C:\[Path to PHP installation]\php-cgi.exe
  - Nombre: PHP\_via\_FastCGI
6. Haga Click en "Solicitar Restricciones" botón y luego configurar la asignación de invocar controlador sólo si la solicitud se asigna a un archivo o una carpeta;
7. Haga clic en Aceptar en todos los cuadros de diálogo para guardar la configuración.



### **Utilizando la herramienta de línea de comandos para crear una asignación de controlador para PHP**

Utilice el comando mostrado a continuación para crear un grupo de IIS FastCGI proceso que utilizará *php-cgi.exe* ejecutable para la tramitación de solicitudes de PHP. Vuelva a colocar el valor del parámetro *fullPath* con la ruta absoluta al fichero *php-cgi.exe*.

#### **Ejemplo #2 > Creación del proceso del grupo IIS FastCGI**

```
%windir%\system32\inetsrv\appcmd set config /section:system.webServer/fastCGI ^
/[fullPath='c:\PHP\php-cgi.exe']
```

Configurar IIS PHP para manejar las peticiones concretas de la ejecución del comando que se muestra a continuación. Vuelva a colocar el valor del parámetro *scriptProcessor* con la ruta de acceso absoluta al fichero *php-cgi.exe*.

#### **Ejemplo #3 Creación de mapas manejador de peticiones PHP**

```
%windir%\system32\inetsrv\appcmd set config /section:system.webServer/handlers ^
/[name='PHP_via_FastCGI', path='*.php', verb='*', modules='FastCgiModule', ^
scriptProcessor='c:\PHP\php-cgi.exe', resourceType='Either']
```

Este comando crea una asignación de controlador de IIS para la extensión de archivo *\*.php*, lo que resultará en todas las direcciones URL que terminan con *.php* siendo manejado por el módulo FastCGI.

## Nota:

En este punto la necesidad de instalación y configuración de los pasos se han completado. Las instrucciones a continuación son opcionales, pero altamente recomendable para lograr una funcionalidad óptima y el rendimiento de PHP en IIS.

## Suplantación de identidad y acceso a archivos del sistema

Se recomienda habilitar la suplantación en PHP FastCGI cuando IIS utilizando. Esta está controlado por la directiva *fastcgi.impersonate* en el fichero *php.ini*. Cuando está habilitada la suplantación, PHP realizará todas las operaciones del sistema de ficheros en nombre de la cuenta de usuario que ha sido determinado por la autenticación de IIS. Esto asegura que, aunque el proceso mismo PHP es compartido a través de diferentes sitios web IIS, los scripts PHP en los sitios web no será capaz de acceder a los demás ficheros, siempre y cuando las cuentas de usuario se utilizan para la autenticación de IIS en cada sitio web.

Por ejemplo IIS 7, en su configuración por defecto, se ha habilitado la autenticación anónima con incorporada IUSR cuenta de usuario utilizada como una identidad por defecto. Esto significa que para que IIS ejecutar scripts PHP, es necesario conceder a la cuenta IUSR permiso de lectura en estos scripts. Si las aplicaciones PHP necesidad de realizar operaciones de escritura de ciertos ficheros o escribir ficheros en una carpeta entonces cuenta IUSR debe tener permiso para escribir en ellos.

Para determinar qué cuenta de usuario se utiliza como una identidad anónima en IIS 7 utilice el siguiente comando. Vuelva a colocar el "Sitio Web predeterminado" con el nombre del sitio Web de IIS que utiliza. En la producción configuración XML aspecto de elemento para el atributo *userName*.

### Ejemplo #4 La determinación de la cuenta utilizada como IIS identidad anónima

```
%windir%\system32\inetsrv\appcmd.exe list config "Default Web Site" ^
/section:anonymousAuthentication
<system.webServer>
<security>
<authentication>
<anonymousAuthentication enabled="true" userName="IUSR" />
</authentication>
</security>
</system.webServer>
```

## Nota:

Si el atributo *userName* no está presente en el elemento *anonymousAuthentication*, o si se establece una cadena vacía, entonces significa que la identidad del grupo de aplicaciones se utiliza como un identidad anónima para ese sitio web.

Para modificar la configuración de permisos de archivos y carpetas, utilice la interfaz de usuario del Explorador de Windows o comando *icacls*.

