

# Manual de PHP

Stig Sæther Bakken  
Alexander Aulbach  
Egon Schmid  
Jim Winstead  
Lars Torben Wilson  
Rasmus Lerdorf  
Andrei Zmievski  
Jouni Ahto

**Editado por**

**Rafael Martínez**

16-11-2001

[Copyright](#) © 1997, 1998, 1999, 2000, 2001 por por el Grupo de documentación de PHP

---

## **Tabla de contenidos**

### [Prefacio](#)

[Sobre este Manual](#)

[Sobre la traducción](#)

### I. [Conceptos Básicos](#)

1. [Introducción](#)

2. [Instalación](#)

3. [Configuración](#)

4. [Seguridad](#)

### II. [Referencia del Lenguaje](#)

5. [Sintaxis básica](#)

6. [Types](#)

7. [Variables](#)

8. [Constantes](#)

9. [Expresiones](#)

10. [Operadores](#)

11. [Estructuras de Control](#)
12. [Funciones](#)
13. [Clases y Objetos](#)
14. [References Explained](#)

### III. [Características](#)

15. [Manejando errores](#)
16. [Creando imágenes GIF](#)
17. [Autenticación HTTP con PHP](#)
18. [Cookies](#)
19. [El envío de archivos](#)
20. [Usando archivos remotos](#)
21. [Manejando conexiones](#)
22. [Conexiones persistentes a bases de datos](#)
23. [Safe Mode](#)

### IV. [Referencia de las Funciones](#)

- I. [Funciones específicas de Apache](#)
- II. [Funciones de matrices](#)
- III. [Funciones Ortográficas](#)
- IV. [Funciones matemáticas de precisión arbitraria](#)
- V. [Bzip2 Compression Functions](#)
- VI. [Funciones de calendario](#)
- VII. [CCVS API Functions](#)
- VIII. [soporte de las funciones COM para Windows](#)
- IX. [Funciones de Clases/Objetos](#)
- X. [Funciones de ClibPDF](#)
- XI. [CURL, Client URL Library Functions](#)
- XII. [Funciones de pago electrónico](#)
- XIII. [Crédit Mutuel CyberMUT functions](#)
- XIV. [Character type functions](#)
- XV. [Funciones de la capa de abstraccion de bases de datos \(dbm-style\)](#)
- XVI. [Funciones de fecha y hora](#)
- XVII. [Funciones para dBase](#)
- XVIII. [Funciones dbm](#)
- XIX. [dbx functions](#)
- XX. [DB++ Functions](#)
- XXI. [Funciones con directorios](#)

XXII. [Funciones de DOM XML](#)

XXIII. [Error Handling and Logging Functions](#)

XXIV. [FrontBase Functions](#)

XXV. [Funciones filePro](#)

XXVI. [Funciones del sistema de ficheros](#)

XXVII. [Funciones Forms Data Format \(Formato de Datos de Formularios\)](#)

XXVIII. [Funciones FTP](#)

XXIX. [Function Handling functions](#)

XXX. [GNU Gettext](#)

XXXI. [GMP functions](#)

XXXII. [Funciones HTTP](#)

XXXIII. [Funciones para Hyperwave](#)

XXXIV. [Funciones para ICAP - Internet Calendar Application Protocol](#)

XXXV. [iconv functions](#)

XXXVI. [Funciones de imágenes](#)

XXXVII. [Funciones IMAP](#)

XXXVIII. [Funciones para Informix](#)

XXXIX. [Funciones InterBase](#)

XL. [Ingres II functions](#)

XLI. [IRC Gateway Functions](#)

XLII. [Java](#)

XLIII. [Funciones LDAP](#)

XLIV. [Funciones de Correo](#)

XLV. [Funciones matemáticas](#)

XLVI. [Multi-Byte String Functions](#)

XLVII. [MCAL functions](#)

XLVIII. [Funciones Criptográficas](#)

XLIX. [Funciones Hash](#)

L. [Funciones de Microsoft SQL Server](#)

LI. [Ming functions for Flash](#)

LII. [Miscelánea de funciones](#)

LIII. [mnoGoSearch Functions](#)

LIV. [funciones mSQL](#)

LV. [Funciones MySQL](#)

LVI. [Funciones de Red](#)

LVII. [ODBC functions](#)

[LVIII. Funciones de Oracle 8](#)  
[LIX. OpenSSL functions](#)  
[LX. Funciones Oracle](#)  
[LXI. Ovrimos SQL functions](#)  
[LXII. Output Control Functions](#)  
[LXIII. Object property and method call overloading](#)  
[LXIV. PDF functions](#)  
[LXV. Verisign Payflow Pro functions](#)  
[LXVI. opciones e información de PHP](#)  
[LXVII. Funciones POSIX](#)  
[LXVIII. Funciones de PostgreSQL](#)  
[LXIX. Funciones de ejecución de programas](#)  
[LXX. Printer functions](#)  
[LXXI. Pspell Functions](#)  
[LXXII. GNU Readline](#)  
[LXXIII. Funciones GNU Recode](#)  
[LXXIV. Funciones de expresiones regulares compatibles con Perl](#)  
[LXXV. Funciones para expresiones regulares](#)  
[LXXVI. Satellite CORBA client extension](#)  
[LXXVII. Funciones Semáforo y de memoria compartida](#)  
[LXXVIII. SESAM database functions](#)  
[LXXIX. Session handling functions](#)  
[LXXX. Shared Memory Functions](#)  
[LXXXI. Shockwave Flash functions](#)  
[LXXXII. Funciones SNMP](#)  
[LXXXIII. Socket functions](#)  
[LXXXIV. Funciones de cadenas](#)  
[LXXXV. Funciones de Sybase](#)  
[LXXXVI. Funciones URL](#)  
[LXXXVII. Funciones sobre variables](#)  
[LXXXVIII. Funciones WDDX](#)  
[LXXXIX. Funciones de intérprete XML](#)  
[XC. XSLT functions](#)  
[XCI. YAZ](#)  
[XCII. NIS funciona](#)  
[XCIII. Zip File Functions \(Read Only Access\)](#)

XCIV. [Funciones de Compresión](#)

V. [PEAR: the PHP Extension and Application Repository](#)

24. [About PEAR](#)

25. [PEAR Coding Standards](#)

XCV. [PEAR Reference Manual](#)

VI. [FAQ: Frequently Asked Questions](#)

26. [General Information](#)

27. [Mailing lists](#)

28. [Obtaining PHP](#)

29. [Database issues](#)

30. [Installation](#)

31. [Build Problems](#)

32. [Using PHP](#)

33. [PHP and HTML](#)

34. [PHP and COM](#)

35. [PHP and other languages](#)

36. [Migrating from PHP 2 to PHP 3](#)

37. [Migrating from PHP 3 to PHP 4](#)

38. [Miscellaneous Questions](#)

VII. [Apéndices](#)

A. [Migrando de PHP/FI 2.0 a PHP 3.0](#)

B. [Migrating from PHP 3.0 to PHP 4.0](#)

C. [Desarrollo en PHP](#)

D. [El debugger de PHP](#)

E. [PHP reserved words](#)

F. [PHP Resource Types](#)

G. [Aliases list](#)

H. [Using PHP from the command line](#)

---

[Siguiete](#)

Prefacio

# Manual de PHP

---

## Copyright

Este manual es © Copyright 1997, 1998, 1999, 2000, 2001 del Grupo de documentación de PHP. Los miembros de este grupo se encuentran listados en la [primera página de este manual](#).

Este manual puede ser redistribuido bajo los términos de la "GNU General Public License" publicada por la "Free Software Foundation"; tanto bajo la versión 2 de esta licencia o bajo versiones posteriores.

---

[Inicio](#)

# Prefacio

## Tabla de contenidos

[Sobre este Manual](#)

[Sobre la traducción](#)

PHP, acrónimo de "PHP: Hypertext Preprocessor", es un lenguaje interpretado de alto nivel embebido en páginas HTML. La mayoría de su sintaxis es similar a C, Java y Perl, con solamente un par de características PHP específicas. La meta de este lenguaje es permitir escribir a los creadores de páginas web, páginas dinámicas de una manera rápida y fácil.

# Sobre este Manual

Este manual está escrito en SGML usando [DocBook DTD](#) y [DSSSL](#) (Document Style and Semantics Specification Language) para su creación. Las herramientas usadas para crear las versiones HTML, TeX y RTF son [Jade](#), escrita por [James Clark](#) y [The Modular DocBook Stylesheets](#) escrita por [Norman Walsh](#). El marco de trabajo de la documentación de PHP fue creado por [Stig Sæther Bakken](#).

# Sobre la traducción

La traducción del manual de PHP al español ha sido posible gracias a la colaboración de un gran número de traductores, que desinteresadamente han usado su tiempo para que todos podamos tener una versión en nuestra lengua de esta documentación.

(Aqui vendra la lista de colaboradores)



# I. Conceptos Básicos

## Tabla de contenidos

1. [Introducción](#)
2. [Instalación](#)
3. [Configuración](#)
4. [Seguridad](#)

# Capítulo 1. Introducción

## Tabla de contenidos

[Qué es PHP?](#)

[Qué se puede hacer con PHP?](#)

[Corta historia de PHP](#)

## Qué es PHP?

PHP (acronimo de "PHP: Hypertext Preprocessor") es un lenguaje interpretado de alto nivel embebido en páginas HTML y ejecutado en el servidor.

Una respuesta corta y concisa, pero que significa realmente? Un ejemplo nos aclarará las cosas:

### Ejemplo 1-1. Un ejemplo introductorio

```
<html>
  <head>
    <title>Ejemplo PHP</title>
  </head>
  <body>
    <?php echo "Hola, este es un ejemplo con PHP!"; ?>
  </body>
</html>
```

Podemos ver que no es lo mismo que un script CGI escrito en otro lenguaje de programación como Perl o C -- En vez de escribir un programa con muchos comandos para crear una salida en HTML, escribimos el código HTML con cierto código PHP embebido (introducido) en el mismo, que producirá cierta salida (en nuestro ejemplo, producir un texto). El código PHP se incluye entre [etiquetas especiales de comienzo y final](#) que nos permitirán entrar y salir del

modo PHP.

Lo que distingue a PHP de la tecnología Javascript, la cual se ejecuta en la máquina cliente, es que el código PHP es ejecutado en el servidor. Si tuviesemos un script similar al de nuestro ejemplo en nuestro servidor, el cliente sólo recibiría el resultado de su ejecución en el servidor, sin ninguna posibilidad de determinar que código ha producido el resultado recibido. El servidor web puede ser incluso configurado para que procese todos los ficheros HTML con PHP.

---

[Anterior](#)

Conceptos Básicos

[Inicio](#)

[Subir](#)

[Siguiete](#)

Qué se puede hacer con PHP?

# Capítulo 2. Instalación

## Tabla de contenidos

[Bajándose la última versión](#)

[Instalación en sistemas UNIX](#)

[Instalación en sistemas Windows 95/98/NT](#)

[¿Problemas?](#)

## Bajándose la última versión

El código fuente y las distribuciones binarias para algunas plataformas (incluído Windows) se pueden encontrar en <http://www.php.net/>.

# Capítulo 3. Configuración

## Tabla de contenidos

[El archivo de configuración](#)

## El archivo de configuración

El archivo de configuración (llamado `php3.ini` en PHP 3.0, y simplemente `php.ini` a partir del PHP 4.0) es leído cuando arranca el PHP. Para las versiones de PHP como módulo de servidor esto sólo ocurre una vez al arrancar el servidor web. Para la versión CGI, esto ocurre en cada llamada.

Cuando se utiliza PHP como módulo Apache, también puede cambiar los ajustes de configuración utilizando directivas en los archivos de configuración del Apache y en los `.htaccess`.

Con el PHP 3.0 hay directivas Apache que se corresponden a cada uno de los ajustes de configuración del `php3.ini`, con la excepción que su nombre va precedido de "php3\_".

Con el PHP 4.0 sólo hay unas pocas directivas de Apache que le permiten cambiar los ajustes de configuración del PHP.

`php_value nombre valor`

Fija el valor de la variable especificada.

`php_flag nombre on/off`

Fija una opción de configuración de tipo Boolean.

`php_admin_value nombre valor`

Fija el valor de la variable especificada. Los ajustes de configuración de tipo "Admin" sólo se pueden fijar desde los archivos principales de configuración del Apache, y no desde los `.htaccess`.

`php_admin_flag nombre on/off`

Fija una opción de configuración de tipo Boolean.

Puede ver los ajustes de los valores de configuración en la salida de [phpinfo\(\)](#). También puede acceder a los valores individuales de los ajustes de configuración utilizando [get\\_cfg\\_var\(\)](#).

## Directivas Generales de Configuración

`asp_tags` boolean

Permite el uso de las etiquetas estilo ASP `<% %>` además de las habituales etiquetas `<?php ?>`. También se incluye el atajo para imprimir variables `<%= $valor %>`. Para más información, vea [Escapando del HTML](#).

**Nota:** El soporte para etiquetas estilo ASP se añadió en la 3.0.4.

`auto_append_file` [string](#)

Especifica el nombre de un archivo que es troceado automáticamente después del archivo principal. El archivo se incluye como si fuese llamado mediante la función [include\(\)](#), así que se utiliza [include\\_path](#).

El valor especial `none` desconecta la adición automática de archivos.

**Nota:** Si el script es terminado con [exit\(\)](#), *no* tendrá lugar la adición automática.

`auto_prepend_file` [string](#)

Especifica el nombre de un archivo que es troceado automáticamente antes del archivo principal. Specifies the name of a file that is automatically parsed before the main file. El archivo se incluye como si fuese llamado mediante la función [include\(\)](#), así que se utiliza [include\\_path](#).

El valor especial `none` desconecta la adición automática de archivos.

*cgi\_ext* [string](#)

*display\_errors* boolean

Determina si los errores se visualizan en pantalla como parte de la salida en HTML o no.

*doc\_root* [string](#)

"Directorio raíz" del PHP en el servidor. Sólo se usa si no está vacío. Si el PHP se configura con [safe mode](#), no se sirven archivos fuera de este directorio.

*engine* boolean

Esta directiva sólo es realmente útil en la versión de PHP como módulo Apache. Se utiliza por sitios que desean habilitar la ejecución del PHP directorio por directorio o en base a cada servidor virtual. Poniendo **php3\_engine off** en los sitios apropiados del archivo `httpd.conf`, se puede habilitar o deshabilitar el PHP.

*error\_log* [string](#)

Nombre del fichero para registrar los errores de un script. Si se utiliza el valor especial `syslog`, los errores se envían al registro de errores del sistema. En UNIX se refiere a `syslog(3)` y en Windows NT al registro de eventos. El registro de errores del sistema no es soportado bajo Windows 95.

*error\_reporting* [integer](#)

Fija el nivel de informe de errores. El parámetro es un entero que representa un campo de bits. Suma los valores de los niveles de informe de error que desea.

**Tabla 3-1. Niveles de Informe de Errores**

valor de bit	informe habilitado
1	errores normales
2	avisos normales
4	errores del troceador (parser)
8	avisos de estilo no críticos

El valor por defecto para esta directiva es 7 (se muestran los errores normales, avisos normales y errores de parser).

## *open\_basedir* [string](#)

Limita los archivos que se pueden abrir por el PHP al árbol de directorios especificado.

Cuando un script intenta abrir un archivo con, por ejemplo, `fopen` o `gzopen`, se comprueba su localización. Si el fichero está fuera del árbol de directorios especificado, PHP se negará a abrirlo. Todos los enlaces simbólicos son resueltos, de modo que no es posible evitar esta limitación usando uno de ellos.

El valor especial `.` indica que el directorio base será aquel en el que reside el script.

Bajo Windows, separe los directorios mediante punto y coma. En el resto de sistemas, sepárelos con dos puntos `:`. Como módulo de Apache, los senderos para `open_basedir` de los directorios padre se heredan ahora automáticamente.

**Nota:** El soporte para directorios múltiples se añadió en la 3.0.7.

El valor por defecto es permitir abrir todos los archivos.

## *gpc\_order* [string](#)

Fija el orden de troceo de variables GET/POST/COOKIE. El valor por defecto de esta directiva es "GPC". Fijándola, por ejemplo, a "GP", hará que el PHP ignore por completo las cookies y que sobrescriba las variables recibidas por GET con las que tengan el mismo nombre y vengan por POST.

## *ignore\_user\_abort* [string](#)

Por defecto está a `on`. Si se cambia a `off`, los script terminarán tan pronto como intenten enviar algo después de que un cliente ha roto la conexión.

[ignore\\_user\\_abort\(\)](#).

## *include\_path* [string](#)

Especifica una lista de directorios en los que las funciones [require\(\)](#), [include\(\)](#) y [fopen\\_with\\_path\(\)](#) buscan los archivos. El formato es similar a la variable de entorno de sistema `PATH`: una lista de directorios separados por dos puntos en UNIX o por punto y coma en Windows.



### Ejemplo 3-1. `include_path` en UNIX

```
include_path=.: /home/httpd/php-lib
```

### Ejemplo 3-2. `include_path` en Windows

```
include_path=".;c:\www\phplib"
```

El valor por defecto para esta directiva es `.` (sólo el directorio actual).

`isapi_ext` [string](#)

`log_errors` boolean

Dice si los mensajes de error de los script deben ser registrados o no en el registro del servidor. Esta opción, por tanto, es específica del mismo.

`magic_quotes_gpc` boolean

Fija el estado `magic_quotes` para operaciones GPC (Get/Post/Cookie). Si `magic_quotes` vale `on`, todas las `'` (comilla sencilla), `"` (comilla doble), `\` (barra invertida) y los NUL son automáticamente marcados con una barra invertida. Si además `magic_quotes_sybase` vale `on`, la comilla sencilla es marcada con otra comilla sencilla en lugar de la barra invertida.

`magic_quotes_runtime` boolean

Si se habilita `magic_quotes_runtime`, muchas de las funciones que devuelven datos de algún tipo de fuente externa incluyendo bases de datos y archivos de texto devolverán las comillas marcadas con una barra invertida. Si también está activo `magic_quotes_sybase`, la comilla simple es marcada con una comilla simple en lugar de la barra invertida.

`magic_quotes_sybase` boolean

Si `magic_quotes_sybase` está a `on`, la comilla simple es marcada con una comilla simple en lugar de la barra invertida cuando están habilitados `magic_quotes_gpc` o `magic_quotes_runtime`.

`max_execution_time` [integer](#)

Fija el tiempo máximo en segundos que se le permite usar a un script antes de ser

finalizado por el intérprete. Así se evita que scripts mal escritos puedan bloquear el servidor.

*memory\_limit* [integer](#)

Fija el tamaño máximo de memoria en bytes que se permite reclamar a un script. Así se evita que script mal escritos se coman toda la memoria disponible de un servidor.

*nsapi\_ext* [string](#)

*short\_open\_tag* boolean

Indica si se debe permitir el formato corto (<? ?>) de la etiqueta de apertura del PHP. Si desea utilizar PHP en combinación con XML, deberá desactivar esta opción. Si está desactivada, deberá utilizar el formato largo de la etiqueta de apertura (<?php ?>).

*sql.safe\_mode* boolean

*track\_errors* boolean

Si está habilitada, el último mensaje de error estará siempre presente en la variable global `$php_errormsg`.

*track\_vars* boolean

Si está activada, la información de entrada de GET, POST y de las cookies se puede encontrar en las matrices asociativas `$HTTP_GET_VARS`, `$HTTP_POST_VARS` y `$HTTP_COOKIE_VARS` respectivamente.

*upload\_tmp\_dir* [string](#)

El directorio temporal utilizado para almacenar archivos cuando se envían al servidor. Debe tener permiso de escritura para el usuario bajo el que corra el PHP.

*user\_dir* [string](#)

El nombre base del directorio utilizado bajo el directorio inicial de un usuario para los archivos PHP. Por ejemplo: `paginas_html`.

*warn\_plus\_overloading* boolean

Si está activada, esta opción hace que el PHP muestre un aviso cuando el operador suma (+) se utiliza en cadenas. Así es más fácil encontrar scripts que necesitan ser reescritos utilizando en su lugar el concatenador de cadenas (.

# Directivas de Configuración de Correo

*SMTP* [string](#)

Nombre DNS o dirección IP del servidor de SMTP que el PHP bajo Windows deberá usar para enviar correo con la función [mail\(\)](#).

*sendmail\_from* [string](#)

La dirección del remitente ("De : ") para los correos enviados desde PHP bajo Windows.

*sendmail\_path* [string](#)

Localización del programa **sendmail**. Generalmente `/usr/sbin/sendmail` o `/usr/lib/sendmail`. **configure** intenta localizarle este archivo lo mejor que puede y fijar un valor por defecto, pero en caso de fallo, lo puede usted fijar aquí.

Los sistemas que no usan sendmail deberán fijar esta directiva al nombre del programa alternativo que ofrezca su sistema de correo, si es que existe. Por ejemplo, los usuarios del [Qmail](#) pueden fijarlo normalmente a `/var/qmail/bin/sendmail`.

# Directivas de Configuración de Modo Seguro

*safe\_mode* boolean

Para activar el modo seguro del PHP. Lea el [Capítulo de seguridad](#) para más información.

*safe\_mode\_exec\_dir* [string](#)

Si el PHP se utiliza en modo seguro, la función [system\(\)](#) y el resto de funciones que ejecutan programas del sistema se niegan a ejecutar programas que no estén en este directorio.

# Directivas de Configuración del Debugger

*debugger.host* [string](#)

Nombre DNS o dirección IP del servidor usado por el debugger.

*debugger.port* [string](#)

Número de puerto usado por el debugger.

*debugger.enabled* boolean

Indica si el debugger está habilitado o no.

## Directivas de Carga de Extensiones

*enable\_dl* boolean

Esta directiva sólo es útil en la versión del PHP como módulo del Apache. Puede habilitar o deshabilitar para un servidor virtual o para un directorio la carga dinámica de extensiones de PHP mediante **dl()**.

La razón principal para deshabilitar la carga dinámica es la seguridad. Con la carga dinámica es posible ignorar las restricciones `safe_mode` y `open_basedir`.

El valor por defecto es permitir la carga dinámica, excepto cuando se usa el modo seguro. En modo seguro, siempre es imposible usar **dl()**.

*extension\_dir* [string](#)

En qué directorio debe buscar el PHP las extensiones cargables dinámicamente.

*extension* [string](#)

Qué extensiones dinámicas debe cargar el PHP cuando arranca.

## Directivas de Configuración de MySQL

*mysql.allow\_persistent* boolean

Si permitir o no conexiones MySQL persistentes.

*mysql.default\_host* [string](#)

El servidor por defecto para utilizar cuando se conecte al servidor de bases de datos si no se especifica otro distinto.

`mysql.default_user` [string](#)

El nombre de usuario por defecto para utilizar cuando se conecta al servidor de base de datos si no se especifica otro.

`mysql.default_password` [string](#)

La clave por defecto para utilizar cuando se conecta al servidor de base de datos si no se especifica otro.

`mysql.max_persistent` [integer](#)

El número máximo de conexiones persistentes de MySQL por proceso.

`mysql.max_links` [integer](#)

El número máximo de conexiones de MySQL por proceso, incluyendo las persistentes.

## Directivas de Configuración de mSQL

`mssql.allow_persistent` boolean

Si se permiten o no conexiones persistentes de mSQL.

`mssql.max_persistent` [integer](#)

El número máximo de conexiones persistentes mSQL por proceso.

`mssql.max_links` [integer](#)

El número máximo de conexiones de mSQL por proceso, incluyendo las persistentes.

## Directivas de Configuración de Postgres

`pgsql.allow_persistent` boolean

Si se permiten o no conexiones persistentes de Postgres.

`pgsql.max_persistent` [integer](#)

El número máximo de conexiones persistentes Postgres por proceso.

`pgsql.max_links` [integer](#)

El número máximo de conexiones de Postgres por proceso, incluyendo las persistentes.

## SESAM Configuration Directives

*sesam\_oml* [string](#)

Name of BS2000 PLAM library containing the loadable SESAM driver modules. Required for using SESAM functions. The BS2000 PLAM library must be set ACCESS=READ,SHARE=YES because it must be readable by the apache server's user id.

*sesam\_configfile* [string](#)

Name of SESAM application configuration file. Required for using SESAM functions. The BS2000 file must be readable by the apache server's user id.

The application configuration file will usually contain a configuration like (see SESAM reference manual):

```
CNF=B  
NAM=K  
NOTYPE
```

*sesam\_messagecatalog* [string](#)

Name of SESAM message catalog file. In most cases, this directive is not necessary. Only if the SESAM message file is not installed in the system's BS2000 message file table, it can be set with this directive.

The message catalog must be set ACCESS=READ,SHARE=YES because it must be readable by the apache server's user id.

## Directivas de Configuración de Sybase

*sybase.allow\_persistent* boolean

Si se permiten o no conexiones persistentes de Sybase.

`sybase.max_persistent` [integer](#)

El número máximo de conexiones persistentes Sybase por proceso.

`sybase.max_links` [integer](#)

El número máximo de conexiones de Sybase por proceso, incluyendo las persistentes.

## Directivas de Configuración de Sybase-CT

`sybct.allow_persistent` boolean

Si se permiten o no conexiones persistentes de Sybase-CT. El valor por defecto es on.

`sybct.max_persistent` [integer](#)

El número máximo de conexiones persistentes Sybase-CT por proceso. El valor por defecto es -1, que significa ilimitadas.

`sybct.max_links` [integer](#)

El número máximo de conexiones de Sybase-CT por proceso, incluyendo las persistentes. El valor por defecto es -1, que significa ilimitadas.

`sybct.min_server_severity` [integer](#)

Los mensajes de servidor con gravedad mayor o igual que

`sybct.min_server_severity` serán reportados como avisos. Este valor también se puede cambiar desde un script usando la función

**sybase\_min\_server\_severity()**. El valor por defecto es 10, que reporta los errores de información con gravedad o mayores.

`sybct.min_client_severity` [integer](#)

Los mensajes de librería de cliente con gravedad mayor o igual que

`sybct.min_client_severity` serán reportados como avisos. Este valor también se puede cambiar desde un script usando la función

**sybase\_min\_client\_severity()**. El valor por defecto es 10, que desconecta los avisos.

`sybct.login_timeout` [integer](#)

El número máximo de segundos de espera por un intento de conexión con éxito antes

de indicar un fallo. Nótese que si se ha excedido *max\_execution\_time* cuando finaliza la espera de un intento de conexión, el script será finalizado antes de que se pueda tomar una acción en caso de fallo. El valor por defecto es 1 minuto.

*sybct.timeout* [integer](#)

El número máximo de segundos de espera por una operación de consulta o `select_db` con éxito antes de indicar un fallo. Nótese que si se ha excedido *max\_execution\_time* cuando finaliza la espera de un intento de conexión, el script será finalizado antes de que se pueda tomar una acción en caso de fallo. El valor por defecto es sin límite.

*sybct.hostname* [string](#)

El nombre de la máquina desde la que dice estarse conectando, para que se visualice con `sp_who()`. El valor por defecto es "none".

## Directivas de Configuración de Informix

*ifx.allow\_persistent* boolean

Si se permiten o no conexiones persistentes de Informix.

*ifx.max\_persistent* [integer](#)

El número máximo de conexiones persistentes de Informix por proceso.

*ifx.max\_links* [integer](#)

El número máximo de conexiones Informix por proceso, incluyendo las persistentes.

*ifx.default\_host* [string](#)

El servidor por defecto al que conectarse si no se especifica uno en [ifx\\_connect\(\)](#) o en [ifx\\_pconnect\(\)](#).

*ifx.default\_user* [string](#)

El id de usuario por defecto para utilizar si no se especifica uno en [ifx\\_connect\(\)](#) o en [ifx\\_pconnect\(\)](#).

*ifx.default\_password* [string](#)

La clave por defecto para utilizar si no se especifica uno en [ifx\\_connect\(\)](#) o en



### [ifx\\_pconnect\(\)](#).

*ifx.blobinfile* boolean

Fíjelo a **TRUE** si desea recibir las columnas blob (objetos binarios grandes) en un archivo, y a **FALSE** si las desea en memoria. Puede cambiar el ajuste en tiempo de ejecución utilizando [ifx\\_blobinfile\\_mode\(\)](#).

*ifx.textasvarchar* boolean

Fíjelo a **TRUE** si desea recibir las columnas TEXT como cadenas normales en las instrucciones select, y a **FALSE** si quiere usar parámetros de identificador de blobs. Puede cambiar el ajuste en tiempo de ejecución utilizando [ifx\\_textasvarchar\(\)](#).

*ifx.byteasvarchar* boolean

Fíjelo a **TRUE** si desea devolver las columnas BYTE como cadenas normales en las instrucciones select, y a **FALSE** si quiere usar parámetros de identificador de blobs. Puede cambiar el ajuste en tiempo de ejecución utilizando [ifx\\_byteasvarchar\(\)](#).

*ifx.charasvarchar* boolean

Fíjelo a **TRUE** si desea suprimir los espacios a la derecha de las columnas CHAR cuando las solicita.

*ifx.nullformat* boolean

Fíjelo a **TRUE** si desea que las columnas **NULL** (nulas) se devuelvan como la cadena literal "NULL", y a **FALSE** si desea que se devuelvan como la cadena vacía "". Puede cambiar el ajuste en tiempo de ejecución utilizando [ifx\\_nullformat\(\)](#).

## Directivas de Configuración de Matemática BC

*bcmath.scale* [integer](#)

Número de dígitos decimales para todas las funciones de bcmath.

## Directivas de Configuración de Capacidades de los Navegadores

*browscap* [string](#)

Nombre del archivo de capacidades del navegador. Vea también [get\\_browser\(\)](#).

## Directivas Unificadas de Configuración de ODBC

*uodbc.default\_db* [string](#)

Fuentes de datos ODBC a utilizar si no se especifica una en [odbc\\_connect\(\)](#) o en [odbc\\_pconnect\(\)](#).

*uodbc.default\_user* [string](#)

Nombre de usuario si no se especifica uno en [odbc\\_connect\(\)](#) o en [odbc\\_pconnect\(\)](#).

*uodbc.default\_pw* [string](#)

Clave para usar si no se especifica una en [odbc\\_connect\(\)](#) o en [odbc\\_pconnect\(\)](#).

*uodbc.allow\_persistent* boolean

Si se permiten o no conexiones persistentes de ODBC.

*uodbc.max\_persistent* [integer](#)

El número máximo de conexiones persistentes de ODBC por proceso.

*uodbc.max\_links* [integer](#)

El número máximo de conexiones ODBC por proceso, incluyendo las persistentes.

---

[Anterior](#)

[¿Problemas?](#)

[Inicio](#)

[Subir](#)

[Siguiete](#)

[Seguridad](#)

# Capítulo 4. Seguridad

## Tabla de contenidos

[Binarios CGI](#)

[Modulo Apache](#)

PHP es un potente lenguaje y el interprete, tanto incluido en el servidor web como modulo o ejecutado como un binario CGI, puede acceder a ficheros, ejecutar comandos y abrir comunicaciones de red en el servidor. Todas estas características hacen que lo que se ejecute en el servidor web sea inseguro por defecto. PHP ha sido diseñado específicamente, para ser un lenguaje mas seguro para escribir programas CGI, que Perl o C y con la correcta seleccion de las opciones de configuración del tiempo de compilación y ejecucion se consigue la exacta combinación de libertad y seguridad que se necesita.

Ya que existen diferentes modos de utilizar PHP, existen multitud de opciones de configuración que permiten controlar su funcionamiento. Una gran selección de opciones garantiza que se pueda usar PHP para diferentes usos, pero tambien significa que existen combinaciones de estas opciones y configuraciones del servidor que producen instalaciones inseguras. Este capitulo explica las diferentes combinaciones de opciones de configuración y las situaciones donde pueden ser usadas de manera segura.

## Binarios CGI

### Posibles ataques

Usando PHP como un binario CGI es una opción para instalaciones que por cualquier causa no quieren integrar PHP como modulo en el software servidor (p.ej: Apache), o usaran PHP con diferentes clases de CGI wrappers para crear entornos chroot y setuid seguros para los scripts. Esta configuración implica generalmente el instalar el binario ejecutable de PHP en el directorio cgi-bin del servidor web. El documento del CERT [CA-96.11](#) recomienda no instalar interpretes en cgi-bin. Aunque el binario PHP puede ser usado como interprete independiente, PHP esta diseñado para prevenir los ataques que esta configuración hace posible.

- Accediendo a ficheros del sistema: `http://my.host/cgi-bin/php?/etc/passwd`

La información introducida después del signo de interrogación (?) es transferida como argumento de la línea de comando al intérprete por el interfaz del CGI. Normalmente los intérpretes abren y ejecutan el fichero especificado como el primer argumento en la línea de comando.

Cuando se ejecuta como un CGI script, PHP rechaza interpretar los argumentos de la línea de comando.

- Accediendo cualquier documento web en el servidor: `http://my.host/cgi-bin/php/secret/doc.html`

La información con el camino (path) de la URL después del nombre del binario PHP, `/secret/doc.html` es usada convencionalmente para especificar el nombre del fichero que será abierto e interpretado por el programa CGI. Normalmente, algunas directivas del servidor web (Apache:Action) son usadas para redireccionar peticiones de documentos como `http://my.host/secret/script.php3` al intérprete PHP. Con esta configuración, el servidor web comprueba primero los permisos de acceso al directorio `/secret`, y después crea la petición redireccionada `http://my.host/cgi-bin/php/secret/script.php3`.

Desafortunadamente, si la petición es hecha de esta forma en un principio, el servidor web no comprueba los permisos de acceso del fichero `/secret/script.php3`, sino solamente del fichero `/cgi-bin/php`. De esta manera cualquier usuario que pueda acceder `/cgi-bin/php` también puede acceder a cualquier documento protegido en el servidor web.

En PHP, a la hora de compilar, la opción de configuración [--enable-force-cgi-redirect](#) y las directivas de configuración a la hora de ejecutar [doc\\_root](#) y [user\\_dir](#) pueden ser usadas para prevenir este ataque, si el árbol de documentos del servidor tiene cualquier directorio con acceso restringido. Ver más adelante la explicación de las diferentes combinaciones.

## Caso 1: solamente se sirven ficheros públicos

Si tu servidor no contiene información que este protegida con clave o acceso de control de

IPs, no se necesitan estas opciones de configuracion. Si tu servidor web no permite realizar redireccionamientos, o el servidor no tiene modo de comunicar al binario PHP que la peticion es una peticion segura redireccionada, podeis especificar la opcion [--disable-force-cgi-redirect](#) en el script de configuracion. De todas maneras, teneis que asegurarnos que vuestros scripts PHP no confíen en la manera al llamar al script, ni de forma directa `http://my.host/cgi-bin/php/dir/script.php3` o por redireccion `http://my.host/dir/script.php3`.

Redireccionamiento puede ser configurado en Apache usando las directivas `AddHandler` y `Action` (ver mas abajo).

## Caso 2: usando `--enable-force-cgi-redirect`

Esta opcion a la hora de compilar previene que alguien llame PHP directamente con una url como la siguiente `http://my.host/cgi-bin/php/secret_dir/script.php3`. PHP solamente analizara en este modo si ha pasado por una regla de redireccionamiento en el servidor.

Normalmente la redireccion en la configuracion de Apache es hecha con la siguientes directivas:

```
Action php3-script /cgi-bin/php
AddHandler php3-script .php3
```

Esta opcion ha sido solo comprobada con el servidor web Apache, y depende de Apache para fijar la variable de entorno CGI no estandar `REDIRECT_STATUS` en las peticiones de redireccionamiento. Si tu servidor web no soporta ningun modo para informar si una peticion es directa o redireccionada, no podeis usar esta opcion y debereis usar alguno de los otros modos de ejecucion de la version CGI documentados aqui.

## Caso 3: Usando `doc_root` or `user_dir`

Incluir contenidos activos, como script y ejecutables, en el directorio de documentos del servidor web, es algunas veces considerada una practica insegura. Si por algun fallo de configuracion, los scripts no son ejecutados pero mostrados como documentos HTML, cualquiera podra conseguir codigo registrado o informacion de seguridad, como p.ej: claves

de acceso. Por ello, muchos administradores prefieren utilizar otra estructura de directorios que contenga solamente los scripts, los cuales seran solamente accesibles via PHP CGI, y por ello siempre seran interpretados y no mostrados.

Habra que tener en cuenta que si el metodo que asegura que las peticiones no son redireccionadas, como hemos descrito en la seccion anterior, no esta disponible, sera necesario configurar un script `doc_root` que sea diferente del "web document root".

Podeis definir el script PHP "document root" con la directiva de configuracion [doc\\_root](#) en el [fichero de configuracion](#), o definir la variable de entorno `PHP_DOCUMENT_ROOT`. Si esta definida, la version CGI de PHP siempre obtendra el nombre del fichero a abrir con `doc_root` y el camino (path) utilizado en la peticion, asi podeis estar seguros que ningun script sera ejecutado fuera de este directorio (excepto para `user_dir`, ver a continuacion)

Otra opcion que se puede usar aqui es [user\\_dir](#). Cuando `user_dir` no esta definido, lo unico que controla la apertura del fichero es `doc_root`. Si intentamos abrir una url tal como esta `http://my.host/~user/doc.php3` no se abra un fichero en el directorio de usuarios, en su lugar se abra un fichero llamado `~user/doc.php3` en el directorio `doc_root`. (si, un directorio que empieza por tilde [~]).

Si `user_dir` esta definido por ejemplo como `public_php`, una peticion tal como `http://my.host/~user/doc.php3`, abra un fichero llamado `doc.php3` en el directorio llamado `public_php` del directorio "home" del usuario. Si el directorio del usuario es `/home/user`, el fichero ejecutado sera `/home/user/public_php/doc.php3`.

La expansion de `user_dir` ocurre sin tener en cuenta la configuracion de `doc_root`, de este modo se puede controlar los accesos al directorio principal (document root) y al directorio de usuario separadamente.

## Caso 4: Analizador PHP fuera del arbol web.

Una opcion muy segura es poner el analizador binario PHP, en algun lugar fuera del arbol de ficheros web. Por ejemplo en `/usr/local/bin`. La unica pega real de esta opcion es que habra que poner una linea similar a:

```
#!/usr/local/bin/php
```

como primera linea en cualquier fichero que contenga codigo PHP. Tambien sera necesario asignar al fichero permisos de ejecucion. De esta manera, es tratado de la misma manera que cualquier otro CGI script escrito en Perl o sh o otro lenguaje utilizado para scripts y que utilicen el mecanismo #! para ejecutarse.

Para conseguir que PHP maneje correctamente con esta configuracion, la informacion de PATH\_INFO y PATH\_TRANSLATED, el analizador PHP deberia ser compilado con la opcion de configuracion [--enable-discard-path](#).

---

[Anterior](#)

Configuración

[Inicio](#)

[Subir](#)

[Siguiete](#)

Modulo Apache

## II. Referencia del Lenguaje

### Tabla de contenidos

5. [Sintaxis básica](#)
6. [Types](#)
7. [Variables](#)
8. [Constantes](#)
9. [Expresiones](#)
10. [Operadores](#)
11. [Estructuras de Control](#)
12. [Funciones](#)
13. [Clases y Objetos](#)
14. [References Explained](#)



---

# Capítulo 5. Sintaxis básica

## Tabla de contenidos

[Saliendo de HTML](#)

[Separación de instrucciones](#)

[Comentarios](#)

## Saliendo de HTML

Hay cuatro formas de salir de HTML y entrar en el "modo de código PHP":

### Ejemplo 5-1. Formas de salir de HTML

```
1. <? echo ("esta es la más simple, una instrucción de procesado SGML\n"); ?>
2. <?php echo("si quiere servir documentos XML, haga esto\n"); ?>
3. <script language="php">
    echo ("a algunos editores (como FrontPage) no les
        gustan las intrucciones de procesado");
</script>
4. <% echo ("Puedes también usar etiquetas tipo ASP"); %>
   <%= $variable; # Esto es una forma abreviada de "<%echo .." %>
```

La primera forma sólo está disponible si se han habilitado las etiquetas cortas. Esto se puede hacer a través de la función `short_tags()`, habilitando la opción de configuración [short\\_open\\_tag](#) en el archivo de configuración de PHP, o compilando PHP con la opción `--enable-short-tags` en **configure**.

La cuarta manera está disponible sólo si se han habilitado las etiquetas tipo ASP usando la opción de configuración [asp\\_tags](#).

**Nota:** El soporte para las etiquetas tipo ASP se añadió en 3.0.4.

La etiqueta de cierre de un bloque incluirá el carácter de nueva línea final si hay uno presente.

# Capítulo 6. Types

## Tabla de contenidos

[Enteros](#)

[Números en punto flotante](#)

[Cadenas](#)

[Arrays](#)

[Objetos](#)

[Type juggling](#)

PHP soporta los siguientes tipos:

- [array](#)
- [números en punto flotante](#)
- [entero](#)
- [objeto](#)
- [cadena](#)

El tipo de una variable normalmente no lo indica el programador; en su lugar, lo decide PHP en tiempo de ejecución dependiendo del contexto en el que se utilice esa variable.

Si se quisiese obligar a que una variable se convierta a un tipo concreto, se podría [forzar](#) la variable o usar la función [settype\(\)](#) para ello.

Nótese que una variable se puede comportar de formas diferentes en ciertas situaciones, dependiendo de qué tipo sea en ese momento. Para más información, vea la sección [Conversión de Tipos](#).

## Enteros

Los enteros se puede especificar usando una de las siguientes sintaxis:

```
$a = 1234; # número decimal  
$a = -123; # un número negativo  
$a = 0123; # número octal (equivalente al 83 decimal)  
$a = 0x12; # número hexadecimal (equivalente al 18 decimal)
```

---

[Anterior](#)

Comentarios

[Inicio](#)

[Subir](#)

[Siguiete](#)

Números en punto flotante

# Capítulo 7. Variables

## Tabla de contenidos

[Conceptos Básicos](#)

[Variables predefinidas](#)

[Ambito de las variables](#)

[Variables variables](#)

[Variables externas a PHP](#)

## Conceptos Básicos

En PHP las variables se representan como un signo de dólar seguido por el nombre de la variable. El nombre de la variable es sensible a minúsculas y mayúsculas.

```
$var = "Bob" ;  
$Var = "Joe" ;  
echo "$var, $Var"; // produce la salida "Bob, Joe"
```

En PHP3, las variables siempre se asignan por valor. Esto significa que cuando se asigna una expresión a una variable, el valor íntegro de la expresión original se copia en la variable de destino. Esto quiere decir que, por ejemplo, después de asignar el valor de una variable a otra, los cambios que se efectúen a una de esas variables no afectará a la otra. Para más información sobre este tipo de asignación, vea [Expresiones](#).

PHP4 ofrece otra forma de asignar valores a las variables: *asignar por referencia*. Esto significa que la nueva variable simplemente referencia (en otras palabras, "se convierte en un alias de" o "apunta a") la variable original. Los cambios a la nueva variable afectan a la original, y viceversa. Esto también significa que no se produce una copia de valores; por tanto, la asignación ocurre más rápidamente. De cualquier forma, cualquier incremento de velocidad se notará sólo en los bucles críticos cuando se asignen grandes arrays u objetos.

Para asignar por referencia, simplemente se antepone un ampersand (&) al comienzo de la variable cuyo valor se está asignando (la variable fuente). Por ejemplo, el siguiente trozo de código produce la salida 'Mi nombre es Bob' dos veces:

```
<?php
$foo = 'Bob';           // Asigna el valor 'Bob' a $foo
$bar = &$foo;          // Referencia $foo vía $bar.
$bar = "Mi nombre es $bar"; // Modifica $bar...
echo $foo;             // $foo también se modifica.
echo $bar;
?>
```

Algo importante a tener en cuenta es que sólo las variables con nombre pueden ser asignadas por referencia.

```
<?php
$foo = 25;
$bar = &$foo;          // Esta es una asignación válida.
$bar = &(amp;24 * 7); // Inválida; referencia una expresión sin nombre.

function test() {
    return 25;
}

$bar = &test();        // Inválida.
?>
```

---

[Anterior](#)

Type juggling

[Inicio](#)

[Subir](#)

[Siguiete](#)

Variables predefinidas

## Capítulo 8. Constantes

PHP define varias constantes y proporciona un mecanismo para definir más en tiempo de ejecución. Las constantes son como las variables, salvo por las dos circunstancias de que las constantes deben ser definidas usando la función [define\(\)](#), y que no pueden ser redefinidas más tarde con otro valor.

Las constantes predefinidas (siempre disponibles) son:

`__FILE__`

El nombre del archivo de comandos que está siendo interpretado actualmente. Si se usa dentro de un archivo que ha sido incluido o requerido, entonces se da el nombre del archivo incluido, y no el nombre del archivo padre.

`__LINE__`

El número de línea dentro del archivo que está siendo interpretado en la actualidad. Si se usa dentro de un archivo incluido o requerido, entonces se da la posición dentro del archivo incluido.

`PHP_VERSION`

La cadena que representa la versión del analizador de PHP en uso en la actualidad; e.g. '3.0.8-dev'.

`PHP_OS`

El nombre del sistema operativo en el cuál se ejecuta el analizador PHP; e.g. 'Linux'.

**TRUE**

Valor verdadero.

**FALSE**

Valor falso.

`E_ERROR`

Denota un error distinto de un error de interpretación del cual no es posible recuperarse.

`E_WARNING`

Denota una condición donde PHP reconoce que hay algo erróneo, pero continuará de todas

formas; pueden ser capturados por el propio archivo de comandos. Un ejemplo sería una inválida regexp en [ereg\(\)](#).

## E\_PARSE

El interprete encontró sintaxis inválida en el archivo de comandos. La recuperación no es posible.

## E\_NOTICE

Ocurrió algo que pudo ser o no un error. La ejecución continúa. Los ejemplos incluyen usar una cadena sin comillas como un índice "hash", o acceder a una variable que no ha sido inicializada.

Las constantes E\_\* se usan típicamente con la función [error\\_reporting\(\)](#) para configurar el nivel de informes de error.

Se pueden definir constantes adicionales usando la función [define\(\)](#).

Nótese que son constantes, no macros tipo C; con una constante sólo se pueden representar datos escalares válidos.

### Ejemplo 8-1. Definiendo Constantes

```
<?php
define("CONSTANTE", "Hola mundo.");
echo CONSTANTE; // muestra "Hola mundo."
?>
```

### Ejemplo 8-2. Usando \_\_FILE\_\_ y \_\_LINE\_\_

```
<?php
function report_error($file, $line, $message) {
    echo "Un error ocurrió en $file en la línea $line: $message.";
}

report_error(__FILE__, __LINE__, "Algo fue mal!");
?>
```

Variables externas a PHP

[Subir](#)

Expresiones



## Capítulo 9. Expresiones

Las expresiones son la piedra angular de PHP. En PHP, casi cualquier cosa que escribes es una expresión. La forma más simple y ajustada de definir una expresión es "cualquier cosa que tiene un valor".

Las formas más básicas de expresiones son las constantes y las variables. Cuando escribes "\$a = 5", estás asignando '5' a \$a. '5', obviamente, tiene el valor 5 o, en otras palabras '5' es una expresión con el valor 5 (en este caso, '5' es una constante entera).

Después de esta asignación, esperarás que el valor de \$a sea 5 también, de manera que si escribes \$b = \$a, esperas que se comporte igual que si escribieses \$b = 5. En otras palabras, \$a es una expresión también con el valor 5. Si todo va bien, eso es exactamente lo que pasará.

Las funciones son un ejemplo algo más complejo de expresiones. Por ejemplo, considera la siguiente función:

```
function foo () {  
    return 5;  
}
```

Suponiendo que estés familiarizado con el concepto de funciones (si no lo estás échale un vistazo al capítulo sobre funciones), asumirás que teclear \$c = foo() es esencialmente lo mismo que escribir \$c = 5, y has acertado. Las funciones son expresiones que valen el valor que retornan. Como foo() devuelve 5, el valor de la expresión 'foo()' es 5. Normalmente las funciones no devuelven un valor fijo, sino que suele ser calculado.

Desde luego, los valores en PHP no se limitan a enteros, y lo más normal es que no lo sean. PHP soporta tres tipos escalares: enteros, punto flotante y cadenas (los tipos escalares son aquellos cuyos valores no pueden 'dividirse' en partes menores, no como los arrays, por ejemplo). PHP también soporta dos tipos compuestos (no escalares): arrays y objetos. Se puede asignar cada uno de estos tipos de valor a variables o bien retornarse de funciones, sin ningún tipo de limitación.

Hasta aquí, los usuarios de PHP/FI 2 no deberían haber notado ningún cambio. Sin embargo, PHP lleva las expresiones mucho más allá, al igual que otros lenguajes. PHP es un lenguaje orientado a expresiones, en el sentido de que casi todo es una expresión. Considera el ejemplo anterior '\$a = 5'. Es sencillo ver que hay dos valores involucrados, el valor de la constante entera '5', y el valor de \$a que está siendo actualizado también a 5. Pero la verdad es que hay un valor adicional implicado aquí, y es el valor de la propia asignación. La asignación misma se evalúa al valor asignado, en este caso 5. En la práctica, quiere decir que '\$a = 5', independientemente de lo que hace, es una expresión con el valor 5. De esta manera, escribir algo como '\$b = (\$a = 5)' es como escribir '\$a = 5; \$b = 5;' (un punto y coma marca el final de una instrucción). Como las asignaciones se evalúan de derecha a izquierda, puedes escribir también '\$b = \$a = 5'.

Otro buen ejemplo de orientación a expresiones es el pre y post incremento y decremento. Los usuarios de PHP/FI 2 y los de otros muchos lenguajes les sonará la notación variable++ y variable--. Esto son las operaciones de incremento y decremento. En PHP/FI 2, la instrucción '\$a++' no tiene valor (no es una expresión), y no puedes asignarla o usarla de ningún otro modo. PHP mejora las características del incremento/decremento haciéndolos también expresiones, como en C. En PHP, como en C, hay dos tipos de incremento - pre-incremento y post-incremento. Ambos, en esencia, incrementan la variable y el efecto en la variable es idéntico. La diferencia radica en el valor de la propia expresión incremento. El preincremento, escrito '++\$variable', se evalúa al valor incrementado (PHP incrementa la variable antes de leer su valor, de

ahí el nombre 'preincremento'). El postincremento, escrito '\$variable++', se evalúa al valor original de \$variable antes de realizar el incremento (PHP incrementa la variable después de leer su valor, de ahí el nombre 'postincremento').

Un tipo muy corriente de expresiones son las expresiones de comparación. Estas expresiones se evalúan a 0 o 1, significando FALSO (**FALSE**) o CIERTO (**TRUE**), respectivamente. PHP soporta > (mayor que), >= (mayor o igual que), == (igual que), != (distinto), < (menor que) y <= (menor o igual que). Estas expresiones se usan frecuentemente dentro de la ejecución condicional como la instrucción `if`.

El último tipo de expresiones que trataremos, es la combinación operador-asignación. Ya sabes que si quieres incrementar \$a en 1, basta con escribir '\$a++' o '++\$a'. Pero qué pasa si quieres añadir más de 1, por ejemplo 3? Podrías escribir '\$a++' múltiples veces, pero no es una forma de hacerlo ni eficiente ni cómoda. Una práctica mucho más corriente es escribir '\$a = \$a + 3'. '\$a + 3' se evalúa al valor de \$a más 3, y se asigna de nuevo a \$a, lo que resulta en incrementar \$a en 3. En PHP, como en otros lenguajes como C, puedes escribir esto de una forma más concisa, que con el tiempo será más clara y también fácil de entender. Añadir 3 al valor actual de \$a se puede escribir como '\$a += 3'. Esto quiere decir exactamente "toma el valor de \$a, súmale 3, y asígnalo otra vez a \$a". Además de ser más corto y claro, también resulta en una ejecución más rápida. El valor de '\$a += 3', como el valor de una asignación normal y corriente, es el valor asignado. Ten en cuenta que NO es 3, sino el valor combinado de \$a más 3 (ése es el valor asignado a \$a). Cualquier operación binaria puede ser usada en forma de operador-asignación, por ejemplo '\$a -= 5' (restar 5 del valor de \$a), '\$b \*= 7' (multiplicar el valor de \$b por 5), etc.

Hay otra expresión que puede parecer extraña si no la has visto en otros lenguajes, el operador condicional ternario:

```
$first ? $second : $third
```

Si el valor de la primera subexpresión es verdadero (distinto de cero), entonces se evalúa la segunda subexpresión, si no, se evalúa la tercera y ése es el valor.

El siguiente ejemplo te ayudará a comprender un poco mejor el pre y post incremento y las expresiones en general:

```
function double($i) {
    return $i*2;
}
$b = $a = 5;          /* asignar el valor cinco a las variables $a y $b */
$c = $a++;           /* postincremento, asignar el valor original de $a (5) a $c */
$e = $d = ++$b;      /* preincremento, asignar el valor incrementado de $b (6) a
                    $d y a $e */

/* en este punto, tanto $d como $e son iguales a 6 */

$f = double($d++);   /* asignar el doble del valor de $d antes
                    del incremento, 2*6 = 12 a $f */
$g = double(++$e);   /* asignar el doble del valor de $e después
                    del incremento, 2*7 = 14 a $g */
$h = $g += 10;       /* primero, $g es incrementado en 10 y termina valiendo 24.
                    después el valor de la asignación (24) se asigna a $h,
                    y $h también acaba valiendo 24. */
```

Al principio del capítulo hemos dicho que describiríamos los distintos tipos de instrucciones y, como prometimos, las expresiones pueden ser instrucciones. Sin embargo, no todas las expresiones son instrucciones. En este caso, una instrucción tiene la forma 'expr';', es decir, una expresión seguida de un punto y coma. En '\$b=\$a=5;', '\$a=5' es una

expresión válida, pero no es una instrucción en sí misma. Por otro lado '\$b=\$a=5:' sí es una instrucción válida.

Una última cosa que vale la pena mencionar, es el valor booleano de las expresiones. En muchas ocasiones, principalmente en condicionales y bucles, no estás interesado en el valor exacto de la expresión, sino únicamente si es CIERTA (**TRUE**) o FALSA (**FALSE**) (PHP no tiene un tipo booleano específico). El valor de verdad de las expresiones en PHP se calcula de forma similar a perl. Cualquier valor numérico distinto de cero es CIERTO (**TRUE**), cero es FALSO (**FALSE**). Fíjate en que los valores negativos son distinto de cero y considerados CIERTO (**TRUE**)! La cadena vacía y la cadena "0" son FALSO (**FALSE**); todas las demás cadenas son **TRUE**. Con los tipos no escalares (arrays y objetos) - si el valor no contiene elementos se considera FALSO (**FALSE**), en caso contrario se considera CIERTO (**TRUE**).

PHP te brinda una completa y potente implementación de expresiones, y documentarla enteramente está más allá del objetivo de este manual. Los ejemplos anteriores, deberían darte una buena idea de qué son las expresiones y cómo construir expresiones útiles. A lo largo del resto del manual, escribiremos *expr* para indicar una expresión PHP válida.

---

[Anterior](#)

Constantes

[Inicio](#)

[Subir](#)

[Siguiete](#)

Operadores

# Capítulo 10. Operadores

## Tabla de contenidos

[Operadores Aritméticos](#)[Operadores de Asignación](#)[Operadores Bit a bit](#)[Operadores de Comparación](#)[Operador de ejecución](#)[Operadores de Incremento/decremento](#)[Operadores Lógicos](#)[Precedencia de Operadores](#)[Operadores de Cadenas](#)

## Operadores Aritméticos

¿Recuerdas la aritmética básica del colegio? Pues estos operadores funcionan exactamente igual.

**Tabla 10-1. Operadores Aritméticos**

ejemplo	nombre	resultado
$\$a + \$b$	Adición	Suma de $\$a$ y $\$b$ .
$\$a - \$b$	Substracción	Diferencia entre $\$a$ y $\$b$ .
$\$a * \$b$	Multiplicación	Producto de $\$a$ and $\$b$ .
$\$a / \$b$	División	Cociente de $\$a$ entre $\$b$ .
$\$a \% \$b$	Módulo	Resto de $\$a$ dividido entre $\$b$ .

# Capítulo 11. Estructuras de Control

## Tabla de contenidos

[if](#)[else](#)[elseif](#)[Sintaxis Alternativa de Estructuras de Control](#)[while](#)[do..while](#)[for](#)[foreach](#)[break](#)[continue](#)[switch](#)[require\(\)](#)[include\(\)](#)[require\\_once\(\)](#)[include\\_once\(\)](#)

Todo archivo de comandos PHP se compone de una serie de sentencias. Una sentencia puede ser una asignación, una llamada a función, un bucle, una sentencia condicional e incluso una sentencia que no haga nada (una sentencia vacía). Las sentencias normalmente acaban con punto y coma. Además, las sentencias se pueden agrupar en grupos de sentencias encapsulando un grupo de sentencias con llaves. Un grupo de sentencias es también una sentencia. En este capítulo se describen los diferentes tipos de sentencias.

## **if**

La construcción `if` es una de las más importantes características de muchos lenguajes, incluido PHP. Permite la ejecución condicional de fragmentos de código. PHP caracteriza una estructura `if` que es similar a la de C:

```
if (expr)
    sentencia
```

Como se describe en la sección sobre expresiones, *expr* se evalúa a su valor condicional. Si *expr* se evalúa como **TRUE**, PHP ejecutará la sentencia, y si se evalúa como **FALSE** - la ignorará.

El siguiente ejemplo mostraría `a es mayor que b` si `$a` fuera mayor que `$b`:

```
if ($a > $b)
    print "a es mayor que b";
```

A menudo, se desea tener más de una sentencia ejecutada de forma condicional. Por supuesto, no hay necesidad de encerrar cada sentencia con una cláusula `if`. En vez de eso, se pueden agrupar varias sentencias en un grupo de sentencias. Por ejemplo, este código mostraría `a es mayor que b` si `$a` fuera mayor que `$b`, y entonces asignaría el valor de `$a` a `$b`:

```
if ($a > $b) {
    print "a es mayor que b";
    $b = $a;
}
```

Las sentencias `if` se pueden anidar indefinidamente dentro de otras sentencias `if`, lo cual proporciona una flexibilidad completa para ejecuciones condicionales en las diferentes partes de tu programa.

---

[Anterior](#)

Operadores de Cadenas

[Inicio](#)

[Subir](#)

[Siguiente](#)

else

# Capítulo 12. Funciones

## Tabla de contenidos

[Funciones definidas por el usuario](#)

[Parámetros de las funciones](#)

[Devolver valores](#)

[old function](#)

[Funciones variable](#)

## Funciones definidas por el usuario

Una función se define con la siguiente sintaxis:

```
function foo ($arg_1, $arg_2, ..., $arg_n) {  
    echo "Función de ejemplo.\n";  
    return $retval;  
}
```

Cualquier instrucción válida de PHP puede aparecer en el cuerpo de la función, incluso otras funciones y definiciones de [clases](#).

En PHP3, las funciones deben definirse antes de que se referencien. En PHP4 no existe tal requerimiento.

PHP no soporta la sobrecarga de funciones, y tampoco es posible redefinir u ocultar funciones previamente declaradas.

PHP3 no soporta un número variable de parámetros, aunque sí soporta parámetros por defecto (ver [Valores por defecto de de los parámetros](#) para más información). PHP4 soporta ambos: ver [Listas de longitud variable de parámetros](#) y las referencias de las funciones

[func\\_num\\_args\(\)](#), [func\\_get\\_arg\(\)](#), y [func\\_get\\_args\(\)](#) para más información.