### Ejemplo #5 Configuración de archivo de los permisos de acceso

```
icacls C:\inetpub\wwwroot\upload /grant IUSR:(OI)(CI)(M)
```

## Establecer *index.php* como un documento predeterminado en IIS

Los documentos por defecto de IIS se utilizan para las solicitudes HTTP que no se especifica un nombre de documento. Con las aplicaciones PHP, *index.php* por lo general actúa como un documento

predeterminado. Para añadir *index.php* a la lista de los documentos por defecto de IIS, utilice este comando:

### Ejemplo #6 Establecer *index.php* como un documento predeterminado en IIS

```
%windir%\system32\inetsrv\appcmd.exe set config ^
-section:system.webServer/defaultDocument /+"files.[value='index.php']" ^
/commit:apphost
```

## FastCGI y configuración de PHP Reciclaje

Configurar la configuración de IIS FastCGI PHP para el reciclaje de los procesos mediante los comandos de abajo. La configuración FastCGI *instanceMaxRequests* controla cuántas peticiones se procesado por un solo proceso *php-cgi.exe* antes de IIS se cierra. La variable de entorno PHP *PHP\_FCGI\_MAX\_REQUESTS* controla el número de veces que un solo proceso *php-cgi.exe* se encargará antes de que se recicla sí misma. Asegúrese de que el valor especificado para la configuración FastCGI *InstanceMaxRequests* es menor o igual al valor especificado para *PHP\_FCGI\_MAX\_REQUESTS*.

### Ejemplo #7 Configuración de PHP FastCGI y el reciclaje

```
%windir%\system32\inetsrv\appcmd.exe set config -section:system.webServer/fastCgi ^
/[fullPath='c:\php\php-cgi.exe'].instanceMaxRequests:10000
%windir%\system32\inetsrv\appcmd.exe set config -section:system.webServer/fastCgi ^
/+"[fullPath='C:\{php_folder}\php-cgi.exe'].environmentVariables.^
[name='PHP_FCGI_MAX_REQUESTS',value='10000']"
```

## FastCGI ajustes de tiempo de espera

Aumentar el tiempo de espera para la configuración de FastCGI si se espera tener a largo ejecutar scripts PHP. Los dos valores que los tiempos de espera de control *activityTimeout* y *requestTimeout*. Utilice los comandos siguientes para cambiar la configuración de tiempo de espera. Asegúrese de sustituir el valor en el parámetro a *fullPath* contendrá la ruta absoluta a el fichero *php-cgi.exe*.

### Ejemplo #8 Configuración de los ajustes de tiempo de espera FastCGI

```
%windir%\system32\inetsrv\appcmd.exe set config -section:system.webServer/fastCgi ^
/[fullPath='C:\php\php-cgi.exe',arguments=''].activityTimeout:"90" /commit:apphost
%windir%\system32\inetsrv\appcmd.exe set config -section:system.webServer/fastCgi ^
/[fullPath='C:\php\php-cgi.exe',arguments=''].requestTimeout:"90" /commit:apphost
```

## Cambiar la ubicación del fichero *php.ini*

PHP busca el fichero *php.ini* en [varios lugares](#) y es posible cambiar las ubicaciones predeterminadas del fichero *php.ini* utilizando variable de entorno *PHPRC*. Encomendar a PHP para cargar el archivo de configuración desde una ubicación personalizada, ejecute el comando se muestra a continuación. La ruta de acceso absoluta al directorio con el fichero *php.ini* debe ser específica como un valor de variable de entorno *PHPRC*.

### Ejemplo #9 Cambiar la ubicación del fichero *php.ini*

```
appcmd.exe set config -section:system.webServer/fastCgi ^
/+"[fullPath='C:\php\php.exe',arguments=''].environmentVariables.^
[name='PHPRC',value='C:\Some\Directory\']" /commit:apphost
```

## Apache 2.x en Microsoft Windows

Esta sección contiene notas y sugerencias específicas de Apache 2.x instaladas con PHP en sistemas Microsoft Windows. También hay [instrucciones y notas para usuarios de Apache 1.3.x en una nueva página](#).

### Nota:

Por favor lea primero [el manual de instalación](#)

### Nota: Soporte Apache 2.2

Los usuarios de Apache 2.2 deben tener en cuenta que el archivo DLL para Apache 2.2 es llamado *php5apache2\_2.dll* a cambio de *php5apache2.dll* y sólo está disponible para PHP 5.2.0 y posteriores. Ver también [» http://snaps.php.net/](http://snaps.php.net/)

Se le recomienda consultar la [» Documentación de Apache](#) Para obtener un conocimiento básico del servidor Apache 2.x. También considere leer las [» notas específicas de Windows](#) para Apache 2.x antes de seguir leyendo aquí.

Apache 2.x está diseñado para ejecutarse en las versiones de Windows designado como servidor de plataforma, tales como Windows NT 4.0, Windows 2000, Windows XP o Windows 7. Aunque Apache 2.x funciona bastante bien en Windows 9x, el soporte en estas plataformas es incompleta, y algunas cosas no funcionan correctamente. No existe un plan para remediar esta situación.

Descargue la versión más reciente de [» Apache 2.x](#) y un accesorio de la versión de PHP. Siga los [pasos del Manual de Instalación](#) y regrese a continuar con la integración de PHP y Apache.

Hay tres formas de configurar PHP para trabajar con Apache 2.x en Windows. Puede ejecutar PHP como un controlador, como CGI o bajo FastCGI.

**Nota:** Recuerde que cuando agrega rutas como valores en los archivos de configuración de Apache en Windows, todas las barras invertidas tal como `c:\directory\file.ext` deberían ser convertidas en barras comunes: `c:/directory/file.ext`. Una barra común al final será necesaria para directorios.

## Instalación como un controlador de Apache

Es necesario insertar las siguientes líneas en Apache *httpd.conf* archivo de configuración para cargar el módulo PHP para Apache 2.x:

### Ejemplo #1 PHP and Apache 2.x como controlador

```
#
LoadModule php5_module "c:/php/php5apache2.dll"
AddHandler application/x-httpd-php .php

# Configurar la ruta de php.ini
PHPIniDir "C:/php"
```

**Nota:** Recuerda sustituir la ruta de acceso actual a PHP para el `C:/php/` en los ejemplos anteriores. Tenga cuidado en utilizar ya sea *php5apache2.dll* o *php5apache2\_2.dll* en la directiva `LoadModule` y verificar que el fichero de referencia se encuentra localizado en la ruta de fichero que se señala en la presente directiva.

La configuración anterior permite habilitar PHP como controlador de cualquier fichero que tiene una extensión `.php`, aunque existen otras extensiones de fichero. Por ejemplo, un fichero

llamado *example.php.txt* será ejecutado por el controlador de PHP. Para garantizar que sólo los ficheros que **terminen en .php** son ejecutados, se utiliza la siguiente configuración en su lugar:

```
<FilesMatch \.php$>
    SetHandler application/x-httpd-php
</FilesMatch>
```

## Ejecutar PHP como CGI

Se debe consultar la [Documentación de Apache CGI](#) para una comprensión más completa de ejecutar CGI en Apache

Para ejecutar PHP como CGI, tendrás que colocar los ficheros php-cgi en un directorio designado como directorio CGI usando la directiva ScriptAlias.

A continuación, se necesita insertar una `#!` línea en los ficheros PHP, apuntando a la ubicación del binario de PHP:

### Ejemplo #2 PHP y Apache 2.x como CGI

```
#!C:/php/php.exe
<?php
    phpinfo();
?>
```

#### Advertencia

Al usar el modo CGI, su servidor está expuesto a diferentes ataques. Por favor, leer la sección [Seguridad con CGI](#) para aprender como defenderse de estos ataques.

## Ejecutar PHP bajo FastCGI

Ejecutar PHP bajo FastCGI tiene una serie de ventajas con respecto a ejecutarlo como un CGI. Si lo establece de esta manera es bastante sencillo:

Obtener `mod_fcgid` de [http://httpd.apache.org/mod\\_fcgid/](http://httpd.apache.org/mod_fcgid/). Los binarios de Win32 están disponibles para la descargar en ese sitio. Instalar el módulo de acuerdo con las instrucciones que vienen con él.

Configurar el servidor web como se muestra abajo, teniendo cuidado de ajustar cualquier ruta que refleje como se tiene instalado las cosas en el sistema:

### Ejemplo #3 Configurar Apache para ejecutar PHP como FastCGI

```
LoadModule fcgid_module modules/mod_fcgid.so
```

```
# Donde está el fichero php.ini?
FcgidInitialEnv PHPRC "c:/php"
```

```
AddHandler fcgid-script .php
FcgidWrapper "c:/php/php-cgi.exe" .php
```

Los ficheros con extensión `.php` ahora serán ejecutados por PHP FastCGI

## Linea de Comandos PHP en Microsoft Windows

Esta sección contiene notas y sugerencias específicas para conseguir ejecutar PHP en la línea de comandos para Windows.