---

[Anterior](#)

[include\\_once\(\)](#)

[Inicio](#)

[Subir](#)

[Siguiete](#)

Parámetros de las funciones



# Capítulo 13. Clases y Objetos

## Tabla de contenidos

[class](#)

## class

Una clase es una colección de variables y de funciones que acceden a esas variables. Una clase se define con la siguiente sintaxis:

```
<?php
class Cart {
    var $items; // Items en nuestro carro de la compra

    // Añadir $num artículos de tipo $artnr al carro

    function add_item ($artnr, $num) {
        $this->items[$artnr] += $num;
    }

    // Sacar $num artículos del tipo $artnr del carro

    function remove_item ($artnr, $num) {
        if ($this->items[$artnr] > $num) {
            $this->items[$artnr] -= $num;
            return true;
        } else {
            return false;
        }
    }
}
?>
```

El ejemplo define una clase llamada Cart que consiste en un array asociativo de artículos en el carro y dos funciones para meter y sacar ítems del carro

Las clases son tipos, es decir, son plantillas para variables. Tienes que crear una variable del tipo deseado con el operador new.

```
$cart = new Cart;
$cart->add_item("10", 1);
```

Este ejemplo crea un objeto \$cart de clase Cart. La función add\_item() de ese objeto se llama para añadir un item del artículo número 10 al carro.

Las Clases pueden ser extensiones de otras clases. Las clases extendidas o derivadas tienen todas las variables y funciones de la clase base y lo que les añadas al extender la definición. La herencia múltiple no está soportada.

```
class Named_Cart extends Cart {
    var $owner;

    function set_owner ($name) {
        $this->owner = $name;
    }
}
```

Ese ejemplo define una clase Named\_Cart (carro con nombre o dueño) que tiene todas las variables y funciones de Cart, y además añade la variable \$owner y una función adicional set\_owner(). Un carro con nombre se crea de la forma habitual y, una vez hecho, puedes acceder al propietario del carro. En los carros con nombre también puedes acceder a las funciones normales del carro:

```
$ncart = new Named_Cart; // Creamos un carro con nombre
$ncart->set_owner ("kris"); // Nombramos el carro
print $ncart->owner; // Imprimimos el nombre del propietario
$ncart->add_item ("10", 1); // Funcionalidad heredada de Cart
```

Entre funciones de una clase, la variable \$this hace referencia al propio objeto. Tienes que usar \$this->loquesea para acceder a una variable o función llamada loquesea del objeto actual.

Los constructores son funciones de una clase que se llaman automáticamente al crear una nueva instancia (objeto) de una clase. Una función se convierte en constructor cuando tiene el mismo

nombre que la clase.

```
class Auto_Cart extends Cart {
    function Auto_Cart () {
        $this->add_item ("10", 1);
    }
}
```

Este ejemplo define una clase `Auto_Cart` que es un `Cart` junto con un constructor que inicializa el carro con un item del tipo de artículo "10" cada vez que se crea un nuevo `Auto_Cart` con "new". Los constructores también pueden recibir parámetros y estos parámetros pueden ser opcionales, lo que los hace más útiles.

```
class Constructor_Cart extends Cart {
    function Constructor_Cart ($item = "10", $num = 1) {
        $this->add_item ($item, $num);
    }
}

// Compramos las mismas cosas aburridas de siempre
$default_cart = new Constructor_Cart;

// Compramos las cosas interesantes
$different_cart = new Constructor_Cart ("20", 17);
```

### Atención

Para las clases derivadas, el constructor de la clase padre no es llamado automáticamente cuando se llama al constructor de la clase derivada.

[Anterior](#)

Funciones variable

[Inicio](#)

[Subir](#)

[Siguiete](#)

References Explained

# Capítulo 14. References Explained

## Tabla de contenidos

[What are References](#)

[What do References](#)

[What aren't References](#)

[Returning References](#)

[Unsetting References](#)

[Spotting the Reference](#)

## What are References

References in PHP are means to call same variable content with different names. They are not like C pointers, they are symbol table aliases. Note that in PHP, variable names and variable content are different, so same content can have different names. The most close analogy is Unix filenames and files - variable names are directory entries, while variable contents is the file itself. References can be thought of as hardlinking in Unix filesystem.

## III. Características

### Tabla de contenidos

- 15. [Manejando errores](#)
- 16. [Creando imágenes GIF](#)
- 17. [Autenticación HTTP con PHP](#)
- 18. [Cookies](#)
- 19. [El envío de archivos](#)
- 20. [Usando archivos remotos](#)
- 21. [Manejando conexiones](#)
- 22. [Conexiones persistentes a bases de datos](#)
- 23. [Safe Mode](#)

# Capítulo 15. Manejando errores

Hay 4 tipos de errores y avisos en PHP. Esto son:

- 1 - Errores Normales de Funciones (Normal Function Errors)
- 2 - Avisos Normales (Normal Warnings)
- 4 - Errores del Analizador de código (Parser Errors)
- 8 - Avisos (Notices, advertencia que puedes ignorar, pero que puede implicar un error en tu código).

Los 4 números de arriba son sumados para definir un nivel de aviso de error. El nivel de aviso de error por defecto es el nivel 7, el cual es la suma de 1+2+4, es decir todo excepto los avisos. Este nivel puede ser cambiado en el fichero `php3.ini` con la directiva `error_reporting`. También puede ser configurado en el fichero de configuración del servidor de páginas Apache `httpd.conf`, con la directiva `php3_error_reporting` o también se puede cambiar en tiempo de ejecución usando la función **`error_reporting()`**.

Todas las [expresiones PHP](#) pueden también ser llamadas con el prefijo "@", el cual desactiva el aviso de errores para esa expresión en particular. Si ocurre un error en una expresión en tal situación y la característica [track\\_errors](#) está habilitada, podrás encontrar el mensaje de error en la variable global `$php_errormsg`.

# Capítulo 16. Creando imágenes GIF

PHP no está limitado a crear solo salidas de HTML. Puede ser usado también para crear ficheros de imágenes GIF, o incluso mejor secuencias de imágenes GIF. Necesitará compilar PHP con la librería de funciones de imágenes GD para esta tarea.

## Ejemplo 16-1. Creación de GIFs con PHP

```
<?php
    Header("Content-type: image/gif");
    $string=implode($argv, " ");
    $im = imagecreatefromgif("images/button1.gif");
    $orange = ImageColorAllocate($im, 220, 210, 60);
    $px = (imagesx($im)-7.5*strlen($string))/2;
    ImageString($im, 3, $px, 9, $string, $orange);
    ImageGif($im);
    ImageDestroy($im);
?>
```

Este ejemplo será llamado desde una página con una línea como esta:

`<imgsrc="button.php3?text">` Este script de arriba `button.php3` toma esta cadena "text" la situa sobre la imagen base, en este caso es "images/button1.gif" y muestra la imagen resultante. Esta es una forma muy conveniente para evitar tener que dibujar un nuevo botón cada vez que quiera cambiar el texto del mismo. Con este método los botones son generados dinámicamente.

# Capítulo 17. Autenticación HTTP con PHP

Las características de autenticación HTTP en PHP solo están disponibles cuando se está ejecutando como un módulo en Apache y hasta ahora no lo están en la versión CGI. En un script PHP como módulo de Apache, se puede usar la función [header\(\)](#) para enviar un mensaje de "Autenticación requerida" al navegador cliente haciendo que muestre una ventana de entrada emergente con nombre de usuario y contraseña. Una vez que el usuario ha rellenado el nombre y la contraseña, la URL que contiene el script PHP vuelve a ser llamada con las variables `$PHP_AUTH_USER`, `$PHP_AUTH_PW` y `$PHP_AUTH_TYPE` rellenas con el nombre de usuario, la contraseña y el tipo de autenticación respectivamente. Sólo autenticación "Básica" está soportada en este momento. Consulte la función [header\(\)](#) para más información.

Un fragmento de script de ejemplo que fuerce la autenticación del cliente en una página sería como el siguiente:

## Ejemplo 17-1. Ejemplo de autenticación HTTP

```
<?php
if(!isset($PHP_AUTH_USER)) {
    Header("WWW-Autenticación: Basic realm=\"Mi Reino\");
    Header("HTTP/1.0 401 No autorizado");
    echo "Texto a enviar si pulsa el botón Cancelar\n";
    exit;
} else {
    echo "Hola $PHP_AUTH_USER.<P>";
    echo "Ha introducido $PHP_AUTH_PW como su contraseña.<P>";
}
?>
```

En vez de, sencillamente, mostrar `$PHP_AUTH_USER` y `$PHP_AUTH_PW`, seguramente quiera comprobar la validez del nombre de usuario y la contraseña. Tal vez enviando una consulta a una base de datos o buscando el usuario en un fichero dbm.

Vigile aquí los navegadores Internet Explorer con bugs. Parecen muy quisquillosos con el orden de las cabeceras. Enviar la cabecera *WWW-Autenticación* antes que la cabecera HTTP/1.0 401 parece ser el truco por ahora.

Para prevenir que alguien escriba un script que revele la contraseña de una página que ha sido autenticada a través de algún mecanismo externo tradicional, las variables `PHP_AUTH` no serán rellenas si algún tipo de autenticación externo ha sido activado para una página en particular. En este caso, la variable `$REMOTE_USER` puede ser usada para identificar al usuario autenticado externamente.

Nota, a pesar de todo, lo ya dicho no protege de que alguien que controle una URL no autenticada robe contraseñas de URLs autenticadas en el mismo servidor.

Tanto Netscape como Internet Explorer borrarán la caché de la ventana de autenticación en el navegador local después de recibir una respuesta 401 del servidor. Esto puede usarse, de forma efectiva, para "desconectar" a un usuario, forzándole a reintroducir su nombre y contraseña. Algunas personas usan esto para "hacer caducar" entradas, o para



proveer un botón de "desconectar".

### Ejemplo 17-2. Ejemplo de autenticación HTTP forzando una reentrada

```
<?php
function authenticate() {
    Header( "WWW-Authenticate: Basic realm=\"Test Autenticación Sistema\"");
    Header( "HTTP/1.0 401 No autorizado");
    echo "Debe introducir un nombre de usuario y contraseña válidos para acceder a
    este recurso\n";
    exit;
}

if(!isset($PHP_AUTH_USER) || ($SeenBefore == 1 && !strcmp($OldAuth,
$PHP_AUTH_USER)) ) {
    authenticate();
}
else {
    echo "Bienvenido: $PHP_AUTH_USER<BR>";
    echo "Old: $OldAuth";
    echo "<FORM ACTION=\".$PHP_SELF\" METHOD=POST>\n";
    echo "<INPUT TYPE=HIDDEN NAME=\"SeenBefore\" VALUE=\"1\">\n";
    echo "<INPUT TYPE=HIDDEN NAME=\"OldAuth\" VALUE=\"$PHP_AUTH_USER\">\n";
    echo "<INPUT TYPE=Submit VALUE=\"Re Authenticate\">\n";
    echo "</FORM>\n";
}
?>
```

Este comportamiento no es requerido por el estándar de autenticación básica de HTTP, por lo que nunca debe depender de esto. Pruebas con Lynx han demostrado que Lynx no borra las credenciales de autenticación con una respuesta 401 del servidor, por lo que pulsando atrás y después adelante abriría el recurso de nuevo (siempre que los requerimientos de contraseña no hayan cambiado).

Además note que esto no funciona usando el servidor IIS de Microsoft y la versión CGI de PHP debido a una limitación del IIS

---

[Anterior](#)

Creando imágenes GIF

[Inicio](#)

[Subir](#)

[Siguiente](#)

Cookies

# Capítulo 18. Cookies

PHP soporta transparentemente cookies HTTP. Las Cookies son un mecanismo que sirve para almacenar datos en el navegador del usuario remoto, para así poder identificar al usuario cuando vuelva. Se pueden poner cookies usando la función **setcookies()**. Las Cookies son parte de la cabecera HTTP, por tanto la función [setcookie\(\)](#) debe ser llamada antes de que se produzca cualquier salida al navegador. Esta limitación es la misma a la de la función [header\(\)](#).

Cualquier cookie enviada a ti desde el cliente, automáticamente se convertirá en una variable PHP igual como ocurre con los métodos de datos GET y POST. Si deseas asignar multiples valores a una cookie simple, añade simplemente `[]` a el nombre de la cookie. Para más detalles ver la función [setcookie\(\)](#).

# Capítulo 19. El envío de archivos

## Tabla de contenidos

[Envío de archivos con el método POST](#)

[Errores comunes](#)

[Envío de más de un archivo](#)

[Soporte del método PUT](#)

## Envío de archivos con el método POST

PHP es capaz de recibir envíos de archivo de cualquier navegador que cumpla la norma RFC-1867 (entre los que se incluyen Netscape Navigator 3 o posterior, Microsoft Internet Explorer 3 con un parche o posterior sin éste). Ésta característica permite que los usuarios envíen archivos de texto y binarios. Mediante la autenticación y funciones de manejo de archivos de PHP, es posible un control total de quién puede enviar archivos y que se hace con éstos una vez recibidos.

Es importante destacar que PHP también soporta el método PUT para envío de archivos tal y como lo utiliza Netscape Composer y el cliente Amaya de W3C. Consulte [Soporte del método PUT](#) para más detalles.

Una página de envío de archivos se puede crear mediante un formulario parecido a éste:

### Ejemplo 19-1. Formulario de envío de archivo

```
<FORM ENCTYPE=&quot;multipart/form-data&quot; ACTION=&quot;_URL_&quot;
METHOD=POST&gt;
<INPUT TYPE=&quot;hidden&quot; name=&quot;MAX_FILE_SIZE&quot;
value=&quot;1000&quot;&gt;
Enviar este archivo: <INPUT NAME=&quot;userfile&quot; TYPE=&quot;file&quot;&gt;
<INPUT TYPE=&quot;submit&quot; VALUE=&quot;Enviar&quot;&gt;
</FORM&gt;
```

La `_URL_` debe tener como destino un script PHP. El campo `MAX_FILE_SIZE` debe encontrarse antes del campo `INPUT` y su valor determina el tamaño máximo de archivo que se puede enviar en bytes. Tras la recepción del archivo se definirán en el script PHP destino las siguientes variables:

- `$userfile` - El archivo temporal que se ha guardado en el servidor.
- `$userfile_name` - El nombre original del archivo enviado.
- `$userfile_size` - El tamaño del archivo recibido.
- `$userfile_type` - El tipo mime del archivo si el navegador envía esta información. Por ejemplo: `&quot;image/gif&quot;`.

Es importante recordar que la primera palabra `&quot;$userfile&quot;`; de éstas variables corresponde al nombre (`"NAME="`) del campo `"INPUT TYPE=file"` del formulario. En el ejemplo anterior usamos `&quot;userfile&quot;`.

Los archivos enviados serán guardados en el directorio temporal por defecto del servidor. Podemos variar este directorio mediante la variable de entorno TMPDIR en el entorno donde corre PHP. No se puede establecer este valor usando [putenv\(\)](#) desde un script PHP.

El script PHP que recibe el archivo enviado debe implementar las acciones que se deben llevar a cabo con el archivo acabado de recibir. Por ejemplo se podría utilizar `$file_size` para decidir descartar los archivos que sean demasiado pequeños o demasiado grandes. Sean cual sean las acciones a tomar se debe borrar el archivo temporal o moverlo a algún otro directorio.

El archivo recibido será eliminado inmediatamente del directorio temporal al finalizar el script PHP que lo recibió si no ha sido movido o renombrado.

---

[Anterior](#)

Cookies

[Inicio](#)

[Subir](#)

[Siguiete](#)

Errores comunes

# Capítulo 20. Usando archivos remotos

Siempre que el soporte para la "envoltura URL fopen" esté habilitado cuando se configura PHP (lo cual ocurre a menos que se pasa explícitamente la opción `--disable-url-fopen-wrapper` a configure), se pueden usar URLs HTTP y FTP con la mayoría de las funciones que toman un archivo como parámetro, incluyendo las sentencias [require\(\)](#) e [include\(\)](#).

**Nota:** No se pueden usar archivos remotos en las sentencias [include\(\)](#) y [require\(\)](#) en Windows.

Por ejemplo, se puede usar este para abrir un archivo en un servidor web remoto, analizar en la salida la información que se quiera, y entonces, usar la información en una consulta a base de datos, o simplemente para sacarls en un estilo que coincida con el resto de su sitio web.

## Ejemplo 20-1. Obtener el título de una página remota

```
<?php
  $archivo = fopen("http://www.php.net/", "r");
  if (!$archivo) {
    echo "<p>No se pudo abrir el archivo remoto.\n";
    exit;
  }
  while (!feof($archivo)) {
    $linea = fgets($archivo, 1024);
    /* Esto sólo funciona si el título y sus etiquetas
       están en una línea. */
    if (eregi("<title>(.*?)</title>", $linea, $salida)) {
      $title = $salida[1];
      break;
    }
  }
}
```

```
fclose($file);  
?>
```

También se puede escribir a archivos en un FTP siempre que se conecte como un usuario con los correctos derechos de acceso, y el archivo no exista ya. Para conectar como un usuario distinto de 'anonymous', se necesita especificar el nombre de usuario (y posiblemente contraseña) dentro de la URL, tales como

'ftp://usuario:clave@ftp.ejemplo.com/camino/a/archivo'. (Se puede usar la misma clase de sintaxis para acceder a archivos via HTTP cuando se requería una autenticación de same sort of syntax to access files via HTTP when they require Basic authentication.)

### Ejemplo 20-2. Storing data on a remote server

```
<?php  
$file = fopen("ftp://ftp.php.net/incoming/outputfile", "w");  
if (!$file) {  
    echo "<p>Unable to open remote file for writing.\n";  
    exit;  
}  
/* Write the data here. */  
fputs($file, "$HTTP_USER_AGENT\n");  
fclose($file);  
?>
```

**Nota:** You might get the idea from the example above to use this technique to write to a remote log, but as mentioned above, you can only write to a new file using the URL fopen() wrappers. To do distributed logging like that, you should take a look at [syslog\(\)](#).

---

[Anterior](#)

Soporte del método PUT

[Inicio](#)

[Subir](#)

[Siguiente](#)

Manejando conexiones

# Capítulo 21. Manejando conexiones

**Nota:** Todo lo siguiente se aplica a partir de la 3.0.7 y posterior.

Internamente en PHP se mantiene el estado de la conexión. Hay 3 posibles estados:

- 0 - NORMAL
- 1 - ABORTED (Abortado)
- 2 - TIMEOUT (Fuera de tiempo)

Cuando un script PHP se está ejecutando se activa el estado NORMAL. Si el cliente remoto se desconecta, se pasa al estado ABORTADO. Esto suele ocurrir cuando el usuario pulsa en el botón STOP del navegador. Si se alcanza el límite de tiempo impuesto por PHP (ver [set\\_time\\_limit\(\)](#)), se pasa al estado TIMEOUT.

Puedes decidir si quieres que la desconexión de un cliente cause que tu script sea abortado. Algunas veces es cómodo que tus scripts se ejecuten por completo, incluso si no existe ya un navegador remoto que reciba la salida. El comportamiento por defecto es sin embargo, que tu script se aborte cuando el cliente remoto se desconecta. Este comportamiento puede ser configurado vía la directiva `ignore_user_abort` en el fichero `php3.ini`, o también con la función [ignore\\_user\\_abort\(\)](#). Si no le especificas al PHP que cuando un usuario aborte lo ignore, tu script terminará su ejecución. La única excepción es si tienes registrada un función de desconexión usando la función [register\\_shutdown\\_function\(\)](#). Con una función de desconexión, cuando un usuario remoto pulsa en el botón STOP, la próxima vez que tu script intenta mostrar algo, PHP detecta que la conexión ha sido abortada y se llama a la función de desconexión. Esta función de desconexión también se llama al final de la ejecución de tu script cuando se ha ejecutado normalmente, de manera que si quieres hacer algo diferente en caso de que un cliente se haya desconectado, puedes usar la función [connection\\_aborted\(\)](#). Esta función devuelve verdadero si la conexión fue abortada.

Tu script también se puede terminar por un temporizador interno. El timeout por defecto es de 30 segundos. Se puede cambiar usando la directiva `max_execution_time` en el fichero `php3.ini` o la correspondiente directiva `php3_max_execution_time` en la configuración del servidor de páginas Apache, como también con la función [set\\_time\\_limit\(\)](#). Cuando el

temporizador expira, el script se aborta como en el caso de la desconexión del cliente, de manera que si se ha definido una función de desconexión, esta se llamará. Dentro de esta función de desconexión, puedes comprobar si fue el timeout el que causó que se llamara a la función de desconexión, llamando a la función [connection\\_timeout\(\)](#). Esta función devolverá verdadero si el timeout causa que se llame a la función de desconexión.

Hay que destacar que ambos, el estado ABORTED y el TIMEOUT, se pueden activar al mismo tiempo. Esto es posible si le dices a PHP que ignore las desconexiones intencionadas de los usuarios. PHP aún notará el hecho de que el usuario puede haberse desconectado, pero el script continuará ejecutándose. Si se alcanza el tiempo límite de ejecución será abortado y, si se ha definido una función de desconexión, esta será llamada. En este punto, encontrarás que las funciones [connection\\_timeout\(\)](#) y [connection\\_aborted\(\)](#) devuelven verdadero. Puedes comprobar ambos estados de una manera simple usando la función [connection\\_status\(\)](#). Esta función devuelve un campo de bit de los estados activos. De este modo, si ambos estados están activos devolvería por ejemplo un valor 3.

---

[Anterior](#)

Usando archivos remotos

[Inicio](#)

[Subir](#)

[Siguiente](#)

Conexiones persistentes a  
bases de datos



## Capítulo 22. Conexiones persistentes a bases de datos

Las conexiones persistentes son enlaces SQL que no se cierran cuando termina la ejecución del archivo de comandos. Cuando se pide una conexión persistente, PHP comprueba si hay ya una conexión persistente idéntica (que permanecía abierta desde antes) - y si existe, la usa. Si no existe, crea un enlace. Una conexión 'idéntica' es una conexión que se abrió hacia el mismo "host", con el mismo nombre de usuario y la misma contraseña (donde sea aplicable).

La gente que no está familiarizada con el modo como trabajan y distribuyen la carga los servidores "web" puede confundir que persistente significa lo que no es. En particular, ellas *no* te dan la habilidad de abrir 'sesiones de usuario' en el mismo enlace SQL, *no* dan la habilidad de construir una transacción de forma eficiente, y no hacen un montón de otras cosas. De hecho, para ser extremadamente claros sobre el tema las conexiones persistentes no te dan *ninguna* funcionalidad que no fuera posible con sus hermanas no-persistentes.

¿Por qué?

Esto tiene que ver con el modo como funcionan los servidores "web". Hay tres modos en que un servidor "web" puede utilizar PHP para generar páginas web.

El primer método es usar PHP como una capa CGI. Cuando corre de este modo, se crea y destruye una instancia del intérprete PHP por cada página solicitada (para una página PHP) a tu servidor. Debido a que se destruye después de cada petición, cualquier recurso que adquiriera (como un enlace a un servidor de base de datos SQL) se cierra cuando es destruido. En este caso, no se gana nada si se intentan usar conexiones persistentes.

El segundo, y más popular, método es correr PHP como un módulo en un servidor web multiproceso, lo cual actualmente sólo incluye Apache. Un servidor multiproceso tiene típicamente un proceso (el padre) que coordina un conjunto de procesos (sus hijos) que realmente hacen el trabajo de servir las páginas web. Cuando entra cada petición de un cliente, es entregada a uno de los hijos que no esté ya sirviendo a otro cliente. Esto significa que cuando el mismo cliente hace una segunda petición al servidor, puede ser atendido por un proceso hijo distinto del de la primera vez. Lo que una conexión persistente hace por ti en

este caso es hacerlo de tal modo que cada proceso hijo sólo necesita conectar a tu SQL server la primera vez que sirve una página que hace uso de una conexión así. Cuando otra página solicita una conexión a SQL server, puede reutilizar la conexión que el hijo estableció previamente.

El último método es usar PHP como un "plug-in" para un servidor web multihilo. En la actualidad es solamente teórico -- PHP no funciona aún como "plug-in" para ningún servidor web multihilo. Hay trabajo en progreso para soportar ISAPI, WSAPI y NSAPI (en Windows), lo cual permitirá a PHP ser utilizado como "plug-in" para servidores web multihilo como Netscape FastTrack, Internet Information Server (IIS) de Microsoft, y O'Reilly's WebSite Pro. Cuando esto ocurra, el comportamiento será exactamente el mismo que para el modelo de multiprocesador descrito anteriormente.

Si las conexiones persistentes no aportan ninguna funcionalidad añadida, ¿para qué son buenas?

La respuesta aquí es extremadamente simple -- eficiencia. Las conexiones persistentes son buenas si las cabeceras de control para crear un enlace a tu servidor SQL es alta. Que estas cabeceras sean o no realmente altas depende de muchos factores. Como, qué clase de base de datos es, si esta o no situada en el mismo ordenador que el servidor web, cómo está de cargada la máquina donde se encuentre el servidor SQL, y otras así. El hecho fundamental es que si la cabecera de conexión es alta, las conexiones persistentes te ayudan considerablemente . Ellas hacen que el proceso hijo simplemente conecte solamente una vez durante todo su intervalo de vida, en vez de cada vez que procesa una pagina que requiere conectar al servidor SQL. Esto significa que por cada hijo que abrió una conexión persistente tendrá su propia conexión persistente al servidor. Por ejemplo, si tienes 20 procesos hijos distintos que corran un archivo de comandos que cree una conexión persistente a tu servidor SQL, tendrías 20 conexiones diferentes a ti servidor SQL, una por cada hijo.

Un resumen importante. Las conexiones persistentes fueron diseñadas para tener una equivalencia uno-a-uno con las conexiones normales. Eso significa que deberís *siempre* ser capaz de reemplazar las conexiones persistentes por conexiones no persistentes y no cambiará, el modo como se comporta el archivo de comandos. *Puede* cambiar la eficiencia del archivo de comandos (y probablemente lo hará), ¡pero no su comportamiento!

---

[Anterior](#)

Manejando conexiones

[Inicio](#)

[Subir](#)

[Siguiente](#)

Safe Mode

# Capítulo 23. Safe Mode

## Tabla de contenidos

### [Functions restricted/disabled by Safe Mode](#)

Safe Mode is an attempt to solve the shared-server security problem. It is architecturally incorrect to try to solve this problem at the PHP level, but since the alternatives at the web server and OS levels aren't very realistic, many people, especially ISP's, use Safe Mode for now.

The configuration directives that control Safe Mode are:

```
safe_mode = Off
open_basedir =
safe_mode_exec_dir =
safe_mode_allowed_env_vars = PHP_
safe_mode_protected_env_vars = LD_LIBRARY_PATH
disable_functions =
```

When [safe\\_mode](#) is on, PHP checks to see if the owner of the current script matches the owner of the file to be operated on by a file function. For example:

```
-rw-rw-r--  1 rasmus  rasmus    33 Jul  1 19:20 script.php
-rw-r--r--  1 root   root      1116 May 26 18:01 /etc/passwd
```

Running this script.php

```
<?php
  readfile('/etc/passwd');
?>
```

results in this error when Safe Mode is enabled:

```
Warning: SAFE MODE Restriction in effect. The script whose uid is 500 is not
allowed to access /etc/passwd owned by uid 0 in /docroot/script.php on line 2
```

If instead of [safe\\_mode](#), you set an `open_basedir` directory then all file operations will be limited to files under the specified directory. For example (Apache `httpd.conf` example):

```
<Directory /docroot>
php_admin_value open_basedir /docroot
</Directory>
```

If you run the same script.php with this open\_basedir setting then this is the result:

```
Warning: open_basedir restriction in effect. File is in wrong directory in
/docroot/script.php on line 2
```

You can also disable individual functions. If we add this to our php.ini file:

```
disable_functions readfile,system
```

Then we get this output:

```
Warning: readfile() has been disabled for security reasons in
/docroot/script.php on line 2
```

## Functions restricted/disabled by Safe Mode

This is a still probably incomplete and possibly incorrect listing of the functions limited by [Safe Mode](#).

**Tabla 23-1. Safe Mode limited functions**

Function	Limitations
<a href="#">dbmopen()</a>	Checks whether the file(s)/directories you are about to operate on, have the same UID as the script that is being executed.
<a href="#">dbase_open()</a>	Checks whether the file(s)/directories you are about to operate on, have the same UID as the script that is being executed.
<a href="#">filepro()</a>	Checks whether the file(s)/directories you are about to operate on, have the same UID as the script that is being executed.
<a href="#">filepro_rowcount()</a>	Checks whether the file(s)/directories you are about to operate on, have the same UID as the script that is being executed.
<a href="#">filepro_retrieve()</a>	Checks whether the file(s)/directories you are about to operate on, have the same UID as the script that is being executed.
<a href="#">ifx_*</a>	sql_safe_mode restrictions, (!= Safe Mode)

<b>ingres_*</b> ()	sql_safe_mode restrictions, (!= Safe Mode)
<b>mysql_*</b> ()	sql_safe_mode restrictions, (!= Safe Mode)
<b>pg_loimport()</b>	Checks whether the file(s)/directories you are about to operate on, have the same UID as the script that is being executed.
<a href="#"><u>posix_mkfifo()</u></a>	Checks whether the directory in which you are about to operate, has the same UID as the script that is being executed.
<a href="#"><u>putenv()</u></a>	Obeys the safe_mode_protected_env_vars and safe_mode_allowed_env_vars ini-directives. See also the documentation on <a href="#"><u>putenv()</u></a>
<b>move_uploaded_file()</b>	Checks whether the file(s)/directories you are about to operate on, have the same UID as the script that is being executed.
<a href="#"><u>chdir()</u></a>	Checks whether the directory in which you are about to operate, has the same UID as the script that is being executed.
<b>dl()</b>	This functions is disabled in <a href="#"><u>safe-mode</u></a>
<a href="#"><u>backtick operator</u></a>	This functions is disabled in <a href="#"><u>safe-mode</u></a>
<b>shell_exec()</b> (functional equivalent of backticks)	This functions is disabled in <a href="#"><u>safe-mode</u></a>
<a href="#"><u>exec()</u></a>	You can only execute executables within the <a href="#"><u>safe_mode_exec_dir</u></a> . For practical reasons it's currently not allowed to have . . components in the path to the executable.
<a href="#"><u>system()</u></a>	You can only execute executables within the <a href="#"><u>safe_mode_exec_dir</u></a> . For practical reasons it's currently not allowed to have . . components in the path to the executable.
<a href="#"><u>passthru()</u></a>	You can only execute executables within the <a href="#"><u>safe_mode_exec_dir</u></a> . For practical reasons it's currently not allowed to have . . components in the path to the executable.
<a href="#"><u>popen()</u></a>	You can only execute executables within the <a href="#"><u>safe_mode_exec_dir</u></a> . For practical reasons it's currently not allowed to have . . components in the path to the executable.
<a href="#"><u>mkdir()</u></a>	Checks whether the directory in which you are about to operate, has the same UID as the script that is being executed.
<a href="#"><u>rmdir()</u></a>	Checks whether the file(s)/directories you are about to operate on, have the same UID as the script that is being executed.