### Nota:

Usted debe leer primero el [manual de instalación](#)

Cómo ejecutar PHP desde la línea de comandos sin hacer ningún cambio a Windows.

```
C:\PHP5\php.exe -f "C:\PHP Scripts\script.php" -- -arg1 -arg2 -arg3
```

Pero hay algunos pasos sencillos que puede seguir para hacer esto simple. Algunas de estas medidas ya se deberían haber tomado, pero se repiten aquí para ser capaz de proporcionar una completa secuencia paso a paso.

- Añada la ubicación del ejecutable de PHP (*php.exe*, *php-win.exe* o *php-cli.exe* dependiendo de su versión de PHP y las preferencias de presentación) a la variable de entorno *PATH*. Lea más acerca de cómo añadir el directorio de PHP *PATH* en la [entrada correspondiente de preguntas frecuentes](#).
- Adicione la extensión **.PHP** de la variable de entorno *PATHEXT*. Esto se puede hacer al mismo tiempo, con la modificación de la variable de entorno *PATH*. Siga los mismos pasos como se describe en las [Preguntas frecuentes](#) pero habría que modificar la variable de entorno *PATHEXT* en lugar de la variable de entorno *PATH*.

### Nota:

La posición en que se coloca el **.PHP** determina qué secuencia de comandos o programa se ejecuta cuando hay nombre de archivos coincidentes. Por ejemplo, colocar **.PHP** antes de **.BAT** hará que el script se ejecute, en lugar de el archivo por lotes, si hay un fichero por lotes con el mismo nombre.

- Asociar la extensión **.PHP** con un tipo de fichero. Esta se hace ejecutando el siguiente comando:

```
assoc .php=phpfile
```
- Asociar el tipo de fichero **phpfile** con el adecuado ejecutable PHP. Esto se hace ejecutando el siguiente comando:

```
ftype phpfile="C:\PHP5\php.exe" -f "%1" -- %~2
```

Si sigue estos pasos permitirá ejecutar scripts PHP desde cualquier directorio sin necesidad de escribir el ejecutable PHP o la extensión **.PHP** y todos los parámetros, será enviada a el script para el procesamiento.

En el ejemplo a continuación se detallan algunos de los cambios de registro que se puede hacer manualmente.

### Ejemplo #1 Registro de cambios

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SOFTWARE\Classes\.php]
@="phpfile"
"Content Type"="application/php"

[HKEY_LOCAL_MACHINE\SOFTWARE\Classes\phpfile]
@="PHP Script"
```



```
"EditFlags"=dword:00000000
"BrowserFlags"=dword:00000008
"AlwaysShowExt"=""

[HKEY_LOCAL_MACHINE\SOFTWARE\Classes\phpfile\DefaultIcon]
@="C:\\PHP5\\php-win.exe,0"

[HKEY_LOCAL_MACHINE\SOFTWARE\Classes\phpfile\shell]
@="Open"

[HKEY_LOCAL_MACHINE\SOFTWARE\Classes\phpfile\shell\Open]
@("&Open"

[HKEY_LOCAL_MACHINE\SOFTWARE\Classes\phpfile\shell\Open\command]
@="\"C:\\PHP5\\php.exe\" -f \"%1\" -- %~2"
```

Con estos cambios el mismo comando se puede escribir como:

```
"C:\PHP Scripts\script" -arg1 -arg2 -arg3
```

O, si la ruta de acceso **"C:\PHP Scripts"** está en el variable de entorno **PATH**:

```
script -arg1 -arg2 -arg3
```

#### **Nota:**

Hay un pequeño problema si tiene intención de utilizar esta técnica y usar scripts PHP como filtro de línea de comandos, como el ejemplo a continuación:

```
dir | "C:\PHP Scripts\script" -arg1 -arg2 -arg3
```

```
0
```

```
dir | script -arg1 -arg2 -arg3
```

Es posible que el script simplemente se cuelga y no salga nada. Para hacer esto operacional, es necesario hacer otro cambio en el registro.

```
Windows Registry Editor Version 5.00
```

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\policies\Explorer]
"InheritConsoleHandles"=dword:00000001
```

Información adicional sobre este tema se pueden encontrar en el [» artículo de Microsoft: 321788](#)