<a href="#"><u>rename()</u></a>	Checks whether the file(s)/directories you are about to operate on, have the same UID as the script that is being executed. Checks whether the directory in which you are about to operate, has the same UID as the script that is being executed.
<a href="#"><u>unlink()</u></a>	Checks whether the file(s)/directories you are about to operate on, have the same UID as the script that is being executed. Checks whether the directory in which you are about to operate, has the same UID as the script that is being executed.
<a href="#"><u>copy()</u></a>	Checks whether the file(s)/directories you are about to operate on, have the same UID as the script that is being executed. Checks whether the directory in which you are about to operate, has the same UID as the script that is being executed. (on <i>source</i> and <i>target</i> )
<a href="#"><u>chgrp()</u></a>	Checks whether the file(s)/directories you are about to operate on, have the same UID as the script that is being executed.
<a href="#"><u>chown()</u></a>	Checks whether the file(s)/directories you are about to operate on, have the same UID as the script that is being executed.
<a href="#"><u>chmod()</u></a>	Checks whether the file(s)/directories you are about to operate on, have the same UID as the script that is being executed. In addition, you cannot set the SUID, SGID and sticky bits
<a href="#"><u>touch()</u></a>	Checks whether the file(s)/directories you are about to operate on, have the same UID as the script that is being executed. Checks whether the directory in which you are about to operate, has the same UID as the script that is being executed.
<a href="#"><u>symlink()</u></a>	Checks whether the file(s)/directories you are about to operate on, have the same UID as the script that is being executed. Checks whether the directory in which you are about to operate, has the same UID as the script that is being executed. (note: only the target is checked)
<a href="#"><u>link()</u></a>	Checks whether the file(s)/directories you are about to operate on, have the same UID as the script that is being executed. Checks whether the directory in which you are about to operate, has the same UID as the script that is being executed. (note: only the target is checked)
<a href="#"><u>getallheaders()</u></a>	In Safe Mode, headers beginning with 'authorization' (case-insensitive) will not be returned. Warning: this is broken with the aol-server implementation of <a href="#"><u>getallheaders()</u></a> !

Any function that uses php4/main/fopen\_wrappers.c ??

---

[Anterior](#)

Conexiones persistentes a bases de  
datos

[Inicio](#)

[Subir](#)

[Siguiete](#)

Referencia de las Funciones

# IV. Referencia de las Funciones

## Tabla de contenidos

- I. [Funciones específicas de Apache](#)
- II. [Funciones de matrices](#)
- III. [Funciones Ortográficas](#)
- IV. [Funciones matemáticas de precisión arbitraria](#)
- V. [Bzip2 Compression Functions](#)
- VI. [Funciones de calendario](#)
- VII. [CCVS API Functions](#)
- VIII. [soporte de las funciones COM para Windows](#)
- IX. [Funciones de Clases/Objetos](#)
- X. [Funciones de ClibPDF](#)
- XI. [CURL, Client URL Library Functions](#)
- XII. [Funciones de pago electrónico](#)
- XIII. [Crédit Mutuel CyberMUT functions](#)
- XIV. [Character type functions](#)
- XV. [Funciones de la capa de abstraccion de bases de datos \(dbm-style\)](#)
- XVI. [Funciones de fecha y hora](#)
- XVII. [Funciones para dBase](#)
- XVIII. [Funciones dbm](#)
- XIX. [dbx functions](#)
- XX. [DB++ Functions](#)
- XXI. [Funciones con directorios](#)
- XXII. [Funciones de DOM XML](#)
- XXIII. [Error Handling and Logging Functions](#)
- XXIV. [FrontBase Functions](#)
- XXV. [Funciones filePro](#)
- XXVI. [Funciones del sistema de ficheros](#)
- XXVII. [Funciones Forms Data Format \(Formato de Datos de Formularios\)](#)
- XXVIII. [Funciones FTP](#)
- XXIX. [Function Handling functions](#)



XXX. [GNU Gettext](#)  
XXXI. [GMP functions](#)  
XXXII. [Funciones HTTP](#)  
XXXIII. [Funciones para Hyperwave](#)  
XXXIV. [Funciones para ICAP - Internet Calendar Application Protocol](#)  
XXXV. [iconv functions](#)  
XXXVI. [Funciones de imágenes](#)  
XXXVII. [Funciones IMAP](#)  
XXXVIII. [Funciones para Informix](#)  
XXXIX. [Funciones InterBase](#)  
XL. [Ingres II functions](#)  
XLI. [IRC Gateway Functions](#)  
XLII. [Java](#)  
XLIII. [Funciones LDAP](#)  
XLIV. [Funciones de Correo](#)  
XLV. [Funciones matemáticas](#)  
XLVI. [Multi-Byte String Functions](#)  
XLVII. [MCAL functions](#)  
XLVIII. [Funciones Criptográficas](#)  
XLIX. [Funciones Hash](#)  
L. [Funciones de Microsoft SQL Server](#)  
LI. [Ming functions for Flash](#)  
LII. [Miscelánea de funciones](#)  
LIII. [mnoGoSearch Functions](#)  
LIV. [funciones mSQL](#)  
LV. [Funciones MySQL](#)  
LVI. [Funciones de Red](#)  
LVII. [ODBC functions](#)  
LVIII. [Funciones de Oracle 8](#)  
LIX. [OpenSSL functions](#)  
LX. [Funciones Oracle](#)  
LXI. [Ovrimos SQL functions](#)  
LXII. [Output Control Functions](#)  
LXIII. [Object property and method call overloading](#)  
LXIV. [PDF functions](#)  
LXV. [Verisign Payflow Pro functions](#)

[LXVI. opciones e información de PHP](#)  
[LXVII. Funciones POSIX](#)  
[LXVIII. Funciones de PostgreSQL](#)  
[LXIX. Funciones de ejecución de programas](#)  
[LXX. Printer functions](#)  
[LXXI. Pspell Functions](#)  
[LXXII. GNU Readline](#)  
[LXXIII. Funciones GNU Recode](#)  
[LXXIV. Funciones de expresiones regulares compatibles con Perl](#)  
[LXXV. Funciones para expresiones regulares](#)  
[LXXVI. Satellite CORBA client extension](#)  
[LXXVII. Funciones Semáforo y de memoria compartida](#)  
[LXXVIII. SESAM database functions](#)  
[LXXIX. Session handling functions](#)  
[LXXX. Shared Memory Functions](#)  
[LXXXI. Shockwave Flash functions](#)  
[LXXXII. Funciones SNMP](#)  
[LXXXIII. Socket functions](#)  
[LXXXIV. Funciones de cadenas](#)  
[LXXXV. Funciones de Sybase](#)  
[LXXXVI. Funciones URL](#)  
[LXXXVII. Funciones sobre variables](#)  
[LXXXVIII. Funciones WDDX](#)  
[LXXXIX. Funciones de intérprete XML](#)  
[XC. XSLT functions](#)  
[XCI. YAZ](#)  
[XCII. NIS funciona](#)  
[XCIII. Zip File Functions \(Read Only Access\)](#)  
[XCIV. Funciones de Compresión](#)

---

[Anterior](#)

[Inicio](#)

[Siguiente](#)

Safe Mode

Funciones específicas de  
Apache

# I. Funciones específicas de Apache

## Tabla de contenidos

[apache\\_lookup\\_uri](#) -- Efectua una petición parcial a la URI especificada y devuelve toda la información sobre ella.

[apache\\_note](#) -- Recibe y establece los valores de una petición en una tabla de notas del Apache

[getallheaders](#) -- Recibe todas las cabeceras de una petición HTTP

[virtual](#) -- Ejecuta una sub-petición al Apache

## II. Funciones de matrices

### Tabla de contenidos

[array](#) -- Crear una matriz

[array\\_count\\_values](#) -- Cuenta todos los valores de una matriz

[array\\_flip](#) -- Intercambia los valores de una matriz

[array\\_keys](#) -- Devuelve todas las claves de una matriz

[array\\_merge](#) -- Combina dos o más matrices

[array\\_pad](#) -- Rellena una matriz con un valor hasta el tamaño especificado

[array\\_pop](#) -- Extrae el último elemento de la matriz

[array\\_push](#) -- Inserta uno o más elementos al final de la matriz

[array\\_reverse](#) -- Devuelve una matriz con los elementos en orden inverso

[array\\_shift](#) -- Extrae un elemento del comienzo de la matriz

[array\\_slice](#) -- Extrae una porción de la matriz

[array\\_splice](#) -- Suprime una porción de la matriz y la sustituye por otra cosa

[array\\_unshift](#) -- Introduce uno o más elementos al principio de la matriz

[array\\_values](#) -- Devuelve todos los valores de una matriz

[array\\_walk](#) -- Aplica una función del usuario a cada elemento de una matriz.

[arsort](#) -- Ordena una matriz en orden inverso y mantiene la asociación de índices

[asort](#) -- Ordena una matriz y mantiene la asociación de índices

[compact](#) -- Crea una matriz que contiene variables y sus valores

[count](#) -- Cuenta los elementos de una variable

[current](#) -- Devuelve el elemento actual de una matriz

[each](#) -- Devuelve el siguiente par clave/valor de una matriz

[end](#) -- Mueve el puntero interno de una tabla al último elemento

[extract](#) -- Importa variables a la tabla de símbolos desde una matriz

[in\\_array](#) -- Devuelve **TRUE** si un valor está en una matriz

[key](#) -- Obtiene una clave de una matriz asociativa

[krsort](#) -- Ordena una matriz por clave en orden inverso

[ksort](#) -- Ordena una matriz por clave

[list](#) -- Asigna variables como si fueran una matriz

[next](#) -- Avanza el puntero interno de una matriz

[pos](#) -- Obtiene el elemento actual de una matriz  
[prev](#) -- Rebobina el puntero interno de una matriz  
[rango](#) -- Crea una matriz que contiene un rango de enteros  
[reset](#) -- Fija el puntero interno de una matriz a su primer elemento  
[rsort](#) -- Ordena una matriz en orden inverso  
[shuffle](#) -- Mezcla una matriz  
[sizeof](#) -- Obtiene el número de elementos de una matriz  
[sort](#) -- Ordena una matriz  
[uasort](#) -- Ordena una matriz mediante una función de comparación definida por el usuario y mantiene la asociación de índices  
[uksort](#) -- Ordena una matriz por claves mediante una función definida por el usuario  
[usort](#) -- Ordena una matriz por valores mediante una función definida por el usuario

---

[Anterior](#)

virtual

[Inicio](#)

[Subir](#)

[Siguiete](#)

array

## III. Funciones Ortográficas

Las funciones **aspell()** permiten comprobar la ortografía de una palabra ofreciendote sugerencias. .

Para estas funciones, son necesarias las librerías aspell (ortográficas) disponibles en <http://metalab.unc.edu/kevina/aspell/>

### Tabla de contenidos

[aspell\\_new](#) -- Lee un nuevo diccionario

[aspell\\_check](#) -- Comprueba una palabra

[aspell\\_check-raw](#) -- Comprueba una palabra sin cambiarla o intentar arreglarla

[aspell\\_suggest](#) -- sugiere la ortografía para una palabra

## IV. Funciones matemáticas de precisión arbitraria

Estas funciones sólo están disponibles si el PHP se configuró con `--enable-bcmath`.

### Tabla de contenidos

[bcadd](#) -- Suma dos números de precisión arbitraria.

[bccomp](#) -- Compara dos números de precisión arbitraria.

[bcdiv](#) -- Divide dos números de precisión arbitraria.

[bcmul](#) -- Obtiene el módulo de un número de precisión arbitraria.

[bcmul](#) -- Multiplica dos números de precisión arbitraria.

[bcpow](#) -- Eleva un número de precisión arbitraria a otro.

[bcscale](#) -- Fija el parámetro de escala por defecto para todas las funciones matemáticas bc.

[bcsqrt](#) -- Obtiene la raíz cuadrada de un número de precisión arbitraria.

[bcsub](#) -- Resta un número de precisión arbitraria de otro.

## V. Bzip2 Compression Functions

This module uses the functions of the [bzip2](#) library by Julian Seward to transparently read and write bzip2 (.bz2) compressed files.

Bzip2 support in PHP is not enabled by default. You will need to use the `--with-bz2` configuration option when compiling PHP to enable bzip2 support. This module requires bzip2/libbzip2 version `>= 1.0.x`.

### Small code example

This example opens a temporary file and writes a test string to it, then prints out the contents of the file.

#### Ejemplo 1. Small bzip2 Example

```
<?php

$filename = "/tmp/testfile.bz2";
$str = "This is a test string.\n";

// open file for writing
$bz = bzopen($filename, "w");

// write string to file
bzwrite($bz, $str);

// close file
bzclos($bz);

// open file for reading
$bz = bzopen($filename, "r");

// read 10 characters
print bzread($bz, 10);

// output until end of the file (or the next 1024 char) and close it.
print bzread($bz);

bzclos($bz);
```



?>

## Tabla de contenidos

[bzclose](#) -- Close a bzip2 file pointer

[bzcompress](#) -- Compress a string into bzip2 encoded data

[bzdecompress](#) -- Decompresses bzip2 encoded data

[bzerrno](#) -- Returns a bzip2 error number

[bzerror](#) -- Returns the bzip2 error number and error string in an array

[bzerrstr](#) -- Returns a bzip2 error string

[bzflush](#) -- Force a write of all buffered data

[bzopen](#) -- Open a bzip2 compressed file

[bzread](#) -- Binary safe bzip2 file read

[bzwrite](#) -- Binary safe bzip2 file write

---

[Anterior](#)

bcsb

[Inicio](#)

[Subir](#)

[Siguiete](#)

bzclose

## VI. Funciones de calendario

Las funciones de calendario sólo están disponibles si ha compilado la extensión de calendario que hay en dl/calendar. Lea el documento dl/README como referencia de uso.

La extensión calendar presenta una serie de funciones para simplificar la conversión entre los distintos formatos de calendario. El intermediario estándar en que se basa es en la Cuenta de Días Juliana. La Cuenta de Días Juliana es una cuenta que comienza mucho antes que lo que mucha gente podría necesitar contar (como alrededor del 4000 AC). Para convertir entre sistemas de calendario, primero deberá convertir a la Cuenta de Días Juliana y luego al sistema de su elección. ¡La Cuenta de Días es muy diferente del Calendario Juliano! Para más información sobre sistemas de calendario, visite <http://genealogy.org/~scottle/cal-overview.html>. En estas instrucciones se han incluido extractos entrecomillados de dicha página.

### Tabla de contenidos

[JDToGregorian](#) -- Convierte de Cuenta de Días a fecha Gregoriana

[GregorianToJD](#) -- Convierte de fecha Gregoriana a Cuenta de Días

[JDToJulian](#) -- Convierte de Cuenta de Días a Calendario Juliano

[JulianToJD](#) -- Convierte de Calendario Juliano a Cuenta de Días

[JDToJewish](#) -- Convierte de Cuenta de Días a Calendario Judío

[JewishToJD](#) -- Convierte del Calendario Judío a la Cuenta de Días

[JDToFrench](#) -- Convierte de Cuenta de Días al Calendario Republicano Francés

[FrenchToJD](#) -- Convierte del Calendario Republicano Francés a la Cuenta de Días

[JDMonthName](#) -- Devuelve el nombre de un mes

[JDDayOfWeek](#) -- Devuelve el día de la semana

[easter\\_date](#) -- devuelve la marca de tiempo UNIX para la medianoche de Pascua de un año dado

[easter\\_days](#) -- obtiene el número de días tras el 21 de marzo en que cae la Pascua en un año dado

bzwrite

[Subir](#)

JDTToGregorian

## VII. CCVS API Functions

These functions interface the CCVS API, allowing you to directly work with CCVS from your PHP scripts. CCVS is [RedHat's](#) solution to the "middle-man" in credit card processing. It lets you directly address the credit card clearing houses via your \*nix box and a modem. Using the CCVS module for PHP, you can process credit cards directly through CCVS via your PHP Scripts. The following references will outline the process.

To enable CCVS Support in PHP, first verify your CCVS installation directory. You will then need to configure PHP with the `--with-ccvs` option. If you use this option without specifying the path to your CCVS installation, PHP Will attempt to look in the default CCVS Install location (`/usr/local/ccvs`). If CCVS is in a non-standard location, run configure with: `--with-ccvs=$ccvs_path`, where `$ccvs_path` is the path to your CCVS installation. Please note that CCVS support requires that `$ccvs_path/lib` and `$ccvs_path/include` exist, and include `cv_api.h` under the include directory and `libccvs.a` under the lib directory.

Additionally, a `ccvsd` process will need to be running for the configurations you intend to use in your PHP scripts. You will also need to make sure the PHP Processes are running under the same user as your CCVS was installed as (e.g. if you installed CCVS as user 'ccvs', your PHP processes must run as 'ccvs' as well.)

Additional information about CCVS can be found at <http://www.redhat.com/products/ccvs>.

This documentation section is being worked on. Until then, RedHat maintains slightly outdated but still useful documentation at <http://www.redhat.com/products/ccvs/support/CCVS3.3docs/ProgPHP.html>.

### Tabla de contenidos

--

## VIII. soporte de las funciones COM para Windows

Estas funciones solo están disponibles en la versión para Windows de PHP. Estas funciones han sido añadidas en PHP4.

### Tabla de contenidos

[com\\_load](#) -- ???

[com\\_invoke](#) -- ???

[com\\_propget](#) -- ???

[com\\_get](#) -- ???

[com\\_propput](#) -- ???

[com\\_propset](#) -- ???

[com\\_set](#) -- ???

# IX. Funciones de Clases/Objetos

## Tabla de contenidos

[get\\_class\\_methods](#) -- Devuelve un vector (matriz unidimensional) con los nombres de los métodos de la clase en question.

[get\\_class\\_vars](#) -- Devuelve un vector con las propiedades (inicializadas por defecto) de la clase

[get\\_object\\_vars](#) -- Devuelve un vector de propiedades del objeto

[method\\_exists](#) -- Comprueba que el método de clase existe

## X. Funciones de ClibPDF

ClibPDF Le permite crear documentos PDF con PHP. Está disponible en [FastIO](#) pero no es software libre. Debería leer la licencia antes de comenzar a utilizar ClibPDF. Si usted no puede cumplir el acuerdo de la licencia considere el utilizar la `pdflib` de Thomas Merz, que tambien es muy potente. La funcionalidad y la API de ClibPDF son similares a la `pdflib` de Thomas Merz pero, de acuerdo con FastIO, ClibPDF es mas rápida y crea documentos mas pequeños. Esto puede haber cambiado con la nueva versión 2.0 de `pdflib`. Un simple banco de pruebas (el ejemplo `pdfclock.c` de `pdflib` 2.0 trasformado en un script php) en realidad no muestra ninguna diferencia en velocidad. Por tanto, pruebe las dos y vea cual hace el mejor trabajo para usted.

Esta documentación debería ser leída junto con el manual de ClibPDF ya que este explica la librería con mucho mas detalle.

Muchas funciones en le ClibPDF nativa y el módulo PHP, así como en `pdflib`, tienen el mismo nombre. Todas las funciones excepto [`cpdf\_open\(\)`](#) toman el manejador del documento con el primer parámetro. Actualmente este manejador no se usa internamente desde que ClibPDF no soporta la creación de varios documentos PDF al mismo tiempo. Realmente, ni debería intentarlo, los resultados son impredecibles. No puedo supervisar cuales son las consecuencias en un sistema multihilo. De acuerdo con el autor de ClibPDF, esto cambiará en alguno de las próximas veriones (la versión actual, cuando eto fue escrito es 1.10). Si usted necesita esta capacidad, use el módulo `pdflib`.

**Nota:** La función [`cpdf\_set\_font\(\)`](#) ha cambiado desde que PHP3 soporta fuentes asiáticas. El parámetro que codifica ya no es un entero sino una cadena.

Una gran ventaja de ClibPDF sobre `pdflib` es la posibilidad de crear el documento PDF completamente en memoria sin usar ficheros temporales. Esto también proporciona la capacidad de pasar coordenadas en una unidad de longitud predefinida. Esta es una cualidad útil pero puede ser simulada con [`pdf\_translate\(\)`](#).

La mayoría de las funciones son fáciles de usar. La parte mas difícil es, probablemente, crear un documento PDF muy simple. El siguiente ejemplo debería ayudarle a comenzar. En él se crea un documento con una página. La página contiene el texto "Times-Roman" con una fuente de 30pt. El texto está subrayado.

### Ejemplo 1. Ejemplo simple de ClibPDF

```
<?php
$cpdf = cpdf_open(0);
cpdf_page_init($cpdf, 1, 0, 595, 842);
cpdf_add_outline($cpdf, 0, 0, 0, 1, "Page 1");
cpdf_set_font($cpdf, "Times-Roman", 30, "WinAnsiEncoding");
cpdf_set_text_rendering($cpdf, 1);
cpdf_text($cpdf, "Times Roman outlined", 50, 750);
cpdf_moveto($cpdf, 50, 740);
cpdf_lineto($cpdf, 330, 740);
cpdf_stroke($cpdf);
cpdf_finalize($cpdf);
Header("Content-type: application/pdf");
cpdf_output_buffer($cpdf);
cpdf_close($cpdf);
?>
```

La distribución de pdflib contiene un ejemplo mas complejo que crea una serie de páginas con un reloj analógico. Aquí está ese ejemplo convertido en PHP usando la extensión ClibPDF:

## Ejemplo 2. Ejemplo con pdfclock de la distribución pdflib 2.0

```
<?php
$radius = 200;
$margin = 20;
$pagecount = 40;

$pdf = cpdf_open(0);
cpdf_set_creator($pdf, "pdf_clock.php3");
cpdf_set_title($pdf, "Reloj Analógico");

while($pagecount-- > 0) {
    cpdf_page_init($pdf, $pagecount+1, 0, 2 * ($radius + $margin), 2 * ($radius +
$margin), 1.0);

    cpdf_set_page_animation($pdf, 4, 0.5, 0, 0, 0); /* limpiar */

    cpdf_translate($pdf, $radius + $margin, $radius + $margin);
    cpdf_save($pdf);
    cpdf_setrgbcolor($pdf, 0.0, 0.0, 1.0);

    /* cambio de minuto */
    cpdf_setlinewidth($pdf, 2.0);
    for ($alpha = 0; $alpha < 360; $alpha += 6)
    {
        cpdf_rotate($pdf, 6.0);
        cpdf_moveto($pdf, $radius, 0.0);
        cpdf_lineto($pdf, $radius-$margin/3, 0.0);
        cpdf_stroke($pdf);
    }

    cpdf_restore($pdf);
    cpdf_save($pdf);

    /* cambios de 5 minutos */
    cpdf_setlinewidth($pdf, 3.0);
    for ($alpha = 0; $alpha < 360; $alpha += 30)
    {
        cpdf_rotate($pdf, 30.0);
        cpdf_moveto($pdf, $radius, 0.0);
        cpdf_lineto($pdf, $radius-$margin, 0.0);
        cpdf_stroke($pdf);
    }

    $ltime = getdate();

    /* dibujar la aguja de las horas */
    cpdf_save($pdf);
    cpdf_rotate($pdf, -(($ltime['minutos']/60.0) + $ltime['horas'] - 3.0) * 30.0);
```



```

cpdf_moveto($pdf, -$radius/10, -$radius/20);
cpdf_lineto($pdf, $radius/2, 0.0);
cpdf_lineto($pdf, -$radius/10, $radius/20);
cpdf_closepath($pdf);
cpdf_fill($pdf);
cpdf_restore($pdf);

/* dibujar el minuterero */
cpdf_save($pdf);
cpdf_rotate($pdf, -(($ltime['segundos']/60.0) + $ltime['minutos'] - 15.0) * 6.0);
cpdf_moveto($pdf, -$radius/10, -$radius/20);
cpdf_lineto($pdf, $radius * 0.8, 0.0);
cpdf_lineto($pdf, -$radius/10, $radius/20);
cpdf_closepath($pdf);
cpdf_fill($pdf);
cpdf_restore($pdf);

/* dibujar la segunda mano */
cpdf_setrgbcolor($pdf, 1.0, 0.0, 0.0);
cpdf_setlinewidth($pdf, 2);
cpdf_save($pdf);
cpdf_rotate($pdf, -(($ltime['segundos'] - 15.0) * 6.0));
cpdf_moveto($pdf, -$radius/5, 0.0);
cpdf_lineto($pdf, $radius, 0.0);
cpdf_stroke($pdf);
cpdf_restore($pdf);

/* dibujar un pequeño círculo en el centro */
cpdf_circle($pdf, 0, 0, $radius/30);
cpdf_fill($pdf);

cpdf_restore($pdf);

cpdf_finalize_page($pdf, $pagecount+1);
}

cpdf_finalize($pdf);
Header("Content-type: application/pdf");
cpdf_output_buffer($pdf);
cpdf_close($pdf);
?>

```

## Tabla de contenidos

- [cpdf\\_global\\_set\\_document\\_limits](#) -- Sets document limits for any pdf document
- [cpdf\\_set\\_creator](#) -- Define el campo creator en el documento PDF
- [cpdf\\_set\\_title](#) -- Define el campo title de un documento PDF
- [cpdf\\_set\\_subject](#) -- Define el valor del campo sujet de un documento PDF
- [cpdf\\_set\\_keywords](#) -- Pone el valor del campo 'keywords'(palabras clave) de un documento PDF
- [cpdf\\_open](#) -- Abre un nuevo documento PDF
- [cpdf\\_close](#) -- Cierra un documento PDF
- [cpdf\\_page\\_init](#) -- Comienza una nueva página
- [cpdf\\_finalize\\_page](#) -- Finaliza una página

[cpdf\\_finalize](#) -- Finaliza un documento

[cpdf\\_output\\_buffer](#) -- Pone el documento PDF en el buffer de memoria

[cpdf\\_save\\_to\\_file](#) -- Escribe el documento PDF en un fichero

[cpdf\\_set\\_current\\_page](#) -- Define la página actual

[cpdf\\_begin\\_text](#) -- Inicializa una sección de texto

[cpdf\\_end\\_text](#) -- Finaliza una sección de texto

[cpdf\\_show](#) -- Muestra el texto en la posición actual

[cpdf\\_show\\_xy](#) -- Muestra texto en la posición

[cpdf\\_text](#) -- Muestra texto con parámetros

[cpdf\\_set\\_font](#) -- Selecciona la fuente y el tamaño actual

[cpdf\\_set\\_leading](#) -- Define la distancias entre las líneas de texto

[cpdf\\_set\\_text\\_rendering](#) -- Determina cómo es presentado el texto

[cpdf\\_set\\_horiz\\_scaling](#) -- Define la escala horizontal del texto

[cpdf\\_set\\_text\\_rise](#) -- Define la elevación del texto

[cpdf\\_set\\_text\\_matrix](#) -- Define la matriz de texto

[cpdf\\_set\\_text\\_pos](#) -- Define la posición del texto

[cpdf\\_set\\_char\\_spacing](#) -- Determina el espacio entre caracteres

[cpdf\\_set\\_word\\_spacing](#) -- Define el espacio entre palabras

[cpdf\\_continue\\_text](#) -- Pone texto en la línea siguiente

[cpdf\\_stringwidth](#) -- Devuelve la anchura del texto en la fuente actual

[cpdf\\_save](#) -- Salva el entorno actual

[cpdf\\_restore](#) -- Restaura un entorno formalmente salvado

[cpdf\\_translate](#) -- Define el sistema de origen de coordenadas

[cpdf\\_scale](#) -- Define la escala

[cpdf\\_rotate](#) -- Define la rotación

[cpdf\\_setflat](#) -- Define la monotonía

[cpdf\\_setlinejoin](#) -- Define el parámetro linejoin

[cpdf\\_setlinecap](#) -- Define el parámetro linecap

[cpdf\\_setmiterlimit](#) -- Define el límite del inglete

[cpdf\\_setlinewidth](#) -- Define la profundidad de la línea

[cpdf\\_setdash](#) -- Defina el patrón de la raya

[cpdf\\_moveto](#) -- Define el punto actual

[cpdf\\_rmoveto](#) -- Define el punto actual

[cpdf\\_curveto](#) -- Dibuja una curva

[cpdf\\_lineto](#) -- Dibuja una línea

[cpdf\\_rlineto](#) -- Dibuja una línea

[cpdf\\_circle](#) -- Dibuja un círculo

[cpdf\\_arc](#) -- Dibuja un arco

[cpdf\\_rect](#) -- Dibuja un rectángulo

[cpdf\\_closepath](#) -- Cierra el camino

[cpdf\\_stroke](#) -- Dibuja una línea a lo largo del camino

[cpdf\\_closepath\\_stroke](#) -- Cierra el camino y dibuja una línea a lo largo del camino

[cpdf\\_fill](#) -- Llena el camino actual

[cpdf\\_fill\\_stroke](#) -- Llena y traza el camino actual

[cpdf\\_closepath\\_fill\\_stroke](#) -- Cierra, llena y traza el camino actual

[cpdf\\_clip](#) -- Ajusta al camino actual

[cpdf\\_setgray\\_fill](#) -- Pone el color de relleno al valor gris

[cpdf\\_setgray\\_stroke](#) -- Define el color para dibujar al valor gris

[cpdf\\_setgray](#) -- Pone el color de relleno y dibujo a gris

[cpdf\\_setrgbcolor\\_fill](#) -- Pone el color de relleno a l valor de clor rgb  
[cpdf\\_setrgbcolor\\_stroke](#) -- Pone el color de dibujo al valor de color rgb  
[cpdf\\_setrgbcolor](#) -- Pone el color de relleno y dibujo al valor de color rgb  
[cpdf\\_add\\_outline](#) -- Añade una marca en la página actual  
[cpdf\\_set\\_page\\_animation](#) -- Define la separación entre páginas  
[cpdf\\_import\\_jpeg](#) -- Abre una imagen JPEG  
[cpdf\\_place\\_inline\\_image](#) -- Situa una imagen en la página  
[cpdf\\_add\\_annotation](#) -- Añade una anotación

---

[Anterior](#)

[Inicio](#)

[Siguiete](#)

method\_exists

[Subir](#)

cpdf\_global\_set\_document\_limits

# XI. CURL, Client URL Library Functions

PHP supports libcurl, a library, created by Daniel Stenberg, that allows you to connect and communicate to many different types of servers with many different types of protocols. libcurl currently supports the http, https, ftp, gopher, telnet, dict, file, and ldap protocols. libcurl also supports HTTPS certificates, HTTP POST, HTTP PUT, FTP uploading (this can also be done with PHP's ftp extension), HTTP form based upload, proxies, cookies and user+password authentication.

In order to use the CURL functions you need to install the [CURL](#) package. PHP requires that you use CURL 7.0.2-beta or higher. PHP will not work with any version of CURL below version 7.0.2-beta.

To use PHP's CURL support you must also compile PHP `--with-curl[=DIR]` where DIR is the location of the directory containing the lib and include directories. In the "include" directory there should be a folder named "curl" which should contain the easy.h and curl.h files. There should be a file named "libcurl.a" located in the "lib" directory.

These functions have been added in PHP 4.0.2.

Once you've compiled PHP with CURL support, you can begin using the curl functions. The basic idea behind the CURL functions is that you initialize a CURL session using the [curl\\_init\(\)](#), then you can set all your options for the transfer via the [curl\\_exec\(\)](#) and then you finish off your session using the [curl\\_close\(\)](#). Here is an example that uses the CURL functions to fetch the PHP homepage into a file:

## Ejemplo 1. Using PHP's CURL module to fetch the PHP homepage

```
<?php
$ch = curl_init ("http://www.php.net/");
$fp = fopen ("php_homepage.txt", "w");

curl_setopt ($ch, CURLOPT_INFILE, $fp);
curl_setopt ($ch, CURLOPT_HEADER, 0);

curl_exec ($ch);
curl_close ($ch);
fclose ($fp);
?>
```

### Tabla de contenidos

[curl\\_init](#) -- Initialize a CURL session

[curl\\_setopt](#) -- Set an option for a CURL transfer

[curl\\_exec](#) -- Perform a CURL session

[curl\\_close](#) -- Close a CURL session

[curl\\_version](#) -- Return the current CURL version

---

[Anterior](#)

cpdf\_add\_annotation

[Inicio](#)

[Subir](#)

[Siguiete](#)

curl\_init

## XII. Funciones de pago electrónico

Estas funciones solo están disponibles si el intérprete ha sido compilado con `--with-cybercash=[DIR]`. Estas funciones han sido añadidas en PHP4.

### Tabla de contenidos

[cybercash\\_encr](#) -- ???

[cybercash\\_decr](#) -- ???

[cybercash\\_base64\\_encode](#) -- ???

[cybercash\\_base64\\_decode](#) --

## XIII. Crédit Mutuel CyberMUT functions

This extension allows you to process credit cards transactions using Crédit Mutuel CyberMUT system ([http://www.creditmutuel.fr/centre\\_commercial/vendez\\_sur\\_internet.html](http://www.creditmutuel.fr/centre_commercial/vendez_sur_internet.html)).

CynerMUT is a popular Web Payment Service in France, provided by the Crédit Mutuel bank. If you are foreign in France, these functions will not be useful for you.

These functions are only available if PHP has been compiled with the `--with-cybermut[=DIR]` option, where DIR is the location of `libcm-mac.a` and `cm-mac.h`. You will require the appropriate SDK for your platform, which may be sent to you after your CyberMUT's subscription (contact them via Web, or go to the nearest Crédit Mutuel).

The use of these functions is almost identical to the original functions, except for the parameters of return for [cybermut\\_creerformulairecm\(\)](#) and [cybermut\\_creerreponsecm\(\)](#), which are returned directly by functions PHP, whereas they had passed in reference in the original functions.

These functions have been added in PHP 4.0.6.

**Nota:** These functions only provide a link to CyberMUT SDK. Be sure to read the CyberMUT Developers Guide for full details of the required parameters.

### Tabla de contenidos

[cybermut\\_creerformulairecm](#) -- Generate HTML form of request for payment

[cybermut\\_testmac](#) -- Make sure that there no was data diddling contained in the received message of confirmation

[cybermut\\_creerreponsecm](#) -- Generate the acknowledgement of delivery of the confirmation of payment

---

[Anterior](#)

cybercash\_base64\_decode

[Inicio](#)

[Subir](#)

[Siguiete](#)

cybermut\_crearformulairecm



## XIV. Character type functions

### Aviso

This module is *EXPERIMENTAL*. That means, that the behaviour of these functions, these function names, in concreto ANYTHING documented here can change in a future release of PHP WITHOUT NOTICE. Be warned, and use this module at your own risk.

These functions check whether a character or string falls into a certain character class according to the current locale.

When called with an integer argument these functions behave exactly like their C counterparts.

When called with a string argument they will check every character in the string and will only return **TRUE** if every character in the string matches the requested criteria.

Passing anything else but a string or integer will return **FALSE** immediately.

### Aviso

These functions are new as of PHP 4.0.4 and might change their name in the near future. Suggestions are to change them to **ctype\_issomething()** instead of **ctype\_something()** or even to make them part of `ext/standard` and use their original C-names, although this would possibly lead to further confusion regarding the [isset\(\)](#) vs. **is\_sometype()** problem.

### Tabla de contenidos

[ctype\\_alnum](#) -- Check for alphanumeric character(s)

[ctype\\_alpha](#) -- Check for alphabetic character(s)

[ctype\\_cntrl](#) -- Check for control character(s)

[ctype\\_digit](#) -- Check for numeric character(s)

[ctype\\_lower](#) -- Check for lowercase character(s)

[ctype\\_graph](#) -- Check for any printable character(s) except space

[ctype\\_print](#) -- Check for printable character(s)

[ctype\\_punct](#) -- Check for any printable character which is not whitespace or an alphanumeric character

[ctype\\_space](#) -- Check for whitespace character(s)

[ctype\\_upper](#) -- Check for uppercase character(s)

[ctype\\_xdigit](#) -- Check for character(s) representing a hexadecimal digit

---

[Anterior](#)

cybermut\_creerreponsecm

[Inicio](#)

[Subir](#)

[Siguiete](#)

ctype\_alnum

## XV. Funciones de la capa de abstraccion de bases de datos (dbm-style)

Estas funciones son la base para el acceso a bases de datos del estilo Berkeley DB.

Este es un nivel de abstraccion general para varias bases de datos. Como tal su funcionalidad esta limitada a un grupo de modernas bases de datos como [Sleepycat Software's DB2](#). (Esta no debe confundirse con IBM DB2 software, la cual es soportada mediante las funciones [ODBC](#).)

El comportamiento de varios aspectos depende de la implementacion de la base de datos. Funciones como [dba\\_optimize\(\)](#) y [dba\\_sync\(\)](#) cumpliran su funcionalidad con unas bases de datos pero no con otras.

Los siguientes manejadores (handlers) estan soportados:

- dbm es el mas antiguo (original) tipo de base de datos de la familia de Berkeley DB. Se debe evitar su uso, si es posible. Nosotros no soportamos las funciones de compatibilidad de DB2 y gdbm, porque ellas solo son compatibles a nivel de codigo fuente, pero no pueden manejar el formato original dbm.
- ndbm es un tipo mas nuevo y mas flexible que dbm. Todavia tiene la mayoria de las limitaciones de dbm (Por lo tanto es descartado).
- gdbm es el gestor de bases de datos de [GNU \(database manager\)](#).
- db2 es [Sleepycat Software's DB2](#). Es descrito como "un conjunto de herramientas de programacion que proveen acceso de alto nivel a bases de datos en aplicaciones standalone o en el modelo cliente/servidor. "
- cdb es "una rapida, de confianza, sencilla herramienta para la creacion y lectura de bases de datos constantes." Fue creada por el autor de qmail y puede encontrarse en [here](#). Como la base es constante solo se soportan las operaciones de lectura.

## Ejemplo 1. Ejemplo de DBA

```
<?php
$id = dba_open("/tmp/test.db", "n", "db2");

if(!$id) {
    echo "dba_open failed\n";
    exit;
}

dba_replace("key", "This is an example!", $id);

if(dba_exists("key", $id)) {
    echo dba_fetch("key", $id);
    dba_delete("key", $id);
}

dba_close($id);
?>
```

DBA es "binary safe" y no tiene ningun limite arbitrario. Hereda todas sus limitaciones de la implementacion de base de datos que tenga.

Todos las bases de datos basadas en ficheros deben proveer un mecanismo para establecer el modo a la hora de crear nuevas bases de datos, si ello es posible. Habitualmente este modo es pasado como el cuarto argumento en [dba\\_open\(\)](#) o en [dba\\_popen\(\)](#).

Se puede acceder a todas las entradas de una base de datos de modo secuencial (lineal) usando las funciones [dba\\_firstkey\(\)](#) y [dba\\_nextkey\(\)](#). No se puede cambiar la base de datos mientras se recorre (traversing) por ella.

## Ejemplo 2. Recorriendo una base de datos

```
<?php
# ...open database...

$key = dba_firstkey($id);

while($key != false) {
    if(...) { # remember the key to perform some action later
        $handle_later[] = $key;
    }
    $key = dba_nextkey($id);
}

for($i = 0; $i < count($handle_later); $i++)
    dba_delete($handle_later[$i], $id);

?>
```

### Tabla de contenidos

[dba\\_close](#) -- Cerrar una base de datos  
[dba\\_delete](#) -- Borra una entrada especificada por la clave key  
[dba\\_exists](#) -- Comprueba si la clave key existe  
[dba\\_fetch](#) -- Extrae los datos especificados por la clave key  
[dba\\_firstkey](#) -- Conseguir la primera clave  
[dba\\_insert](#) -- Insertar una entrada  
[dba\\_nextkey](#) -- Extraer la siguiente clave  
[dba\\_popen](#) -- Apertura persistente de una base de datos  
[dba\\_open](#) -- Abrir una base de datos  
[dba\\_optimize](#) -- Optimiza la base de datos  
[dba\\_replace](#) -- Reemplaza o inserta una entrada  
[dba\\_sync](#) -- Sincroniza la base de datos

# XVI. Funciones de fecha y hora

## Tabla de contenidos

[checkdate](#) -- valida una fecha u hora

[date](#) -- da formato a la fecha/hora local

[getdate](#) -- obtiene información de fecha y hora

[gettimeofday](#) -- obtiene la hora actual

[gmdate](#) -- da formato a una fecha/hora GMT/CUT

[gmmktime](#) -- obtiene el valor timestamp UNIX de una fecha GMT

[gmstrftime](#) -- da formato a una fecha/hora GMT/CUT según las convenciones locales

[microtime](#) -- devuelve el valor timestamp UNIX actual con microsegundos

[mktime](#) -- obtiene el timestamp UNIX de una fecha

[strftime](#) -- da formato a la hora o fecha local de acuerdo con las convenciones locales

[time](#) -- devuelve el timestamp UNIX actual

## XVII. Funciones para dBase

Estas funciones permiten el acceso a datos almacenados en formato dBase (dbf).

No hay soporte para índices o campos Memo. Tampoco hay soporte para bloqueo: si dos procesos concurrentes en el servidor modifican el mismo fichero dBase, probablemente se destruirán los datos.

A diferencia de las bases de datos SQL, las "bases de datos" dBase no pueden cambiar su definición. Una vez creado el fichero, la definición de la base de datos es fija. No hay índices que aceleren la búsqueda u organicen los datos de distinto modo. Los ficheros dBase son simples ficheros secuenciales con registros de longitud fija. Los nuevos registros se añaden al final del fichero y los registros borrados se conservan hasta que se llama a la función **dbase\_pack()**.

Se recomienda no utilizar ficheros dBase como bases de datos, sino elegir cualquier servidor SQL; MySQL o Postgres son opciones habituales con PHP. El soporte para dBase se proporciona para permitir importar y exportar datos a y desde la base de datos web, ya que este formato de ficheros es aceptado habitualmente por las hojas de datos y los organizadores de Windows. La importación y exportación de datos es lo único para lo que sirve el soporte dBase.

### Tabla de contenidos

[dbase\\_create](#) -- crea una base de datos dBase

[dbase\\_open](#) -- abre un fichero dBase

[dbase\\_close](#) -- cierra un fichero dBase

[dbase\\_pack](#) -- "empaqueta" un fichero dBase

[dbase\\_add\\_record](#) -- añade un registro a un fichero dBase

[dbase\\_replace\\_record](#) -- reemplaza un registro en un fichero dBase

[dbase\\_delete\\_record](#) -- borra un registro del fichero dBase

[dbase\\_get\\_record](#) -- lee un registro de un fichero dBase

[dbase\\_get\\_record\\_with\\_names](#) -- lee un registro de un fichero dBase como array asociativo

[dbase\\_numfields](#) -- cuenta el número de campos en un fichero dBase

[dbase\\_numrecords](#) -- cuenta el número de registros en un fichero dBase

---

[Anterior](#)  
time

[Inicio](#)  
[Subir](#)

[Siguiete](#)  
dbase\_create



## XVIII. Funciones dbm

Estas funciones le permiten almacenar registros en una base de datos estilo dbm. Este tipo de base de datos (soportadas por las librerías db y gdbm de Berkeley, así como por algunas librerías del sistema y por una librería incluida para acceso a archivos de texto) guarda pares clave/valor (en oposición a los registros completos soportados por las bases de datos relacionales).

### Ejemplo 1. ejemplo de dbm

```
$dbm = dbmopen("vistoya", "w");
if (dbmexists($dbm, $idusuario)) {
    $visto_ya = dbmfetch($dbm, $idusuario);
} else {
    dbminsert($dbm, $idusuario, time());
}
do_stuff();
dbmreplace($dbm, $idusuario, time());
dbmclose($dbm);
```

### Tabla de contenidos

[dbmopen](#) -- abre una base de datos dbm

[dbmclose](#) -- cierra una base de datos dbm

[dbmexists](#) -- dice si existe un valor para una clave dada en la base de datos dbm

[dbmfetch](#) -- obtiene un valor para una clave desde la base de datos dbm

[dbminsert](#) -- inserta un valor para una clave en la base de datos dbm

[dbmreplace](#) -- sustituye el valor de una clave en la base de datos dbm

[dbmdelete](#) -- borra el valor de una clave de una base de datos dbm

[dbmfirstkey](#) -- obtiene la primera clave de una base de datos dbm

[dbmnextkey](#) -- obtiene la siguiente clave de una base de datos dbm

[dblist](#) -- describe la librería compatible dbm que se está usando

---

---

[Anterior](#)

dbase\_numrecords

[Inicio](#)

[Subir](#)

[Siguiete](#)

dbmopen

# XIX. dbx functions

## Aviso

This module is *EXPERIMENTAL*. That means, that the behaviour of these functions, these function names, in concreto ANYTHING documented here can change in a future release of PHP WITHOUT NOTICE. Be warned, and use this module at your own risk.

The dbx module is a database abstraction layer (db 'X', where 'X' is a supported database). The dbx functions allow you to access all supported databases using a single calling convention. In order to have these functions available, you must compile PHP with dbx support by using the `--enable-dbx` option and all options for the databases that will be used, e.g. for MySQL you must also specify `--with-mysql`. The dbx-functions themselves do not interface directly to the databases, but interface to the modules that are used to support these databases. To be able to use a database with the dbx-module, the module must be either linked or loaded into PHP, and the database module must be supported by the dbx-module. Currently, MySQL, PostgreSQL, Microsoft SQL Server, FrontBase and ODBC are supported, but others will follow (soon, I hope :-).

Documentation for adding additional database support to dbx can be found at <http://www.guidance.nl/php/dbx/doc/>.

## Tabla de contenidos

[dbx\\_close](#) -- Close an open connection/database

[dbx\\_connect](#) -- Open a connection/database

[dbx\\_error](#) -- Report the error message of the latest function call in the module (not just in the connection)

[dbx\\_query](#) -- Send a query and fetch all results (if any)

[dbx\\_sort](#) -- Sort a result from a dbx\_query by a custom sort function

[dbx\\_compare](#) -- Compare two rows for sorting purposes

## XX. DB++ Functions

### Aviso

This module is *EXPERIMENTAL*. That means, that the behaviour of these functions, these function names, in concreto ANYTHING documented here can change in a future release of PHP WITHOUT NOTICE. Be warned, and use this module at your own risk.

## DB++ Database Functions

db++, made by the german company [Concept asa](#), is a relational database system with high performance and low memory and disk usage in mind. While providing SQL as an additional language interface it is not really a SQL database in the first place but provides its own AQL query language which is much more influenced by the relational algebra then SQL is.

Concept asa always had an interest in supporting open source languages, db++ has had Perl and Tcl call interfaces for years now and uses Tcl as its internal stored procedure language.

## Requirements

You need the development libraries and header files included in every db++ installaion archive. [Concept asa](#) provides additional [documentation](#) and [Demo versions](#) of db++ for Linux, some other UNIX versions and Windows95/NT.

## Installation

Creation and installation of this extension requires the db++ client libraries and header files to be installed on your system. You have to run **configure** with option `--with-dbplus` to build this extension.

**configure** looks for the client libraries and header files under the default pathes

`/usr/dbplus`, `/usr/local/dbplus` and `/opt/dbplus`. If you have installed db++ in a different place you have add the installation path to the **configure** option like this: `--with-dbplus=/your/installation/path`.

## db++ error codes

**Tabla 1. DB++ Error Codes**

PHP Constant	db++ constant	meaning
DBPLUS_ERR_NOERR	ERR_NOERR	Null error condition
DBPLUS_ERR_DUPLICATE	ERR_DUPLICATE	Tried to insert a duplicate tuple
DBPLUS_ERR_EOSCAN	ERR_EOSCAN	End of scan from rget()
DBPLUS_ERR_EMPTY	ERR_EMPTY	Relation is empty (server)
DBPLUS_ERR_CLOSE	ERR_CLOSE	The server can't close
DBPLUS_ERR_WLOCKED	ERR_WLOCKED	The record is write locked
DBPLUS_ERR_LOCKED	ERR_LOCKED	Relation was already locked
DBPLUS_ERR_NOLOCK	ERR_NOLOCK	Relation cannot be locked
DBPLUS_ERR_READ	ERR_READ	Read error on relation
DBPLUS_ERR_WRITE	ERR_WRITE	Write error on relation
DBPLUS_ERR_CREATE	ERR_CREATE	Create() system call failed
DBPLUS_ERR_LSEEK	ERR_LSEEK	Lseek() system call failed
DBPLUS_ERR_LENGTH	ERR_LENGTH	Tuple exceeds maximum length
DBPLUS_ERR_OPEN	ERR_OPEN	Open() system call failed
DBPLUS_ERR_WOPEN	ERR_WOPEN	Relation already opened for writing
DBPLUS_ERR_MAGIC	ERR_MAGIC	File is not a relation
DBPLUS_ERR_VERSION	ERR_VERSION	File is a very old relation
DBPLUS_ERR_PGFSIZE	ERR_PGFSIZE	Relation uses a different page size

DBPLUS_ERR_CRC	ERR_CRC	Invalid crc in the superpage
DBPLUS_ERR_PIPE	ERR_PIPE	Piped relation requires lseek()
DBPLUS_ERR_NIDX	ERR_NIDX	Too many secondary indices
DBPLUS_ERR_MALLOC	ERR_MALLOC	Malloc() call failed
DBPLUS_ERR_NUSERS	ERR_NUSERS	Error use of max users
DBPLUS_ERR_PREEEXIT	ERR_PREEEXIT	Caused by invalid usage
DBPLUS_ERR_ONTRAP	ERR_ONTRAP	Caused by a signal
DBPLUS_ERR_PREPROC	ERR_PREPROC	Error in the preprocessor
DBPLUS_ERR_DBPARSE	ERR_DBPARSE	Error in the parser
DBPLUS_ERR_DBRUNERR	ERR_DBRUNERR	Run error in db
DBPLUS_ERR_DBPREEXIT	ERR_DBPREEXIT	Exit condition caused by prexit() * procedure
DBPLUS_ERR_WAIT	ERR_WAIT	Wait a little (Simple only)
DBPLUS_ERR_CORRUPT_TUPLE	ERR_CORRUPT_TUPLE	A client sent a corrupt tuple
DBPLUS_ERR_WARNING0	ERR_WARNING0	The Simple routines encountered a non fatal error which was corrected
DBPLUS_ERR_PANIC	ERR_PANIC	The server should not really die but after a disaster send ERR_PANIC to all its clients
DBPLUS_ERR_FIFO	ERR_FIFO	Can't create a fifo
DBPLUS_ERR_PERM	ERR_PERM	Permission denied
DBPLUS_ERR_TCL	ERR_TCL	TCL_error
DBPLUS_ERR_RESTRICTED	ERR_RESTRICTED	Only two users
DBPLUS_ERR_USER	ERR_USER	An error in the use of the library by an application programmer
DBPLUS_ERR_UNKNOWN	ERR_UNKNOWN	

---

## Tabla de contenidos

[dbplus\\_add](#) -- Add a tuple to a relation  
[dbplus\\_aql](#) -- Perform AQL query  
[dbplus\\_chdir](#) -- Get/Set database virtual current directory  
[dbplus\\_close](#) -- Close a relation  
[dbplus\\_curr](#) -- Get current tuple from relation  
[dbplus\\_errcode](#) -- Get error string for given errorcode or last error  
[dbplus\\_erno](#) -- Get error code for last operation  
[dbplus\\_find](#) -- Set a constraint on a relation  
[dbplus\\_first](#) -- Get first tuple from relation  
[dbplus\\_flush](#) -- Flush all changes made on a relation  
[dbplus\\_freealllocks](#) -- Free all locks held by this client  
[dbplus\\_freelock](#) -- Release write lock on tuple  
[dbplus\\_freerlocks](#) -- Free all tuple locks on given relation  
[dbplus\\_getlock](#) -- Get a write lock on a tuple  
[dbplus\\_getunique](#) -- Get a id number unique to a relation  
[dbplus\\_info](#) -- ???  
[dbplus\\_last](#) -- Get last tuple from relation  
[dbplus\\_lockrel](#) -- Request write lock on relation  
[dbplus\\_next](#) -- Get next tuple from relation  
[dbplus\\_open](#) -- Open relation file  
[dbplus\\_prev](#) -- Get previous tuple from relation  
[dbplus\\_rchperm](#) -- Change relation permissions  
[dbplus\\_rcreate](#) --  
[dbplus\\_rctxact](#) --  
[dbplus\\_rctlke](#) --  
[dbplus\\_resolve](#) -- Resolve host information for relation  
[dbplus\\_rkeys](#) -- Specify new primary key for a relation  
[dbplus\\_restorepos](#) -- ???  
[dbplus\\_ropen](#) -- Open relation file local  
[dbplus\\_rquery](#) -- Perform local (raw) AQL query  
[dbplus\\_rename](#) -- Rename a relation  
[dbplus\\_rsecindex](#) -- Create a new secondary index for a relation  
[dbplus\\_runlink](#) -- Remove relation from filesystem  
[dbplus\\_rzap](#) -- Remove all tuples from relation  
[dbplus\\_savepos](#) -- ???

[dbplus\\_setindex](#) -- ???

[dbplus\\_setindexbynumber](#) -- ???

[dbplus\\_sql](#) -- Perform SQL query

[dbplus\\_tcl](#) -- Execute TCL code on server side

[dbplus\\_tremove](#) -- Remove tuple and return new current tuple

[dbplus\\_undo](#) -- ???

[dbplus\\_undoprepere](#) -- ???

[dbplus\\_unlockrel](#) -- Give up write lock on relation

[dbplus\\_unselect](#) -- Remove a constraint from relation

[dbplus\\_update](#) -- Update specified tuple in relation

[dbplus\\_xlockrel](#) -- Request exclusive lock on relation

[dbplus\\_xunlockrel](#) -- Free exclusive lock on relation

---

[Anterior](#)

dbx\_compare

[Inicio](#)

[Subir](#)

[Siguiete](#)

dbplus\_add



# XXI. Funciones con directorios

## Tabla de contenidos

[chdir](#) -- cambia de directorio

[dir](#) -- clase directorio

[closedir](#) -- cierra el manejador de directorios

[opendir](#) -- abre el manejador de directorios

[readdir](#) -- lee las entradas del manejador de directorios

[rewinddir](#) -- rebobinar el manejador de directorios

## XXII. Funciones de DOM XML

### Aviso

This module is *EXPERIMENTAL*. That means, that the behaviour of these functions, these function names, in concreto ANYTHING documented here can change in a future release of PHP WITHOUT NOTICE. Be warned, and use this module at your own risk.

Estas funciones son disponibles solamente si PHP fué configurado con `--with-dom=[DIR]`, usando al libreria de XML de GNOME. Usted va a necesitar como mínimo libxml-2.0.0 (la versión beta no trabajará). Estas funciones fueron añadidas en PHP4.

Este module define las siguientes constantes:

**Tabla 1. Constantes de XML**

Constante	Valor	Descripción
XML_ELEMENT_NODE	1	
XML_ATTRIBUTE_NODE	2	
XML_TEXT_NODE	3	
XML_CDATA_SECTION_NODE	4	
XML_ENTITY_REF_NODE	5	
XML_ENTITY_NODE	6	
XML_PI_NODE	7	
XML_COMMENT_NODE	8	
XML_DOCUMENT_NODE	9	
XML_DOCUMENT_TYPE_NODE	10	
XML_DOCUMENT_FRAG_NODE	11	
XML_NOTATION_NODE	12	
XML_GLOBAL_NAMESPACE	1	

Este modulo define un número de clases. Las funciones de DOM XML devuelven un árbol conteniendo la estructura del documento XML, en el cual cada nodo es un objeto perteneciente a una de estas clases.

### Tabla de contenidos

[xmldoc](#) -- Crea un objeto DOM representando un documento XML

[xmldocfile](#) -- Crea un objeto DOM a partir de un archivo XML

[xmltree](#) -- Crea un árbol de objetos PHP a partir de un documento XML

---

[Anterior](#)  
rewinddir

[Inicio](#)  
[Subir](#)

[Siguiete](#)  
xmldoc

# XXIII. Error Handling and Logging Functions

These are functions dealing with error handling and logging. They allow you to define your own error handling rules, as well as modify the way the errors can be logged. This allows you to change and enhance error reporting to suit your needs.

With the logging functions, you can send messages directly to other machines, to an email (or email to pager gateway!), to system logs, etc., so you can selectively log and monitor the most important parts of your applications and websites.

The error reporting functions allow you to customize what level and kind of error feedback is given, ranging from simple notices to customized functions returned during errors.

## Tabla de contenidos

[error\\_log](#) -- envía un mensaje de error a algún lugar

[error\\_reporting](#) -- establece que errores PHP son registrados

[restore\\_error\\_handler](#) -- Restores the previous error handler function

[set\\_error\\_handler](#) -- Sets a user-defined error handler function.

[trigger\\_error](#) -- Generates a user-level error/warning/notice message

[user\\_error](#) -- Generates a user-level error/warning/notice message

## XXIV. FrontBase Functions

These functions allow you to access FrontBase database servers. In order to have these functions available, you must compile php with fbsql support by using the `--with-fbsql` option. If you use this option without specifying the path to fbsql, php will search for the fbsql client libraries in the default installation location for the platform. Users who installed FrontBase in a non standard directory should always specify the path to fbsql: `--with-fbsql=/path/to/fbsql`. This will force php to use the client libraries installed by FrontBase, avoiding any conflicts.

More information about FrontBase can be found at <http://www.frontbase.com/>.

Documentation for FrontBase can be found at <http://www.frontbase.com/cgi-bin/WebObjects/FrontBase.woa/wa/productsPage?currentPage=Documentation>.

Frontbase support has been added to PHP 4.0.6.

### Tabla de contenidos

[fbsql\\_affected\\_rows](#) -- Get number of affected rows in previous FrontBase operation

[fbsql\\_autocommit](#) -- Enable or disable autocommit.

[fbsql\\_change\\_user](#) -- Change logged in user of the active connection

[fbsql\\_close](#) -- Close FrontBase connection

[fbsql\\_commit](#) -- Commits a transaction to the database

[fbsql\\_connect](#) -- Open a connection to a FrontBase Server

[fbsql\\_create\\_db](#) -- Create a FrontBase database

[fbsql\\_create\\_blob](#) -- Create a BLOB

[fbsql\\_create\\_clob](#) -- Create a CLOB

[fbsql\\_database\\_password](#) -- Sets or retrieves the password for a FrontBase database

[fbsql\\_data\\_seek](#) -- Move internal result pointer

[fbsql\\_db\\_query](#) -- Send a FrontBase query

[fbsql\\_db\\_status](#) -- Get the status for a given database

[fbsql\\_drop\\_db](#) -- Drop (delete) a FrontBase database

[fbsql\\_erro](#) -- Returns the numerical value of the error message from previous FrontBase

operation

[fbsql\\_error](#) -- Returns the text of the error message from previous FrontBase operation

[fbsql\\_fetch\\_array](#) -- Fetch a result row as an associative array, a numeric array, or both

[fbsql\\_fetch\\_assoc](#) -- Fetch a result row as an associative array

[fbsql\\_fetch\\_field](#) -- Get column information from a result and return as an object

[fbsql\\_fetch\\_lengths](#) -- Get the length of each output in a result

[fbsql\\_fetch\\_object](#) -- Fetch a result row as an object

[fbsql\\_fetch\\_row](#) -- Get a result row as an enumerated array

[fbsql\\_field\\_flags](#) -- Get the flags associated with the specified field in a result

[fbsql\\_field\\_name](#) -- Get the name of the specified field in a result

[fbsql\\_field\\_len](#) -- Returns the length of the specified field

[fbsql\\_field\\_seek](#) -- Set result pointer to a specified field offset

[fbsql\\_field\\_table](#) -- Get name of the table the specified field is in

[fbsql\\_field\\_type](#) -- Get the type of the specified field in a result

[fbsql\\_free\\_result](#) -- Free result memory

[fbsql\\_insert\\_id](#) -- Get the id generated from the previous INSERT operation

[fbsql\\_list\\_dbs](#) -- List databases available on a FrontBase server

[fbsql\\_list\\_fields](#) -- List FrontBase result fields

[fbsql\\_list\\_tables](#) -- List tables in a FrontBase database

[fbsql\\_next\\_result](#) -- Move the internal result pointer to the next result

[fbsql\\_num\\_fields](#) -- Get number of fields in result

[fbsql\\_num\\_rows](#) -- Get number of rows in result

[fbsql\\_pconnect](#) -- Open a persistent connection to a FrontBase Server

[fbsql\\_query](#) -- Send a FrontBase query

[fbsql\\_read\\_blob](#) -- Read a BLOB from the database

[fbsql\\_read\\_clob](#) -- Read a CLOB from the database

[fbsql\\_result](#) -- Get result data

[fbsql\\_rollback](#) -- Rollback a transaction to the database

[fbsql\\_set\\_lob\\_mode](#) -- Set the LOB retrieve mode for a FrontBase result set

[fbsql\\_select\\_db](#) -- Select a FrontBase database

[fbsql\\_start\\_db](#) -- Start a database on local or remote server

[fbsql\\_stop\\_db](#) -- Stop a database on local or remote server

[fbsql\\_tablename](#) -- Get table name of field

[fbsql\\_warnings](#) -- Enable or disable FrontBase warnings

user\_error

[Subir](#)

fbsql\_affected\_rows

## XXV. Funciones filePro

Estas funciones permiten acceso en modo de solo-lectura a datos guardados en bases de datos filePro.

filePro es una marca registrada de fP Technologies, Inc. Mas informacion sobre filePro puede encontrarse en <http://www.fptech.com/>.

### Tabla de contenidos

[filepro](#) -- lee y verifica el fichero de mapeo

[filepro\\_fieldname](#) -- obtiene el nombre de un campo

[filepro\\_fieldtype](#) -- obtiene el tipo de campo

[filepro\\_fieldwidth](#) -- obtiene la anchura de un campo

[filepro\\_retrieve](#) -- extrae informacion de una base de datos filePro

[filepro\\_fieldcount](#) -- encuentra cuantos campos existen en una base de datos filePro

[filepro\\_rowcount](#) -- encuentra cuantas filas existen en una base de datos filePro



# XXVI. Funciones del sistema de ficheros

## Tabla de contenidos

[basename](#) -- Devuelve la parte del path correspondiente al nombre del fichero

[chgrp](#) -- Cambia el grupo de un fichero

[chmod](#) -- Cambia permisos de un fichero

[chown](#) -- Cambia el propietario de un fichero

[clearstatcache](#) -- Limpia la cache de estado de un fichero

[copy](#) -- Copia un fichero

[delete](#) -- Una entrada manual inútil

[dirname](#) -- Devuelve la parte del path correspondiente al directorio

[diskfreespace](#) -- Devuelve el espacio disponible en un directorio

[fclose](#) -- Cierra el apuntador a un fichero abierto

[feof](#) -- Verifica si el apuntador a un fichero está al final del fichero (end-of-file)

[fgetc](#) -- Obtiene un caracter del fichero apuntado

[fgetcsv](#) -- Obtiene una línea del fichero apuntado y extrae los campos CSV

[fgets](#) -- Obtiene una línea del fichero apuntado

[fgetss](#) -- Obtiene una línea del fichero apuntado y quita las etiquetas HTML

[file](#) -- lee un fichero completo hacia un array

[file\\_exists](#) -- Verifica si un fichero existe

[fileatime](#) -- Obtiene la última fecha de acceso a un fichero

[filectime](#) -- Obtiene la fecha de cambio del inode del fichero

[filegroup](#) -- Obtiene el grupo de un fichero

[fileinode](#) -- Obtiene el inode del fichero

[filemtime](#) -- Obtiene la fecha de modificación del fichero

[fileowner](#) -- Obtiene el propietario del fichero

[fileperms](#) -- Obtiene los permisos del fichero

[filesize](#) -- Obtiene el tamaño del fichero

[filetype](#) -- Obtiene el tipo de fichero

[flock](#) -- Bloqueo de ficheros portable y asesorado

[fopen](#) -- Abre un fichero o una URL  
[fpassthru](#) -- Saca todos los datos restantes del fichero apuntado  
[fputs](#) -- Escribe en el fichero apuntado  
[fread](#) -- Lee ficheros en plan binario  
[fseek](#) -- Sitúa el apuntador a un fichero  
[ftell](#) -- Pregunta por la posición del apuntador de lectura/escritura de un fichero  
[fwrite](#) -- Escribe ficheros en plan binario  
[set\\_file\\_buffer](#) -- Fija el buffer de fichero del fichero apuntado  
[is\\_dir](#) -- Dice si el fichero nombrado es un directorio  
[is\\_executable](#) -- Dice si el fichero nombrado es ejecutable  
[is\\_file](#) -- Dice si el fichero nombrado es un fichero regular  
[is\\_link](#) -- Dice si el fichero indicado es un enlace simbólico  
[is\\_readable](#) -- Dice si el fichero indicado se puede leer  
[is\\_writable](#) -- Dice si se puede escribir en el fichero indicado  
[link](#) -- Crea un enlace fuerte  
[linkinfo](#) -- Consigue información sobre un enlace  
[mkdir](#) -- Crea un directorio  
[pclose](#) -- Cierra el fichero de proceso apuntado  
[popen](#) -- Abre el fichero de proceso apuntado  
[readfile](#) -- Muestra el contenido de un fichero  
[readlink](#) -- Devuelve el objetivo de un enlace simbólico  
[rename](#) -- Renombra un fichero  
[rewind](#) -- Rebobina la posición del apuntador al fichero  
[rmdir](#) -- Elimina un directorio  
[stat](#) -- Da información sobre un fichero  
[lstat](#) -- Da información sobre un fichero o enlace simbólico  
[symlink](#) -- Crea un enlace simbólico  
[tempnam](#) -- Crea un fichero de nombre único  
[touch](#) -- Fija la fecha de modificación de un fichero  
[umask](#) -- Cambia la umask actual  
[unlink](#) -- Borra un fichero

## XXVII. Funciones Forms Data Format (Formato de Datos de Formularios)

El Formato de Datos de Formulario (FDF) está diseñado para el manejo de formularios en archivos PDF. Se aconseja leer la información disponible en <http://partners.adobe.com/asn/developer/acrosdk/forms.html> para más información sobre lo que es FDF y cómo se usa en general.

**Nota:** Actualmente Adobe sólo proporciona una versión compatible con libc5 para Linux. Las pruebas con glibc2 provocaron un fallo de segmentado. Si alguien es capaz de hacerla funcionar, por favor coméntelo en esta página.

**Nota:** Si tiene problemas configurando php con soporte de fdf, compruebe si el archivo de cabecera FdfTk.h y la librería libFdfTk.so están en su lugar correcto. Deberían encontrarse respectivamente en fdfdir-dir/include y en fdfdir-dir/lib. Este problema no se dará si se limita a desempaquetar la distribución del FdfTk.

La idea general del FDF es similar a los formularios HTML. La diferencia básicamente está en el formato en que se transmiten los datos al servidor cuando se pulsa el botón de envío (este es realmente el Formato de Datos de Formulario) y el formato del formulario en sí mismo (que es el Formato de Documento Portable, PDF). Procesar los datos del FDF es una de las características que proporcionan las funciones fdf. Pero aún hay más. Uno también puede tomar un formulario PDF y rellenar los campos de entrada con datos sin modificar el formulario en sí mismo. En dicho caso, lo que se hace es crear un documento FDF ([fdf\\_create\(\)](#)), fijar los valores de cada campo de entrada ([fdf\\_set\\_value\(\)](#)) y asociarlo con un formulario PDF ([fdf\\_set\\_file\(\)](#)). Finalmente, debe ser enviado al navegador con el MIMEType application/vnd.fdf. El plug-in de Acrobat reader de su navegador reconoce el MIMEType, lee el formulario PDF asociado y rellena los datos a partir del documento FDF.

Los siguientes ejemplos muestran cómo se evalúan los datos de los formularios.

### Ejemplo 1. Evaluando un documento FDF

```
<?php
// Guarda los datos FDF en un archivo temporal
$fdffp = fopen("test.fdf", "w");
fwrite($fdffp, $HTTP_FDF_DATA, strlen($HTTP_FDF_DATA));
fclose($fdffp);

// Abre archivo temporal y evalúa los datos
// El formulario pdf contenía varios campos de texto con los nombres
// volumen, fecha, comentario, editorial, preparador, y dos casillas de verificación
// muestra_editorial y muestra_preparador.
$fdf = fdf_open("test.fdf");
$volume = fdf_get_value($fdf, "volumen");
echo "El campo volumen tiene el valor '<B>$volume</B>'\<BR>";

$date = fdf_get_value($fdf, "fecha");
echo "El campo fecha tiene el valor '<B>$date</B>'\<BR>";

$comment = fdf_get_value($fdf, "comentario");
echo "El campo comentario tiene el valor '<B>$comment</B>'\<BR>";
```

```
if(fdf_get_value($fdf, "muestra_editorial") == "On") {
    $publisher = fdf_get_value($fdf, "editorial");
    echo "El campo editorial tiene el valor '<B>$publisher</B>'  
<BR>";
} else
    echo "No se debe mostrar la editorial.<BR>";

if(fdf_get_value($fdf, "muestra_preparador") == "On") {
    $preparer = fdf_get_value($fdf, "preparador");
    echo "El campo preparador tiene el valor '<B>$preparer</B>'  
<BR>";
} else
    echo "No se debe mostrar el preparador.<BR>";
fdf_close($fdf);
?>
```

### Tabla de contenidos

[fdf\\_open](#) -- Abrir un documento FDF

[fdf\\_close](#) -- Cerrar un documento FDF

[fdf\\_create](#) -- Crear un documento FDF

[fdf\\_save](#) -- Guardar un documento FDF

[fdf\\_get\\_value](#) -- Obtener el valor de un campo

[fdf\\_set\\_value](#) -- Fijar el valor de un campo

[fdf\\_next\\_field\\_name](#) -- Obtener el nombre del siguiente campo

[fdf\\_set\\_ap](#) -- Ajusta la apariencia de un campo

[fdf\\_set\\_status](#) -- Fija el valor de la clave /STATUS

[fdf\\_get\\_status](#) -- Obtener el valor de la clave /STATUS

[fdf\\_set\\_file](#) -- Fijar el valor de la clave /F

[fdf\\_get\\_file](#) -- Obtener el valor de la clave /F

---

[Anterior](#)

unlink

[Inicio](#)

[Subir](#)

[Siguiete](#)

fdf\_open

## XXVIII. Funciones FTP

FTP es el acrónimo de File Transfer Protocol (Protocolo de Transferencia de Ficheros).

Cuando se utiliza el módulo FTP, se definen las siguientes constantes: FTP\_ASCII, y FTP\_BINARY.

### Tabla de contenidos

[ftp\\_connect](#) -- Establece una conexión FTP

[ftp\\_login](#) -- Comienza la sesion en una conexión FTP

[ftp\\_pwd](#) -- Devuelve el nombre del directorio actual

[ftp\\_cdup](#) -- Cambia al directorio padre

[ftp\\_chdir](#) -- Cambia de directorio en un servidor FTP

[ftp\\_mkdir](#) -- Crea un directorio

[ftp\\_rmdir](#) -- Borra un directorio

[ftp\\_nlist](#) -- Devuelve una lista de ficheros del directorio dado.

[ftp\\_rawlist](#) -- Devuelve una lista detallada de ficheros del directorio dado.

[ftp\\_systype](#) -- Devuelve el identificador de tipo de sistema del servidor FTP remoto.

[ftp\\_pasv](#) -- Activa o desactiva el modo pasivo.

[ftp\\_get](#) -- Descarga un fichero del servidor FTP.

[ftp\\_fget](#) -- Descarga un fichero del servidor FTP y lo guarda en un fichero abierto.

[ftp\\_put](#) -- Sube un fichero al servidor FTP.

[ftp\\_fput](#) -- Sube un fichero abierto al servidor FTP.

[ftp\\_size](#) -- Devuelve el tamaño del fichero especificado.

[ftp\\_mdtm](#) -- Devuelve la fecha de última modificación del fichero especificado.

[ftp\\_rename](#) -- Renombra un fichero del servidor FTP.

[ftp\\_delete](#) -- Borra un fichero del servidor FTP.

[ftp\\_quit](#) -- Cierra una conexión FTP

# XXIX. Function Handling functions

These functions all handle various operations involved in working with functions.

## Tabla de contenidos

[call\\_user\\_func](#) -- Call a user function given by the first parameter

[create\\_function](#) -- Create an anonymous (lambda-style) function

[func\\_get\\_arg](#) -- Devuelve un elemento de la lista de argumentos.

[func\\_get\\_args](#) -- Devuelve un array que contiene la lista de argumentos de una función.

[func\\_num\\_args](#) -- Devuelve el número de argumentos pasados a la función.

[function\\_exists](#) -- Devuelve **TRUE** si la función dada ha sido definida

[register\\_shutdown\\_function](#) -- Registra una función para su ejecución en el cierre.

# XXX. GNU Gettext

## Tabla de contenidos

[bindtextdomain](#) -- Establece la ruta para un dominio

[dcgettext](#) -- Omite el dominio para una única búsqueda

[dgettext](#) -- Omite el dominio actual

[gettext](#) -- Realiza una búsqueda del mensaje en el dominio actual

[textdomain](#) -- Establece el dominio actual

---

## XXXI. GMP functions

These functions allow you to work with arbitrary-length integers using GNU MP library. In order to have these functions available, you must compile PHP with GMP support by using the `--with-gmp` option.

You can download the GMP library from <http://www.swox.com/gmp/>. This site also has the GMP manual available.

You will need GMP version 2 or better to use these functions. Some functions may require more recent version of the GMP library.

These functions have been added in PHP 4.0.4.

**Nota:** Most GMP functions accept GMP number arguments, defined as `resource` below. However, most of these functions will accept also numeric and string arguments, given it's possible to convert the latter to number. Also, if there's faster function that can operate on integer arguments, it would be used instead of slower function when supplied arguments are integers. This is done transparently, so the bootom line is that you can use integers in every function that expects GMP number. See also [gmp\\_init\(\)](#) function.

### Ejemplo 1. Factorial function using GMP



```

<?php
function fact ($x) {
    if ($x <= 1)
        return 1;
    else
        return gmp_mul ($x, fact ($x-1));
}

print gmp_strval (fact (1000)) . "\n";
?>

```

This will calculate factorial of 1000 (pretty big number) very fast.

### **Tabla de contenidos**

[gmp\\_init](#) -- Create GMP number

[gmp\\_intval](#) -- Convert GMP number to integer

[gmp\\_strval](#) -- Convert GMP number to string

[gmp\\_add](#) -- Add numbers

[gmp\\_sub](#) -- Subtract numbers

[gmp\\_mul](#) -- Multiply numbers

[gmp\\_div\\_q](#) -- Divide numbers

[gmp\\_div\\_r](#) -- Remainder of the division of numbers

[gmp\\_div\\_qr](#) -- Divide numbers and get quotient and remainder

[gmp\\_div](#) -- Divide numbers

[gmp\\_mod](#) -- Modulo operation

[gmp\\_divexact](#) -- Exact division of numbers

[gmp\\_cmp](#) -- Compare numbers

[gmp\\_neg](#) -- Negate number

[gmp\\_abs](#) -- Absolute value

[gmp\\_sign](#) -- Sign of number

[gmp\\_fact](#) -- Factorial

[gmp\\_sqrt](#) -- Square root

[gmp\\_sqrtrem](#) -- Square root with remainder

[gmp\\_perfect\\_square](#) -- Perfect square check

[gmp\\_pow](#) -- Raise number into power

[gmp\\_powm](#) -- Raise number into power with modulo

[gmp\\_prob\\_prime](#) -- Check if number is "probably prime"  
[gmp\\_gcd](#) -- Calculate GCD  
[gmp\\_gcdext](#) -- Calculate GCD and multipliers  
[gmp\\_invert](#) -- Inverse by modulo  
[gmp\\_legendre](#) -- Legendre symbol  
[gmp\\_jacobi](#) -- Jacobi symbol  
[gmp\\_random](#) -- Random number  
[gmp\\_and](#) -- Logical AND  
[gmp\\_or](#) -- Logical OR  
[gmp\\_xor](#) -- Logical XOR  
[gmp\\_setbit](#) -- Set bit  
[gmp\\_clrbit](#) -- Clear bit  
[gmp\\_scan0](#) -- Scan for 0  
[gmp\\_scan1](#) -- Scan for 1  
[gmp\\_popcount](#) -- Population count  
[gmp\\_hamdist](#) -- Hamming distance

---

[Anterior](#)  
textdomain

[Inicio](#)  
[Subir](#)

[Siguiete](#)  
gmp\_init

## XXXII. Funciones HTTP

Estas funciones permiten manipular la salida que se envía al navegador remoto a nivel de protocolo HTTP.

### Tabla de contenidos

[header](#) -- Manda una cabecera HTTP

[setcookie](#) -- Envía una cookie

# XXXIII. Funciones para Hyperwave

## Introducción

Hyperwave ha sido desarrollado en el [IICM](#) en Graz. Comenzó con el nombre Hyper-G y cambió a Hyperwave cuando fue comercializado (Si lo recuerdo bien, fue en 1996).

Hyperwave no es software gratuito. La versión actual, 4.1, está disponible en [www.hyperwave.com](http://www.hyperwave.com). Se puede solicitar gratuitamente una versión limitada (30 días).

Hyperwave es un sistema de información similar a una base de datos (HIS, Hyperwave Information Server - Servidor Hyperwave de Información). Su objetivo es el almacenamiento y manipulación de documentos. Un documento puede ser cualquier bloque posible de datos que también puede ser almacenado en un archivo. Cada documento se acompaña por su registro de objeto. El registro de objeto contiene metadatos para el documento. Los metadatos son una lista de atributos que pueden ser extendidos por el usuario. Ciertos atributos siempre son fijados por el servidor Hyperwave, otros pueden ser modificados por el usuario. Un atributo es un par nombre/valor de la forma nombre=valor. El registro completo del objeto tiene tantos de estos pares como guste el usuario. El nombre de un atributo no tiene porqué ser único, p. ej. un título puede aparecer varias veces en el registro de un objeto. Esto tiene sentido si se desea especificar un título en diferentes idiomas. En dicho caso existe la convención de que cada valor de título esté precedido por la abreviatura de dos letras del idioma, seguida por dos puntos, como p. ej. 'en:Title in English' o 'es:Título en Español'. Otros atributos tales como descripciones o palabras clave son candidatos potenciales a esta diferenciación. También se pueden reemplazar las abreviaturas de idioma por cualquier otra cadena siempre y cuando estén separadas por los dos puntos del resto del valor del atributo.

Cada registro de objeto tiene una representación nativa como cadena con cada par nombre/valor separado por una línea nueva. La extensión Hyperwave también conoce una segunda representación que consiste en un array asociativo donde el nombre del atributo es la clave. Los valores de atributo multilingües en sí mismos forman otro array asociativo donde la clave es la abreviatura del idioma. Realmente cualquier atributo múltiple forma una tabla asociativa donde la cadena a la izquierda de los dos puntos en el valor de atributo es la clave. (Esto no se ha implementado por completo. Sólo los atributos Title, Description y Keyword

son tratados adecuadamente.)

Aparte de los documentos, todos los hiper-enlaces contenidos en un documento son almacenados también como registros de objeto. Cuando el documento sea insertado en la base de datos, los hiper-enlaces que haya en un documento serán borrados del mismo y almacenados como objetos individuales. El registro de objeto del enlace contiene información acerca de dónde comienza y dónde termina. Para recuperar el documento original se deberá recuperar el documento sin los enlaces y la lista de los mismos para reinsertarla (Las funciones [hw\\_pipedocument\(\)](#) y [hw\\_gettext\(\)](#) hacen esto para usted). La ventaja de separar los enlaces del documento es obvia. Una vez un documento al que apunta un enlace cambia de nombre, el enlace puede modificarse fácilmente. El documento que contiene el enlace no se ve afectado. Incluso se puede añadir un enlace a un documento sin alterarlo.

Decir que [hw\\_pipedocument\(\)](#) y [hw\\_gettext\(\)](#) hacen automáticamente la inserción de enlaces no es tan simple como suena. Insertar los enlaces implica una cierta jerarquía en los documentos. En un servidor web esto viene dado por el sistema de archivos, pero el Hyperwave tiene su propia jerarquía y los nombres no representan la posición de un objeto en dicha jerarquía. Por tanto, la creación de los enlaces precisa primeramente de realizar un mapeado entre el espacio de nombres y la jerarquía del Hyperwave y el espacio de nombres respectivo de una jerarquía de web. La diferencia fundamental entre Hyperwave y la web es la distinción clara entre nombres y jerarquía que se da en el primero. El nombre no contiene ninguna información sobre la posición del objeto en la jerarquía. En la web, el nombre también contiene la información sobre la posición en la jerarquía del objeto. Esto nos lleva a dos posibles formas de mapeo. O bien se reflejan la jerarquía del Hyperwave y el nombre del objeto Hyperwave en el URL o sólo el nombre. Para facilitar las cosas, se utiliza el segundo método. El objeto Hyperwave de nombre 'mi\_objeto' es mapeado a 'http://host/mi\_objeto' sin importar dónde reside dentro de la jerarquía de Hyperwave. Un objeto con el nombre 'padre/mi\_objeto' podría ser el hijo de 'mi\_objeto' en la jerarquía Hyperwave, aunque en el espacio de nombres web aparezca justamente lo opuesto y el usuario pueda ser llevado a confusión. Esto sólo se puede evitar seleccionando nombres de objetos razonables.

Hecha esta decisión surge un segundo problema. ¿Cómo implicar al PHP? el URL `http://host/mi_objeto` no llamará a ningún script PHP a no ser que se le diga al servidor que lo transforme en p. ej. `'http://host/script_php3/mi_objeto'` y que el `'script_php3'` luego evalúe la variable `$PATH_INFO` y recupere el objeto con nombre 'mi\_objeto' del servidor Hyperwave. Hay sólo un pequeño inconveniente que se puede resolver fácilmente. Cuando se reescribe cualquier URL no se permite el acceso a ningún otro documento en el servidor web. Un script de PHP para buscar en el servidor Hyperwave sería imposible. Por lo tanto se necesitará al menos una segunda regla de reescritura para que excluya ciertos URL, como los que empiecen p. ej. por `http://host/Hyperwave`. Básicamente esto sería compartir un espacio de nombres entre el servidor web y el servidor Hyperwave.

Los enlaces se insertan en los documentos basándose en el mecanismo citado más arriba.

Se vuelve más complicado si el PHP no se está ejecutando como módulo del servidor o como script CGI, sino que se ejecuta como aplicación, p. ej. para volcar el contenido del servidor de Hyperwave a un CD-ROM. En dicho caso tiene sentido mantener la jerarquía Hyperwave y mapearla en el sistema de archivos. Esto entra conflicto con los nombres de los objetos si estos reflejan su propia jerarquía (p. ej. eligiendo nombres que comienzan por '/'). Por tanto, la '/' tendrá que ser reemplazada por otro carácter, p. ej. '\_' para continuar.

El protocolo de red para comunicarse con el servidor Hyperwave se denomina [HG-CSP](#) (Hyper-G Client/Server Protocol, Protocolo Hyper-G Cliente/Servidor). Está basado en mensajes que inician ciertas acciones, p. ej. obtener el registro de un objeto. En versiones anteriores del Servidor Hyperwave se distribuyeron dos clientes nativos (Harmony, Amadeus) para la comunicación con el servidor. Ambos desaparecieron cuando se comercializó el Hyperwave. Para sustituirlo se proporcionó el llamado wavemaster. El wavemaster es como un conversor de protocolo de HTTP a HG-CSP. La idea es realizar toda la administración de la base de datos y la visualización de documentos con una interfaz web. El wavemaster implementa una serie de posicionadores para acciones que permiten personalizar la interfaz. Dicho conjunto de posicionadores se denomina el Lenguaje PLACE. El PLACE no posee muchas de las características de un lenguaje de programación real y las extensiones al mismo únicamente alargan la lista de posicionadores. Esto ha obligado al uso de JavaScript que, en mi opinión, no hace la vida más fácil.

Añadir soporte de Hyperwave al PHP rellenaría el espacio que deja la falta de un lenguaje de programación que permita personalizar la interfaz. El PHP implementa todos los mensajes definidos en el HG-CSP pero además proporciona comandos más poderosos, p. ej. recuperar documentos completos.

El Hyperwave tiene su propia terminología para dar nombre a ciertos tipos de información. Esta ha sido ampliamente extendida y anulada. Casi todas las funciones operan en uno de los siguientes tipos de datos.

- ID de objeto: Un valor entero único para cada objeto del servidor Hyperwave. También es uno de los atributos del registro de objeto (ObjectID). Los ID de objeto se usan generalmente como parámetros de entrada que especifican a un objeto.
- registro de objeto: Una cadena con pares atributo-valor con la forma atributo=valos. Los pares están separados unos de otros por un retorno de carro. Un registro de objeto puede convertirse fácilmente en una tabla (array) de objetos usando `hw_object2array()`. Varias funciones devuelven registros de objeto. Los nombres de

dichas funciones terminan en obj.

- tabla de objetos: Una tabla asociativa con todos los atributos de un objeto. La clave es el nombre del atributo. Si un atributo aparece más de una vez en un registro de objeto será convertido en otra tabla asociativa o indizada. Los atributos que dependen del idioma (como el título, claves o descripción) se convertirán en una tabla asociativa con la abreviatura del idioma como clave. El resto de los atributos múltiples crearán una tabla indizada. Las funciones de PHP nunca devuelven tablas de objetos.
- hw\_document: Este es un nuevo tipo de datos que almacena el documento actual, p. ej. HTML, PDF, etc. Está algo optimizado para documentos HTML pero puede usarse para cualquier formato.

Varias funciones que devuelven una tabla de registros de objeto también devuelven una tabla asociativa con información estadística sobre los mismos. La tabla es el último elemento del registro de objeto. La tabla estadística contiene los siguientes elementos:

Hidden

Número de registros de objeto con el atributo PresentationHints puesto a Hidden.

CollectionHead

Número de registros de objeto con el atributo PresentationHints puesto a CollectionHead.

FullCollectionHead

Número de registros de objeto con el atributo PresentationHints puesto a FullCollectionHead.

CollectionHeadNr

Índice a una tabla de registros de objeto con el atributo PresentationHints puesto a CollectionHead.

FullCollectionHeadNr

Índice a una tabla de registros de objeto con el atributo PresentationHints puesto a FullCollectionHead.

Total

Total: Número de registros de objeto.

# Integración con Apache

La extensión Hyperwave se utiliza mejor cuando el PHP se compila como un módulo de Apache. En tal caso el servidor Hyperwave subyacente puede ser ocultado casi por completo de los usuarios si el Apache utiliza su motor de re-escritura. Las siguientes instrucciones explicarán esto.

Como el PHP con soporte Hyperwave incluido en el Apache se ha diseñado para sustituir la solución nativa de Hyperwave basada en Wavemaster, asumiré que el servidor Apache sólo sirve como interfaz web para el Hyperwave. Esto no es necesario, pero simplifica la configuración. El concepto es bastante sencillo. Primeramente necesita un script PHP que evalúe la variable `PATH_INFO` y que trate su valor como el nombre de un objeto Hyperwave. Llamemos a este script 'Hyperwave'. El URL `http://nombre.servidor/Hyperwave/nombre_de_objeto` devolvería entonces el objeto Hyperwave llamado 'nombre\_de\_objeto'. Dependiendo del tipo del objeto, así reaccionará el script. Si es una colección, probablemente devolverá un lista de hijos. Si es un documento devolverá el tipo MIME y el contenido. Se puede mejorar ligeramente si se usa el motor de re-escritura del Apache. Desde el punto de vista del usuario será más sencillo si el URL `http://nombre.servidor/nombre de objeto` devuelve el objeto. La regla de reescritura es muy sencilla:

```
RewriteRule ^/(.*) /usr/local/apache/htdocs/HyperWave/$1 [L]
```

Ahora todo URL apunta a un objeto en el servidor Hyperwave. Esto provoca un problema sencillo de resolver. No hay forma de ejecutar otro script, p. ej. para buscar, salvo el script 'Hyperwave'. Esto se puede solucionar con otra regla de reescritura como la siguiente:

```
RewriteRule ^/hw/(.*) /usr/local/apache/htdocs/hw/$1 [L]
```

Esta reservará el directorio `/usr/local/apache/htdocs/hw` para script adicionales y otros archivos. Sólo hay que asegurarse que esta regla se evalúa antes de la otra. Sólo hay un pequeño inconveniente: todos los objetos Hyperwave cuyo nombre comienza por 'hw/' serán ocultados. así que asegúrese que no utiliza dichos nombres. Si necesita más directorios, p. ej. para imágenes, simplemente añada más reglas o sitúe los archivos en un solo directorio. Por último, no olvide conectar el motor de re-escritura con

```
RewriteEngine on
```



Mi experiencia me ha demostrado que necesitará los siguientes scripts:

- para devolver el script en sí
- para permitir las búsquedas
- para identificarse
- para ajustar su perfil
- uno para cada función adicional como mostrar los atributos del objeto, mostrar información sobre usuarios, mostrar el estado del servidor, etc.

## Pendientes

Aún hay varias cosas pendientes:

- `hw_InsertDocument` debe dividirse en `hw_InsertObject()` y `hw_PutDocument()`.
- Los nombres de algunas funciones aún no están fijados.
- Muchas funciones precisan la conexión actual como primer parámetro. Esto obliga a escribir mucho, lo cual no es con frecuencia necesario si sólo hay una conexión abierta. Una conexión por defecto mejoraría esto.
- La conversión de registro de objeto a tabla de objeto necesita manipular cualquier atributo múltiple.

### Tabla de contenidos

[hw\\_Array2Objrec](#) -- convierte atributos de tabla de objeto a registro de objeto

[hw\\_Children](#) -- id de objeto de los hijos

[hw\\_ChildrenObj](#) -- registros de objeto de los hijos

[hw\\_Close](#) -- cierra la conexión Hyperwave

[hw\\_Connect](#) -- abre una conexión

[hw\\_Cp](#) -- copia objetos

[hw\\_Deleteobject](#) -- borra objetos

[hw\\_DocByAnchor](#) -- id del objeto al que pertenece un enlace

[hw\\_DocByAnchorObj](#) -- registro de objeto al que pertenece un enlace

[hw\\_DocumentAttributes](#) -- registro de objeto de un documento\_hw

[hw\\_DocumentBodyTag](#) -- marca 'BODY' de un documento\_hw

[hw\\_DocumentContent](#) -- devuelve el contenido de un documento\_hw

[hw\\_DocumentSetContent](#) -- fija/sustituye el contenido de un documento\_hw

[hw\\_DocumentSize](#) -- tamaño de un documento\_hw

[hw\\_ErrorMsg](#) -- devuelve un mensaje de error

[hw\\_EditText](#) -- recupera documento de texto

[hw\\_Error](#) -- número de error

[hw\\_Free\\_Document](#) -- libera un documento\_hw

[hw\\_GetParents](#) -- id de objeto de los padres

[hw\\_GetParentsObj](#) -- registros de objeto de los padres

[hw\\_GetChildColl](#) -- id de objeto de colecciones hijas

[hw\\_GetChildCollObj](#) -- registros de objeto de colecciones hijas

[hw\\_GetRemote](#) -- Obtiene un documento remoto

[hw\\_GetRemoteChildren](#) -- Obtiene el hijo del documento remoto

[hw\\_GetSrcByDestObj](#) -- Devuelve los enlaces que apuntan al objeto

[hw\\_GetObject](#) -- registro de objeto

[hw\\_GetAndLock](#) -- devuelve registro de objeto y lo bloquea

[hw\\_GetText](#) -- obtiene un documento de texto

[hw\\_GetObjectByQuery](#) -- buscar objeto

[hw\\_GetObjectByQueryObj](#) -- buscar objeto

[hw\\_GetObjectByQueryColl](#) -- buscar objeto en colección

[hw\\_GetObjectByQueryCollObj](#) -- buscar objeto en colección

[hw\\_GetChildDocColl](#) -- id de objeto de documentos hijos de una colección

[hw\\_GetChildDocCollObj](#) -- registros de objeto de documentos hijos de una colección

[hw\\_GetAnchors](#) -- id de objeto de los enlaces de un documento

[hw\\_GetAnchorsObj](#) -- registros de objeto de los enlaces de un documento

[hw\\_Mv](#) -- mueve objetos

[hw\\_Identify](#) -- identificarse como usuario

[hw\\_InCollections](#) -- comprueba si los id de objeto están en las colecciones

[hw\\_Info](#) -- información sobre conexión

[hw\\_InsColl](#) -- insertar colección

[hw\\_InsDoc](#) -- insertar documento

[hw\\_InsertDocument](#) -- subir cualquier objeto

[hw\\_InsertObject](#) -- inserta un registro de objeto

[hw\\_mapid](#) -- Mapea in id global a un id virtual local

[hw\\_Modifyobject](#) -- modifica el registro de objeto

[hw\\_New\\_Document](#) -- crear nuevo documento

[hw\\_Objrec2Array](#) -- convierte atributos de registro de objeto a tabla de objetos

[hw\\_OutputDocument](#) -- muestra el documento\_hw

[hw\\_pConnect](#) -- hacer una conexión de base de datos permanente

[hw\\_PipeDocument](#) -- recupera cualquier documento

[hw\\_Root](#) -- ID del objeto raíz

[hw\\_Unlock](#) -- desbloquea objeto

[hw\\_Who](#) -- Lista de los usuarios actualmente conectados

[hw Username](#) -- nombre del usuario actualmente conectado

---

[Anterior](#)

setcookie

[Inicio](#)

[Subir](#)

[Siguiete](#)

hw\_Array2Objrec

# XXXIV. Funciones para ICAP - Internet Calendar Application Protocol

Para hacer funcionar estas funciones, deberá compilar el PHP con `--with-icap`. Eso indicará que se precisa la instalación de la librería ICAP. Obtenga la última versión en <http://icap.chek.com/>, compílela e instálela.

## Tabla de contenidos

[icap\\_open](#) -- Abre una conexión ICAP

[icap\\_close](#) -- Cierra un stream ICAP

[icap\\_fetch\\_event](#) -- Obtiene un evento del stream de calendario/

[icap\\_list\\_events](#) -- Devuelve una lista de eventos entre dos instantes dados

[icap\\_store\\_event](#) -- Almacena un evento en un calendario ICAP

[icap\\_delete\\_event](#) -- Borra un evento de un calendario ICAP

[icap\\_snooze](#) -- Apaga la alarma de un evento

[icap\\_list\\_alarms](#) -- Devuelve una lista de los eventos que una alarma ha disparado en el instante dado

## XXXV. iconv functions

This module contains an interface to the iconv library functions. To be able to use the functions defined in this module you must compile you PHP interpreter using the --with-iconv option. In order to do so, you must have iconv() function in standard C library or libiconv installed on your system. libiconv library is available from <http://clisp.cons.org/~haible/packages-libiconv.html>.

iconv library function converts files between various encoded character sets. The supported character sets depend on iconv() implementation on your system. Note that iconv() function in some system is not work well as you expect. In this case, you should install libiconv library.

### Tabla de contenidos

[iconv](#) -- Convert string to requested character encoding

[iconv\\_get\\_encoding](#) -- Get current setting for character encoding conversion

[iconv\\_set\\_encoding](#) -- Set current setting for character encoding conversion

[ob\\_iconv\\_handler](#) -- Convert character encoding as output buffer handler

# XXXVI. Funciones de imágenes

Puede usar las funciones de imágenes de PHP para obtener el tamaño de imágenes JPEG, GIF y PNG, y si tiene la librería GD (disponible en <http://www.boutell.com/gd/>) además será capaz de crear y manipular imágenes.

## Tabla de contenidos

[getImageSize](#) -- Obtiene el tamaño de una imagen GIF, JPG o PNG

[imageArc](#) -- Dibuja una elipse parcial

[imageChar](#) -- Dibuja un carácter horizontalmente

[imageCharUp](#) -- Dibuja un carácter vertical

[imageColorAllocate](#) -- Reserva un color para una imagen

[imageColorAt](#) -- Obtiene el índice del color de un pixel

[imageColorClosest](#) -- Obtiene el índice del color más cercano al color especificado

[imageColorExact](#) -- Devuelve el índice del color especificado

[imageColorResolve](#) -- Devuelve el índice del color especificado o su posible alternativa más cercana

[imageColorSet](#) -- Establece el color para el índice de la paleta especificado

[imageColorsForIndex](#) -- Obtiene los colores de un índice

[imageColorsTotal](#) -- Encuentra el número de colores de la paleta de una imagen

[imageColorTransparent](#) -- Define un color como transparente

[imageCopyResized](#) -- Copia y redimensiona parte de una imagen

[imageCreate](#) -- Crea una nueva imagen

[imageCreateFromGif](#) -- Crea una nueva imagen desde un fichero o una URL

[imageDashedLine](#) -- Dibuja una línea discontinua

[imageDestroy](#) -- Destruye una imagen

[imageFill](#) -- Relleno

[imageFilledPolygon](#) -- Dibuja un polígono relleno

[imageFilledRectangle](#) -- dibuja un rectángulo relleno

[imageFillToBorder](#) -- Relleno de un color específico

[imageFontHeight](#) -- Devuelve la altura de una fuente

[imageFontWidth](#) -- Devuelve la anchura de una fuente

[ImageGif](#) -- Envía una imagen al navegador o a un fichero  
[ImageInterlace](#) -- Activa o desactiva el entrelazado  
[ImageLine](#) -- Dibuja una línea  
[ImageLoadFont](#) -- Carga una fuente nueva  
[ImagePolygon](#) -- Dibuja un polígono  
[ImagePSBBox](#) -- Devuelve el borde que rodea un rectángulo de texto usando fuentes PostScript Type1  
[ImagePSEncodeFont](#) -- Cambia el vector de codificación de caracteres de una fuente  
[ImagePSFreeFont](#) -- Libera la memoria usada por un fuente PostScript Type 1  
[ImagePSLoadFont](#) -- Carga una fuente PostScript Type 1 desde un fichero  
[ImagePSText](#) -- Para dibujar una cadena de texto sobre una imagen usando fuentes PostScript Type1  
[ImageRectangle](#) -- Dibuja un rectángulo  
[ImageSetPixel](#) -- Dibuja un pixel  
[ImageString](#) -- Dibuja una cadena de texto horizontalmente  
[ImageStringUp](#) -- Dibuja una cadena de texto verticalmente  
[ImageSX](#) -- Obtiene la anchura de la imagen  
[ImageSY](#) -- Obtiene la altura de la imagen  
[ImageTTFBBox](#) -- Devuelve un caja que rodea al texto usando fuentes TypeType  
[ImageTTFText](#) -- Escribe texto en la imagen usando fuentes TrueType

---

[Anterior](#)

ob\_iconv\_handler

[Inicio](#)

[Subir](#)

[Siguiete](#)

GetImageSize

## XXXVII. Funciones IMAP

Para hacer funcionar estas funciones, debe compilar PHP con `--with-imap`. Esto requiere que la librería `c-client` esté instalada. Obtenga la última versión de <ftp://ftp.cac.washington.edu/imap/> y compílela. Luego copie `c-client/c-client.a` al directorio `/usr/local/lib` o a cualquier otro directorio de su `LINK` path y copie `c-client/rfc822.h`, `mail.h` y `linkage.h` al directorio `/usr/local/include` o a cualquier otro de su `INCLUDE` path.

Decir que estas funciones no están limitadas al protocolo IMAP, a pesar de sus nombres. La librería subyacente `c-client` también soporta NNTP, POP3 y métodos de acceso local a buzones de correo. Vea [imap\\_open\(\)](#) para una mayor información.

### Tabla de contenidos

- [imap\\_append](#) -- Agrega una cadena de mensaje al buzón especificado
- [imap\\_base64](#) -- Decodifica texto codificado en BASE64
- [imap\\_body](#) -- Lee el cuerpo del mensaje
- [imap\\_check](#) -- Comprueba el estado del buzón actual
- [imap\\_close](#) -- Cierra una sesión IMAP
- [imap\\_createmailbox](#) -- Crea un buzón nuevo
- [imap\\_delete](#) -- Marca un mensaje para ser borrado en el buzón actual
- [imap\\_deletemailbox](#) -- Elimina un buzón
- [imap\\_expunge](#) -- Elimina todos los mensajes marcados como borrados
- [imap\\_fetchbody](#) -- Localiza una sección particular en el cuerpo del mensaje
- [imap\\_fetchstructure](#) -- Lee la estructura de un mensaje concreto
- [imap\\_header](#) -- Lee la cabecera del mensaje
- [imap\\_headers](#) -- Returns headers for all messages in a mailbox
- [imap\\_listmailbox](#) -- Lee la lista de buzones
- [imap\\_getmailboxes](#) -- Lee la lista de buzones, devolviendo información detallada de cada uno
- [imap\\_listsubscribed](#) -- Lista todos los buzones suscritos
- [imap\\_getsubscribed](#) -- Lista todos los buzones suscritos
- [imap\\_mail\\_copy](#) -- Copia los mensajes especificados a un buzón
- [imap\\_mail\\_move](#) -- Mueve los mensajes especificados a un buzón



[imap\\_num\\_msg](#) -- Informa del número de mensajes en el buzón actual

[imap\\_num\\_recent](#) -- Informa el número de mensajes recientes en el buzón actual

[imap\\_open](#) -- Abre una sesión IMAP

[imap\\_ping](#) -- Comprueba si la sesión IMAP está aún activa

[imap\\_renamemailbox](#) -- Renombra un buzón

[imap\\_reopen](#) -- Reabre una sesión IMAP a un nuevo buzón

[imap\\_subscribe](#) -- Subscribe to a mailbox

[imap\\_undelete](#) -- Desmarca los mensajes que están marcados como borrados

[imap\\_unsubscribe](#) -- Unsubscribe from a mailbox

[imap\\_qprint](#) -- Convierte una cadena quoted-printable a una cadena de 8 bit

[imap\\_8bit](#) -- Convierte una cadena de 8bit a una cadena quoted-printable

[imap\\_binary](#) -- Convierte una cadena de 8bit a una cadena base64

[imap\\_scanmailbox](#) -- Lee la lista de buzones y toma una cadena para buscar en el texto del buzón

[imap\\_mailboxmsginfo](#) -- Obtiene información acerca del buzón actual

[imap\\_rfc822\\_write\\_address](#) -- Devuelve una dirección de correo correctamente formateada dado el buzón, host, e información personal.

[imap\\_rfc822\\_parse\\_adrlist](#) -- Examina la cadena dirección

[imap\\_setflag\\_full](#) -- Activa flags en los mensajes

[imap\\_clearflag\\_full](#) -- Limpia los flags de los mensajes

[imap\\_sort](#) -- Ordena un array de cabeceras de mensajes

[imap\\_fetchheader](#) -- Devuelve la cabecera del mensaje

[imap\\_uid](#) -- Esta función devuelve el UID del número de secuencia del mensaje dado

[imap\\_msgno](#) -- Esta función devuelve el número de secuencia del mensaje para el UID dado.

[imap\\_search](#) -- Esta función devuelve un array de mensajes que coinciden con el criterio de búsqueda dado.

[imap\\_last\\_error](#) -- Esta función devuelve el último error IMAP (si se produjo) que ocurrió durante la petición de esta página.

[imap\\_errors](#) -- Esta función devuelve todos los errores IMAP (si hubo) que han ocurrido durante la petición de la página o desde que la pila de errores se inicializó.

[imap\\_alerts](#) -- Esta función devuelve todos los mensajes de alerta IMAP (si hubo) que han ocurrido durante la petición de la página o desde que la pila de alertas fue inicializada.

[imap\\_status](#) -- Esta función devuelve el información de estado de otro buzón distinto al actual.

ImageTTFText

[Subir](#)

imap\_append

## XXXVIII. Funciones para Informix

El conector para Informix Online (ODS) 7.x, SE 7.x y Universal Server (IUS) 9.x se encuentra implementado en "functions/ifx.ec" y "functions/php3\_ifx.h". Para ODS 7.x está completado, con total soporte para columnas de tipo BYTE y TEXT. Para IUS 9.x está parcialmente finalizado: los tipos de datos nuevos están allí (en el IUS 9.x), pero la funcionalidad para SLOB y CLOB se encuentra bajo construcción todavía.

### Notas de configuración:

Antes de ejecutar el guión (script) "configure", asegúrate que la variable "INFORMIXDIR" ha sido definida.

Si ejecutas "configure --with\_informix=yes" entonces el guión de configuración detectará automáticamente las librerías y los directorios include. Puedes obviar esta detección definiendo las variables de entorno "IFX\_LIBDIR", "IFX\_LIBS" y "IFX\_INCDIR". Definirás la variable de compilación condicional "HAVE\_IFX\_IUS" si la versión de Informix es 9.00 o superior.

### Algunas notas sobre el uso de BLOBs (columnas de tipo TEXT y BYTE):

BLOBs son normalmente manipulados por enteros, los cuales representan identificadores de BLOB. Las consultas de selección devuelven un "blob id" para columnas de tipo BYTE y TEXT. Si eliges trabajar con los BLOBs en memoria (con: "ifx\_blobinfile(0);") entonces puedes obtener el contenido con "string\_var = ifx\_get\_blob(\$blob\_id);". Si prefieres manipularlos en ficheros usa "ifx\_blobinfile(1);" y "ifx\_get\_blob(\$blob\_id);" devolverá el nombre del archivo. En este caso, utiliza las funciones habituales de entrada y salida de ficheros para obtener el contenido de los blob.

Para consultas de inserción y actualización debes crear estos identificadores de blob con "ifx\_create\_blob(..);". Entonces pondrás los identificadores de blob en un array y sustituirás en la cadena de la consulta las columnas de tipo blob por una interrogación (?). Para inserciones y actualizaciones eres responsable

de definir el contenido de los blob con `ifx_update_blob(...)`.

La conducta de columnas BLOB puede ser modificada mediante variables de configuración, las cuales pueden ser definidas en tiempo de ejecución mediante funciones.

variable de configuración: `ifx.textasvarchar`

variable de configuración: `ifx.byteasvarchar`

funciones en tiempo de ejecución:

`ifx_textasvarchar(0)`: usa identificadores de blob para columnas de tipo TEXT en las consultas de selección

`ifx_byteasvarchar(0)`: usa identificadores de blob para columnas de tipo BYTE en las consultas de selección

`ifx_textasvarchar(1)`: devuelve columnas de tipo TEXT como si fueran de tipo VARCHAR, sin tener que usar identificadores de blob en las consultas de selección

`ifx_byteasvarchar(1)`: devuelve columnas de tipo BYTE como si fueran de tipo VARCHAR, sin tener que usar identificadores de blob en las consultas de selección.

variable de configuración: `ifx.blobinfile`

función en tiempo de ejecución:

`ifx_blobinfile_mode(0)`: devuelve columnas de tipo BYTE en memoria, el identificador de blob te permite obtener el contenido.

`ifx_blobinfile_mode(1)`: devuelve columnas de tipo BYTE en un fichero, el identificador te permite saber el nombre de dicho archivo.

Si defines `ifx_text/byteasvarchar` a 1 entonces puedes usar columnas de tipo TEXT y BYTE en las consultas de selección como campos de tipo VARCHAR, pero teniendo en cuenta que tendrán un mayor tamaño que el

habitual. Ya que en PHP todas las cadenas son posibles, esto permite datos binarios. De esta forma, se pueden manejar correctamente. La información devuelta puede contener cualquier cosa, tú eres responsable del contenido.

Si defines `ifx_blobinfile` a 1, utiliza el nombre del archivo devuelto por `ifx_get_blob(..)` para acceder a los contenidos de los blobs. En este caso, ERES REPOSABLE DE ELIMINAR EL ARCHIVO TEMPORAL GENERADO POR INFORMIX cuando accedas a los registros. Cada nueva fila obtenida creará un nuevo archivo temporal para cada columna de tipo BYTE.

El directorio donde se guardan los archivos temporales puede ser definido por la variable de entorno `blobdir`, por defecto es ".", es decir, el directorio actual. Así, `putenv(blobdir=tmpblob)`; definirá un directorio donde se localizarán todos los ficheros temporales y facilitará su borrado. Todos los nombres de los archivos comienzan por "blb".

### **Recortado (trimming) automático de datos de tipo "char" (SQLCHAR y SQLNCHAR):**

Puede ser definido con la variable de configuración

`ifx.charasvchar`: si se define a 1 eliminará automáticamente los espacios en blanco al final de la cadena.

### **Valores NULL:**

La variable de configuración `ifx.nullformat` (y en tiempo de ejecución [`ifx\_nullformat\(\)`](#)) cuando sea definida a **TRUE** devolverá columnas **NULL** como la cadena "**NULL**", si es definida a **FALSE** entonces la cadena vacía. Esto permite distinguir entre columnas **NULL** y vacías.

### **Tabla de contenidos**

[`ifx\_connect`](#) -- Abre una conexión con un servidor Informix

[`ifx\_pconnect`](#) -- Abre una conexión permanente con Informix

[`ifx\_close`](#) -- Cierra una conexión con Informix

[`ifx\_query`](#) -- Envía una consulta a Informix

[`ifx\_prepare`](#) -- Prepara una sentencia SQL para su ejecución

[`ifx\_do`](#) -- Ejecuta una sentencia SQL preparada previamente

[`ifx\_error`](#) -- Devuelve el código de error de la última llamada a Informix

[ifx\\_errormsg](#) -- Devuelve el mensaje de error de la última llamada a Informix  
[ifx\\_affected\\_rows](#) -- Obtiene el número de registros procesados por una consulta  
[ifx\\_getsqlca](#) -- Después de una consulta, obtiene el contenido de sqlca.sqlerrd[0..5]  
[ifx\\_fetch\\_row](#) -- Obtiene registros como un array (vector) enumerado  
[ifx\\_htmltbl\\_result](#) -- Muestra todos los registros de una consulta en una tabla HTML  
[ifx\\_fieldtypes](#) -- Obtiene los campos de una consulta SQL  
[ifx\\_fieldproperties](#) -- Indica las propiedades de los campos de una consulta SQL  
[ifx\\_num\\_fields](#) -- Devuelve el número de columnas en una consulta  
[ifx\\_num\\_rows](#) -- Cuenta los registros ya leídos de una consulta  
[ifx\\_free\\_result](#) -- Libera los recursos de una consulta  
[ifx\\_create\\_char](#) -- Crea un objeto char  
[ifx\\_free\\_char](#) -- Elimina un objeto char  
[ifx\\_update\\_char](#) -- Actualiza el contenido de un objeto char  
[ifx\\_get\\_char](#) -- Obtiene el contenido de un objeto char  
[ifx\\_create\\_blob](#) -- Crea un objeto blob  
[ifx\\_copy\\_blob](#) -- Duplica el objeto blob dado  
[ifx\\_free\\_blob](#) -- Borra el objeto blob  
[ifx\\_get\\_blob](#) -- Obtiene el contenido de un objeto blob  
[ifx\\_update\\_blob](#) -- Actualiza el contenido de un objeto blob  
[ifx\\_blobinfile\\_mode](#) -- Define el modo por defecto para los blob en todas las consultas de selección  
[ifx\\_textasvarchar](#) -- Define el modo por defecto para los campos de tipo text  
[ifx\\_byteasvarchar](#) -- Define el modo por defecto para los campos de tipo byte  
[ifx\\_nullformat](#) -- Define el valor por defecto cuando se leen valores nulos  
[ifxus\\_create\\_slob](#) -- Crea un objeto slob y lo abre  
[ifx\\_free\\_slob](#) -- Elimina un objeto slob  
[ifxus\\_close\\_slob](#) -- Cierra un objeto slob  
[ifxus\\_open\\_slob](#) -- Abre un objeto slob  
[ifxus\\_tell\\_slob](#) -- Devuelve el fichero actual o la posición en memoria  
[ifxus\\_seek\\_slob](#) -- Define el fichero o posición en memoria  
[ifxus\\_read\\_slob](#) -- Lee un número de bytes (nbytes) de un objeto slob  
[ifxus\\_write\\_slob](#) -- Escribe una cadena en un objeto slob

# XXXIX. Funciones InterBase

## Tabla de contenidos

[ibase\\_connect](#) --

[ibase\\_pconnect](#) --

[ibase\\_close](#) --

[ibase\\_query](#) --

[ibase\\_fetch\\_row](#) --

[ibase\\_free\\_result](#) --

[ibase\\_prepare](#) --

[ibase\\_bind](#) --

[ibase\\_execute](#) --

[ibase\\_free\\_query](#) --

[ibase\\_timefmt](#) --

# XL. Ingres II functions

## Aviso

This module is *EXPERIMENTAL*. That means, that the behaviour of these functions, these function names, in concreto ANYTHING documented here can change in a future release of PHP WITHOUT NOTICE. Be warned, and use this module at your own risk.

These functions allow you to access Ingres II database servers.

In order to have these functions available, you must compile php with Ingres support by using the `--with-ingres` option. You need the Open API library and header files included with Ingres II. If the `II_SYSTEM` environment variable isn't correctly set you may have to use `--with-ingres=DIR` to specify your Ingres installation directory.

When using this extension with Apache, if Apache does not start and complains with "PHP Fatal error: Unable to start ingres\_ii module in Unknown on line 0" then make sure the environment variable `II_SYSTEM` is correctly set. Adding "export `II_SYSTEM="/home/ingres/II"` in the script that starts Apache, just before launching httpd, should be fine.

**Nota:** If you already used PHP extensions to access other database servers, note that Ingres doesn't allow concurrent queries and/or transaction over one connection, thus you won't find any result or transaction handle in this extension. The result of a query must be treated before sending another query, and a transaction must be committed or rolled back before opening another transaction (which is automatically done when sending the first query).

## Tabla de contenidos

[ingres\\_connect](#) -- Open a connection to an Ingres II database.

[ingres\\_pconnect](#) -- Open a persistent connection to an Ingres II database.

[ingres\\_close](#) -- Close an Ingres II database connection

[ingres\\_query](#) -- Send a SQL query to Ingres II

[ingres\\_num\\_rows](#) -- Get the number of rows affected or returned by the last query



[ingres\\_num\\_fields](#) -- Get the number of fields returned by the last query  
[ingres\\_field\\_name](#) -- Get the name of a field in a query result.  
[ingres\\_field\\_type](#) -- Get the type of a field in a query result.  
[ingres\\_field\\_nullable](#) -- Test if a field is nullable.  
[ingres\\_field\\_length](#) -- Get the length of a field.  
[ingres\\_field\\_precision](#) -- Get the precision of a field.  
[ingres\\_field\\_scale](#) -- Get the scale of a field.  
[ingres\\_fetch\\_array](#) -- Fetch a row of result into an array.  
[ingres\\_fetch\\_row](#) -- Fetch a row of result into an enumerated array.  
[ingres\\_fetch\\_object](#) -- Fetch a row of result into an object.  
[ingres\\_rollback](#) -- Roll back a transaction.  
[ingres\\_commit](#) -- Commit a transaction.  
[ingres\\_autocommit](#) -- Switch autocommit on or off.

---

[Anterior](#)

ibase\_timefmt

[Inicio](#)

[Subir](#)

[Siguiete](#)

ingres\_connect

# XLI. IRC Gateway Functions

Internet Relay Chat Gateway ...

Based on [IRCG](#), from Sascha Schumann.

## Tabla de contenidos

[ircg\\_pconnect](#) -- Connect to an IRC server

[ircg\\_fetch\\_error\\_msg](#) -- Returns the error from previous ircg operation

[ircg\\_set\\_current](#) -- Set current connection for output

[ircg\\_join](#) -- Join a channel on a connected server

[ircg\\_part](#) -- Leave a channel on server

[ircg\\_msg](#) -- Send message to channel or user on server

[ircg\\_notice](#) -- Send a notice to a user on server

[ircg\\_nick](#) -- Change nickname on server

[ircg\\_topic](#) -- Set topic for channel on server

[ircg\\_channel\\_mode](#) -- Set channel mode flags for user

[ircg\\_html\\_encode](#) -- Encodes HTML preserving output

[ircg\\_whois](#) -- Query user information for nick on server

[ircg\\_kick](#) -- Kick a user out of a channel on server

[ircg\\_ignore\\_add](#) -- Add a user to your ignore list on a server

[ircg\\_ignore\\_del](#) -- Remove a user from your ignore list on a server

[ircg\\_disconnect](#) -- Close connection to server

[ircg\\_is\\_conn\\_alive](#) -- Check connection status

[ircg\\_lookup\\_format\\_messages](#) -- Select a set of format strings for display of IRC messages

[ircg\\_register\\_format\\_messages](#) -- Register a set of format strings for display of IRC messages

## XLII. Java

There are two possible ways to bridge PHP and Java: you can either integrate PHP into a Java Servlet environment, which is the more stable and efficient solution, or integrate Java support into PHP. The former is provided by a SAPI module that interfaces with the Servlet server, the latter by the Java extension.

PHP 4 ext/java provides a simple and effective means for creating and invoking methods on Java objects from PHP. The JVM is created using JNI, and everything runs in-process. Build instructions for ext/java can be found in `php4/ext/java/README`.

### Ejemplo 1. Java Example

```
<?php
// get instance of Java class java.lang.System in PHP
$system = new Java('java.lang.System');

// demonstrate property access
print 'Java version=' . $system->getProperty('java.version') . ' <br>';
print 'Java vendor=' . $system->getProperty('java.vendor') . ' <br>';
print 'OS=' . $system->getProperty('os.name') . ' ' .
      $system->getProperty('os.version') . ' on ' .
      $system->getProperty('os.arch') . ' <br>';

// java.util.Date example
$formatter = new Java('java.text.SimpleDateFormat',
                    "EEEE, MMMM dd, yyyy 'at' h:mm:ss a zzzz");

print $formatter->format(new Java('java.util.Date'));
?>
```

## Ejemplo 2. AWT Example

```
<?php
// This example is only intended to be run as a CGI.

$frame = new Java('java.awt.Frame', 'PHP');
$button = new Java('java.awt.Button', 'Hello Java World!');

$frame->add('North', $button);
$frame->validate();
$frame->pack();
$frame->visible = True;

$thread = new Java('java.lang.Thread');
$thread->sleep(10000);

$frame->dispose();
?>
```

### Notes:

- `new Java()` will create an instance of a class if a suitable constructor is available. If no parameters are passed and the default constructor is useful as it provides access to classes like `java.lang.System` which expose most of their functionality through static methods.
- Accessing a member of an instance will first look for bean properties then public fields. In other words, `print $date.time` will first attempt to be resolved as `$date.getTime()`, then as `$date.time`.
- Both static and instance members can be accessed on an object with the same syntax. Furthermore, if the java object is of type `java.lang.Class`, then static members of the class (fields and methods) can be accessed.
- Exceptions raised result in PHP warnings, and **NULL** results. The warnings may be eliminated by prefixing the method call with an "@" sign. The following APIs may be used to retrieve and reset the last error:
  - [java\\_last\\_exception\\_get\(\)](#)
  - [java\\_last\\_exception\\_clear\(\)](#)
- Overload resolution is in general a hard problem given the differences in types between the two languages. The PHP Java extension employs a simple, but fairly effective, metric for determining which overload is the best match.

Additionally, method names in PHP are not case sensitive, potentially increasing the number of overloads to select from.

Once a method is selected, the parameters are coerced if necessary, possibly with a loss of data (example: double precision floating point numbers will be converted to boolean).

- In the tradition of PHP, arrays and hashtables may pretty much be used interchangeably. Note that hashtables in PHP may only be indexed by integers or strings; and that arrays of primitive types in Java can not be sparse. Also note that these constructs are passed by value, so may be expensive in terms of memory and time.

sapi/servlet builds upon the mechanism defined by ext/java to enable the entire PHP processor to be run as a servlet. The primary advantage of this from a PHP perspective is that web servers which support servlets typically take great care in pooling and reusing JVMs. Build instructions for the Servlet SAPI module can be found in `php4/sapi/README`. Notes:

- While this code is intended to be able to run on any servlet engine, it has only been tested on Apache's Jakarta/tomcat to date. Bug reports, success stories and/or patches required to get this code to run on other engines would be appreciated.
- PHP has a habit of changing the working directory. sapi/servlet will eventually change it back, but while PHP is running the servlet engine may not be able to load any classes from the CLASSPATH which are specified using a relative directory syntax, or find the work directory used for administration and JSP compilation tasks.

### Tabla de contenidos

[java\\_last\\_exception\\_clear](#) -- Clear last Java exception

[java\\_last\\_exception\\_get](#) -- Get last Java exception

---

[Anterior](#)

ircg\_register\_format\_messages

[Inicio](#)

[Subir](#)

[Siguiete](#)

java\_last\_exception\_clear

## XLIII. Funciones LDAP

### Introducción a LDAP

LDAP es el protocolo de acceso a directorios ligero (Lightweight Directory Access Protocol), un protocolo usado para acceder a "Servidores de Directorio". El directorio es una clase especial de base de datos que contiene información estructurada en forma de árbol.

El concepto es similar a la estructura de directorios de los discos duros, pero en este caso, el directorio raíz es "El Mundo" y los subdirectorios de primer nivel son los "países". Niveles inferiores de la estructura de directorio contienen entradas para compañías, organizaciones o lugares, y en niveles aún inferiores se encuentran las entradas para la gente, y quizás de equipos informáticos y documentos.

Para referirse a un fichero en un subdirectorio del disco duro se usa algo como

```
/usr/local/misapps/docs
```

Las barras marcan cada división en la referencia al fichero, y la secuencia es leída de izquierda a derecha.

El equivalente a la referencia a un fichero en LDAP es el "distinguished name" (nombre distinguible), abreviado como "dn". Un ejemplo de dn podría ser.

```
cn=Pedro Pérez,ou=Contabilidad,o=Mi Compañía,c=ES
```

Las comas marcan cada división en la referencia, y la secuencia se lee de derecha a izquierda. Este dn se leería como ..

```
country = ES  
organization = Mi Compañía  
organizationalUnit = Contabilidad  
commonName = Pedro Pérez
```

De la misma manera que no hay reglas estrictas sobre como organizar la estructura de directorios de un disco duro, un administrador de un servidor de directorio puede establecer cualquier estructura que sea útil para sus propósitos. Sin embargo hay algunos acuerdos tácitos que siempre deben seguirse. El mensaje es que no se puede escribir código para acceder un directorio si no se conoce algo de su estructura, igual que no se puede usar una base de datos sin algún conocimiento sobre lo que está disponible en ella.

### Ejemplo de código completo

Recuperar información para todas las entradas donde el apellido empiece por "P" de un servidor de directorio, mostrando un extracto con el nombre y dirección de correo electrónico.

### Ejemplo 1. ejemplo de búsqueda LDAP

```
<?php
// La secuencia básica para trabajar con LDAP es conectar, autenticarse,
// buscar, interpretar el resultado de la búsqueda y cerrar la conexión.

echo "<h3>Prueba de consulta LDAP</h3>";
echo "Conectando ...";
$ds=ldap_connect("localhost"); // Debe ser un servidor LDAP válido!
echo "El resultado de la conexión es ".$ds."<p>";

if ($ds) {
    echo "Autenticandose ...";
    $r=ldap_bind($ds); // Autenticación anónima, típicamente con
                      // acceso de lectura
    echo "El resultado de la autenticación es ".$r."<p>";

    echo "Buscando (sn=P*) ...";
    // Búsqueda de entradas por apellidos
    $sr=ldap_search($ds,"o=Mi Compañía, c=ES", "sn=P*");
    echo "El resultado de la búsqueda es ".$sr."<p>";

    echo "El número de entradas devueltas es ".ldap_count_entries($ds,$sr)."<p>";

    echo "Recuperando entradas ...<p>";
    $info = ldap_get_entries($ds, $sr);
    echo "Devueltos datos de ".$info["count"]." entradas:<p>";

    for ($i=0; $i<$info["count"]; $i++) {
        echo "dn es: ". $info[$i]["dn"] ."<br>";
        echo "La primera entrada cn es: ". $info[$i]["cn"][0] ."<br>";
        echo "La primera entrada email es: ". $info[$i]["mail"][0] ."<p>";
    }

    echo "Cerrando conexión";
    ldap_close($ds);
} else {
    echo "<h4>Ha sido imposible conectar al servidor LDAP</h4>";
}
?>
```

## Usando las llamadas LDAP de PHP

Es necesario conseguir y compilar la librerías cliente de LDAP ya sea del paquete ldap-3.3 de la Universidad de Michigan o del Netscape Directory SDK. También es necesario recompilar PHP con soporte LDAP activado para que las funciones LDAP de PHP funcionen.

Antes de usarse las llamadas LDAP se debe saber ..

- El nombre o dirección del servidor de directorio que se va a usar
- El "dn base" del servidor (la parte del directorio global contenida en ese servidor, que puede ser por ejemplo "o=Mi Compañía,c=ES")
- Si es necesaria contraseña para acceder al servidor (muchos servidores ofrecen acceso de lectura para usuarios anónimos pero requieren un password para cualquier otro acceso)

La secuencia típica de llamadas LDAP suele implementarse en aplicaciones que siguen el siguiente patrón:

```
ldap_connect() // establecer la conexión con el servidor
|
ldap_bind()    // login anónimo o autenticado
|
Hacer búsquedas o actualizaciones en el directorio
y mostrar los resultados
|
ldap_close()   // Cerrar la conexión
```

## Más información

Mucha información acerca de LDAP puede ser consultada en

- [Netscape](#)
- [Universidad de Michigan](#)
- [Proyecto OpenLDAP](#)
- [LDAP World](#)

El SDK de Netscape contiene una Guía de Programación muy útil en formato html.

### Tabla de contenidos

[ldap\\_add](#) -- Añade entradas a un directorio LDAP  
[ldap\\_mod\\_add](#) -- Añade valores de atributos  
[ldap\\_mod\\_del](#) -- Borra valores de atributos  
[ldap\\_mod\\_replace](#) -- Reemplaza valores de atributos  
[ldap\\_bind](#) -- Autentifica en un directorio LDAP  
[ldap\\_close](#) -- Cierra una conexión a un servidor LDAP  
[ldap\\_connect](#) -- Conecta con un servidor LDAP  
[ldap\\_count\\_entries](#) -- Cuenta el número de entradas de una búsqueda  
[ldap\\_delete](#) -- Borra una entrada de un directorio  
[ldap\\_dn2ufn](#) -- Convierte un dn al formato User Friendly Naming  
[ldap\\_explode\\_dn](#) -- Divide un DN en las partes que le componen  
[ldap\\_first\\_attribute](#) -- Devuelve el primer atributo  
[ldap\\_first\\_entry](#) -- Devuelve el identificador del primer resultado  
[ldap\\_free\\_result](#) -- Libera la memoria que almacena los resultados  
[ldap\\_get\\_attributes](#) -- Obtiene los atributos de una entrada de un resultado de búsqueda  
[ldap\\_get\\_dn](#) -- Obtiene el DN de una entrada de un resultado  
[ldap\\_get\\_entries](#) -- Obtiene todas las entradas de un resultado



[ldap\\_get\\_values](#) -- Obtiene todos los valores de un atributo de una entrada  
[ldap\\_get\\_values\\_len](#) -- Obtiene todos los valores binarios de un atributo de una entrada  
[ldap\\_list](#) -- Búsqueda Single-level (Nivel Único)  
[ldap\\_modify](#) -- Modifica una entrada LDAP  
[ldap\\_next\\_attribute](#) -- Obtiene el siguiente atributo de una entrada  
[ldap\\_next\\_entry](#) -- Obtiene la siguiente entrada de un resultado  
[ldap\\_read](#) -- Lee una entrada  
[ldap\\_search](#) -- Busca en un árbol LDAP  
[ldap\\_unbind](#) -- Hace logout de un directorio LDAP  
[ldap\\_err2str](#) -- Convierte un código numérico de error LDAP en un mensaje.  
[ldap\\_errno](#) -- Devuelve el código numérico de error para el último comando LDAP.  
[ldap\\_error](#) -- Devuelve el mensaje de error del último comando LDAP

---

[Anterior](#)

[Inicio](#)

[Siguiente](#)

java\_last\_exception\_get

[Subir](#)

ldap\_add

# XLIV. Funciones de Correo

The [mail\(\)](#) Funciones que permiten enviar correo.

## Tabla de contenidos

[mail](#) -- Envía correo

# XLV. Funciones matemáticas

## Introducción

Estas funciones matemáticas solo manejan valores dentro de los rangos de los tipos long y double de su ordenador. Si necesita manejar números mayores, pege un vistazo a [funciones matemáticas de precisión arbitraria](#).

## Constantes matemáticas

Los siguientes valores estan definidos como constantes en PHP por la extensión matemática:

**Tabla 1. Constantes matemáticas**

Constante	Valor	Descripción
M_PI	3.14159265358979323846	El valor de $\pi$ (pi)

### Tabla de contenidos

[abs](#) -- Valor absoluto

[acos](#) -- Arco coseno

[asin](#) -- Arco seno

[atan](#) -- Arco tangente

[atan2](#) -- Arco tangente de dos variables

[base\\_convert](#) -- Convierte un número entre bases arbitrarias

[BinDec](#) -- binario a decimal

[ceil](#) -- Redondea fracciones hacia arriba

[cos](#) -- coseno

[DecBin](#) -- decimal a binario

[DecHex](#) -- decimal a hexadecimal

[DecOct](#) -- decimal a octal

[exp](#) -- e elevado a...  
[floor](#) -- redondea fracciones hacia abajo  
[getrandmax](#) -- Muestra el mayor valor aleatorio posible  
[HexDec](#) -- hexadecimal a decimal  
[log](#) -- Logaritmo natural  
[log10](#) -- Logaritmo en base-10  
[max](#) -- encuentra el valor mayor  
[min](#) -- encuentra el valor menor  
[mt\\_rand](#) -- genera un valor aleatorio mejorado  
[mt\\_srand](#) -- Introduce la semilla del generador de números aleatorios mejorado  
[mt\\_getrandmax](#) -- muestra el mayor valor aleatorio posible  
[number\\_format](#) -- formatea un número en grupos de miles  
[OctDec](#) -- octal a decimal  
[pi](#) -- devuelve el valor de pi  
[pow](#) -- expresión exponencial  
[rand](#) -- genera un valor aleatorio  
[round](#) -- Redondea un float.  
[sin](#) -- seno  
[sqrt](#) -- Raíz cuadrada  
[srand](#) -- introduce la semilla del generador de números aleatorios  
[tan](#) -- tangente

---

[Anterior](#)

mail

[Inicio](#)

[Subir](#)

[Siguiete](#)

abs

# XLVI. Multi-Byte String Functions

## Aviso

This module is *EXPERIMENTAL*. That means, that the behaviour of these functions, these function names, in concreto ANYTHING documented here can change in a future release of PHP WITHOUT NOTICE. Be warned, and use this module at your own risk.

## Introduction

### Aviso

This module is EXPERIMENTAL. Function name/API is subject to change. Current conversion filter supports Japanese only.

There are many languages in which all characters can be expressed by single byte. Multi-byte character codes are used to express many characters for many languages. `mbstring` is developed to handle Japanese characters. However, many `mbstring` functions are able to handle character encoding other than Japanese.

A multi-byte character encoding represents single character with consecutive bytes. Some character encoding has shift(escape) sequences to start/end multi-byte character strings. Therefore, a multi-byte character string may be destroyed when it is divided and/or counted unless multi-byte character encoding safe method is used. This module provides multi-byte character safe string functions and other utility functions such as conversion functions.

Since PHP is basically designed for ISO-8859-1, some multi-byte character encoding does not work well with PHP. Therefore, it is important to set `mbstring.internal_encoding` to a character encoding that works with PHP.

### PHP4 Character Encoding Requirements

- Per byte encoding
- Single byte characters in range of 00h-7fh which is compatible with ASCII
- Multi-byte characters without 00h-7fh

These are examples of internal character encoding that works with PHP and does NOT work with PHP.

```
Character encodings work with PHP:  
ISO-8859-*, EUC-JP, UTF-8
```

```
Character encodings do NOT work with PHP:  
JIS, SJIS
```

Character encoding, that does not work with PHP, may be converted with `mbstring`'s HTTP input/output conversion feature/function.

**Nota:** SJIS should not be used for internal encoding unless the reader is familiar with parser/compiler, character encoding and character encoding issues.

**Nota:** If you use database with PHP, it is recommended that you use the same character encoding for both database and `internal_encoding` for ease of use and better performance.

If you are using PostgreSQL, it supports character encoding that is different from backend character encoding. See the PostgreSQL manual for details.

## How to Enable mbstring

`mbstring` is an extended module. You must enable module with `configure` script. Refer to the [Install](#) section for details.

The following configure options are related to `mbstring` module.

- `--enable-mbstring` : Enable `mbstring` functions. This option is required to use `mbstring` functions.
- `--enable-mbstr-enc-trans` : Enable HTTP input character encoding conversion using `mbstring` conversion engine. If this feature is enabled, HTTP input character encoding may be converted to `mbstring.internal_encoding` automatically.

## HTTP Input and Output

HTTP input/output character encoding conversion may convert binary data also. Users are supposed to control character encoding conversion if binary data is used for HTTP input/output.

If `enctype` for HTML form is set to `multipart/form-data`, `mbstring` does not convert character encoding in POST data. If it is the case, strings are needed to be converted to internal character encoding.

- HTTP Input

There is no way to control HTTP input character conversion from PHP script. To disable HTTP input character conversion, it has to be done in `php.ini`.

### Ejemplo 1. Disable HTTP input conversion in `php.ini`

```
;; Disable HTTP Input conversion
mbstring.http_input = pass
```

When using PHP as an Apache module, it is possible to override PHP ini setting per Virtual Host in `httpd.conf` or per directory with `.htaccess`. Refer to the [Configuration](#) section and Apache Manual for details.

- HTTP Output

There are several ways to enable output character encoding conversion. One is using `php.ini`, another is using [ob\\_start\(\)](#) with [mb\\_output\\_handler\(\)](#) as `ob_start` callback function.

**Nota:** For PHP3-i18n users, `mbstring`'s output conversion differs from PHP3-i18n. Character encoding is converted using output buffer.

## Ejemplo 2. php.ini setting example

```
;; Enable output character encoding conversion for all PHP pages

;; Enable Output Buffering
output_buffering    = On

;; Set mb_output_handler to enable output conversion
output_handler      = mb_output_handler
```

## Ejemplo 3. Script example

```
<?php

// Enable output character encoding conversion only for this page

// Set HTTP output character encoding to SJIS
mb_http_output('SJIS');

// Start buffering and specify "mb_output_handler" as
// callback function
ob_start('mb_output_handler');

?>
```

# Supported Character Encoding

Currently, the following character encoding is supported by `mbstring` module. Character encoding may be specified for `mbstring` functions' encoding parameter.

The following character encoding is supported in this PHP extension :

UCS-4, UCS-4BE, UCS-4LE, UCS-2, UCS-2BE, UCS-2LE, UTF-32, UTF-32BE, UTF-32LE, UCS-2LE, UTF-16, UTF-16BE, UTF-16LE, UTF-8, UTF-7, ASCII, EUC-JP, SJIS, eucJP-win, SJIS-win, ISO-2022-JP, JIS, ISO-8859-1, ISO-8859-2, ISO-8859-3, ISO-8859-4, ISO-8859-5, ISO-8859-6, ISO-8859-7, ISO-8859-8, ISO-8859-9, ISO-8859-10, ISO-8859-13, ISO-8859-14, ISO-8859-15, byte2be, byte2le, byte4be, byte4le, BASE64, 7bit, 8bit and UTF7-IMAP.

`php.ini` entry, which accepts encoding name, accepts "auto" and "pass" also. `mbstring` functions, which accepts encoding name, and accepts "auto".

If "pass" is set, no character encoding conversion is performed.

If "auto" is set, it is expanded to "ASCII, JIS, UTF-8, EUC-JP, SJIS".

See also [mb\\_detect\\_order\(\)](#)

**Nota:** "Supported character encoding" does not mean that it works as internal character code.

# php.ini settings

- `mbstring.internal_encoding` defines default internal character encoding.
- `mbstring.http_input` defines default HTTP input character encoding.
- `mbstring.http_output` defines default HTTP output character encoding.
- `mbstring.detect_order` defines default character code detection order. See also [mb\\_detect\\_order\(\)](#).
- `mbstring.substitute_character` defines character to substitute for invalid character encoding.

Web Browsers are supposed to use the same character encoding when submitting form. However, browsers may not use the same character encoding. See [mb\\_http\\_input\(\)](#) to detect character encoding used by browsers.

If `enctype` is set to `multipart/form-data` in HTML forms, `mbstring` does not convert character encoding in POST data. The user must convert them in the script, if conversion is needed.

Although, browsers are smart enough to detect character encoding in HTML. `charset` is better to be set in HTTP header. Change `default_charset` according to character encoding.

## Ejemplo 4. php.ini setting example

```
;; Set default internal encoding
;; Note: Make sure to use character encoding works with PHP
mbstring.internal_encoding = UTF-8 ; Set internal encoding to UTF-8

;; Set default HTTP input character encoding
;; Note: Script cannot change http_input setting.
mbstring.http_input = pass ; No conversion.
mbstring.http_input = auto ; Set HTTP input to auto
; "auto" is expanded to "ASCII,JIS,UTF-8,EUC-JP,SJIS"
mbstring.http_input = SJIS ; Set HTTP2 input to SJIS
mbstring.http_input = UTF-8,SJIS,EUC-JP ; Specify order

;; Set default HTTP output character encoding
mbstring.http_output = pass ; No conversion
mbstring.http_output = UTF-8 ; Set HTTP output encoding to UTF-8

;; Set default character encoding detection order
mbstring.detect_order = auto ; Set detect order to auto
mbstring.detect_order = ASCII,JIS,UTF-8,SJIS,EUC-JP ; Specify order

;; Set default substitute character
mbstring.substitute_character = 12307 ; Specify Unicode value
mbstring.substitute_character = none ; Do not print character
mbstring.substitute_character = long ; Long Example: U+3000,JIS+7E7E
```



### Ejemplo 5. php.ini setting for EUC-JP users

```
;; Disable Output Buffering
output_buffering      = Off

;; Set HTTP header charset
default_charset       = EUC-JP

;; Set HTTP input encoding conversion to auto
mbstring.http_input  = auto

;; Convert HTTP output to EUC-JP
mbstring.http_output = EUC-JP

;; Set internal encoding to EUC-JP
mbstring.internal_encoding = EUC-JP

;; Do not print invalid characters
mbstring.substitute_character = none
```

### Ejemplo 6. php.ini setting for SJIS users

```
;; Enable Output Buffering
output_buffering      = On

;; Set mb_output_handler to enable output conversion
output_handler        = mb_output_handler

;; Set HTTP header charset
default_charset       = Shift_JIS

;; Set http input encoding conversion to auto
mbstring.http_input  = auto

;; Convert to SJIS
mbstring.http_output = SJIS

;; Set internal encoding to EUC-JP
mbstring.internal_encoding = EUC-JP

;; Do not print invalid characters
mbstring.substitute_character = none
```

## Basics for Japanese multi-byte character

Most Japanese characters need more than 1 byte per character. In addition, several character encoding schemas are used under a Japanese environment. There are EUC-JP, Shift\_JIS(SJIS) and ISO-2022-JP(JIS) character encoding. As Unicode becomes popular, UTF-8 is used also. To develop Web applications for a Japanese environment, it is important to use the character set for the task in hand, whether HTTP input/output, RDBMS and E-mail.

- Storage for a character can be up to six bytes

- A multi-byte character is usually twice of the width compared to single-byte characters. Wider characters are called "zen-kaku" - meaning full width, narrower characters are called "han-kaku" - meaning half width. "zen-kaku" characters are usually fixed width.
- Some character encoding defines shift(escape) sequence for entering/exiting multi-byte character strings.
- ISO-2022-JP must be used for SMTP/NNTP.
- "i-mode" web site is supposed to use SJIS.

## References

Multi-byte character encoding and its related issues are very complex. It is impossible to cover in sufficient detail here. Please refer to the following URLs and other resources for further readings.

- Unicode/UTF/UCS/etc

<http://www.unicode.org/>

- Japanese/Korean/Chinese character information

<ftp://ftp.ora.com/pub/examples/nutshell/ujip/doc/cjk.inf>

### Tabla de contenidos

[mb\\_language](#) -- Set/Get current language

[mb\\_parse\\_str](#) -- Parse GET/POST/COOKIE data and set global variable

[mb\\_internal\\_encoding](#) -- Set/Get internal character encoding

[mb\\_http\\_input](#) -- Detect HTTP input character encoding

[mb\\_http\\_output](#) -- Set/Get HTTP output character encoding

[mb\\_detect\\_order](#) -- Set/Get character encoding detection order

[mb\\_substitute\\_character](#) -- Set/Get substitution character

[mb\\_output\\_handler](#) -- Callback function converts character encoding in output buffer

[mb\\_preferred\\_mime\\_name](#) -- Get MIME charset string

[mb\\_strlen](#) -- Get string length

[mb\\_strpos](#) -- Find position of first occurrence of string in a string

[mb\\_strrpos](#) -- Find position of last occurrence of a string in a string

[mb\\_substr](#) -- Get part of string

[mb\\_strcut](#) -- Get part of string

[mb\\_strwidth](#) -- Return width of string

[mb\\_strimwidth](#) -- Get truncated string with specified width

[mb\\_convert\\_encoding](#) -- Convert character encoding

[mb\\_detect\\_encoding](#) -- Detect character encoding

[mb\\_convert\\_kana](#) -- Convert "kana" one from another ("zen-kaku", "han-kaku" and more)

[mb\\_encode\\_mimeheader](#) -- Encode string for MIME header

[mb\\_decode\\_mimeheader](#) -- Decode string in MIME header field

[mb\\_convert\\_variables](#) -- Convert character code in variable(s)

[mb\\_encode\\_numericentity](#) -- Encode character to HTML numeric string reference

[mb\\_decode\\_numericentity](#) -- Decode HTML numeric string reference to character

[mb\\_send\\_mail](#) -- Send encoded mail.

---

[Anterior](#)

tan

[Inicio](#)

[Subir](#)

[Siguiete](#)

mb\_language

## XLVII. MCAL functions

MCAL stands for Modular Calendar Access Library.

Libmcal is a C library for accessing calendars. It's written to be very modular, with pluggable drivers. MCAL is the calendar equivalent of the IMAP module for mailboxes.

With mcal support, a calendar stream can be opened much like the mailbox stream with the IMAP support. Calendars can be local file stores, remote ICAP servers, or other formats that are supported by the mcal library.

Calendar events can be pulled up, queried, and stored. There is also support for calendar triggers (alarms) and reoccurring events.

With libmcal, central calendar servers can be accessed and used, removing the need for any specific database or local file programming.

To get these functions to work, you have to compile PHP with `--with-mcal`. That requires the mcal library to be installed. Grab the latest version from <http://mcal.chek.com/> and compile and install it.

The following constants are defined when using the MCAL module: MCAL\_SUNDAY, MCAL\_MONDAY, MCAL\_TUESDAY, MCAL\_WEDNESDAY, MCAL\_THURSDAY, MCAL\_FRIDAY, MCAL\_SATURDAY, MCAL\_RECUR\_NONE, MCAL\_RECUR\_DAILY, MCAL\_RECUR\_WEEKLY, MCAL\_RECUR\_MONTHLY\_MDAY, MCAL\_RECUR\_MONTHLY\_WDAY, MCAL\_RECUR\_YEARLY, MCAL\_JANUARY, MCAL\_FEBRUARY, MCAL\_MARCH, MCAL\_APRIL, MCAL\_MAY, MCAL\_JUNE, MCAL\_JULY, MCAL\_AUGUST, MCAL\_SEPTEMBER, MCAL\_OCTOBER, MCAL\_NOVEMBER, and MCAL\_DECEMBER. Most of the functions use an internal event structure that is unique for each stream. This alleviates the need to pass around large objects between functions. There are convenience functions for setting, initializing, and retrieving the event structure values.

**Tabla de contenidos**

[mcald\\_open](#) -- Opens up an MCAL connection

[mcald\\_close](#) -- Close an MCAL stream

[mcald\\_fetch\\_event](#) -- Fetches an event from the calendar stream

[mcald\\_list\\_events](#) -- Return a list of events between two given datetimes

[mcald\\_append\\_event](#) -- Store a new event into an MCAL calendar

[mcald\\_store\\_event](#) -- Modify an existing event in an MCAL calendar

[mcald\\_delete\\_event](#) -- Delete an event from an MCAL calendar

[mcald\\_snooze](#) -- Turn off an alarm for an event

[mcald\\_list\\_alarms](#) -- Return a list of events that has an alarm triggered at the given datetime

[mcald\\_event\\_init](#) -- Initializes a streams global event structure

[mcald\\_event\\_set\\_category](#) -- Sets the category of the streams global event structure

[mcald\\_event\\_set\\_title](#) -- Sets the title of the streams global event structure

[mcald\\_event\\_set\\_description](#) -- Sets the description of the streams global event structure

[mcald\\_event\\_set\\_start](#) -- Sets the start date and time of the streams global event structure

[mcald\\_event\\_set\\_end](#) -- Sets the end date and time of the streams global event structure

[mcald\\_event\\_set\\_alarm](#) -- Sets the alarm of the streams global event structure

[mcald\\_event\\_set\\_class](#) -- Sets the class of the streams global event structure

[mcald\\_is\\_leap\\_year](#) -- Returns if the given year is a leap year or not

[mcald\\_days\\_in\\_month](#) -- Returns the number of days in the given month

[mcald\\_date\\_valid](#) -- Returns **TRUE** if the given year, month, day is a valid date

[mcald\\_time\\_valid](#) -- Returns **TRUE** if the given year, month, day is a valid time

[mcald\\_day\\_of\\_week](#) -- Returns the day of the week of the given date

[mcald\\_day\\_of\\_year](#) -- Returns the day of the year of the given date

[mcald\\_date\\_compare](#) -- Compares two dates

[mcald\\_next\\_recurrence](#) -- Returns the next recurrence of the event

[mcald\\_event\\_set\\_recur\\_none](#) -- Sets the recurrence of the streams global event structure

[mcald\\_event\\_set\\_recur\\_daily](#) -- Sets the recurrence of the streams global event structure

[mcald\\_event\\_set\\_recur\\_weekly](#) -- Sets the recurrence of the streams global event structure

[mcald\\_event\\_set\\_recur\\_monthly\\_mday](#) -- Sets the recurrence of the streams global event structure

[mcald\\_event\\_set\\_recur\\_monthly\\_wday](#) -- Sets the recurrence of the streams global event structure

[mcald\\_event\\_set\\_recur\\_yearly](#) -- Sets the recurrence of the streams global event structure

[mcald\\_fetch\\_current\\_stream\\_event](#) -- Returns an object containing the current streams event structure

[mcald\\_event\\_add\\_attribute](#) -- Adds an attribute and a value to the streams global event structure

---

[Anterior](#)

mb\_send\_mail

[Inicio](#)

[Subir](#)

[Siguiete](#)

mcal\_open

## XLVIII. Funciones Criptográficas

Estas funciones trabajan usando [mcrypt](#).

Esta es una interfaz a la librería mcrypt, que soporta una gran variedad de algoritmos de bloque como DES, TripleDES, Blowfish (por defecto), 3-WAY, SAFER-SK64, SAFER-SK128, TWOFISH, TEA, RC2 y GOST en los modos de cifrado CBC, OFB, CFB y ECB. Adicionalmente, soporta RC6 e IDEA que se consideran "no-libres".

Para usarlos, descarga libmcrypt-x.x.tar.gz de [aquí](#) y sigue las instrucciones de instalación incluidas. Necesitas compilar PHP con el parámetro `--with-mcrypt` para activar esta extensión.

mcrypt puede usarse para encriptar y desencriptar usando los cifrados mencionados arriba. Los cuatro comandos importantes de mcrypt ([mcrypt\\_cfb\(\)](#), [mcrypt\\_cbc\(\)](#), [mcrypt\\_ecb\(\)](#), y [mcrypt\\_ofb\(\)](#)) pueden operar en ambos modos que se llaman MCRYPT\_ENCRYPT y MCRYPT\_DECRYPT, respectivamente.

### Ejemplo 1. Encripta un valor de entrada con TripleDES en modo ECB

```
<?php
$key = "esta es una clave muy secreta";
$input = "Nos vemos a las 9 en punto en el lugar secreto.";

$encrypted_data = mcrypt_ecb(MCRYPT_TripleDES, $key, $input, MCRYPT_ENCRYPT);
?>
```

Este ejemplo devolverá los datos encriptados como una cadena en `$encrypted_data`.

mcrypt puede operar en cuatro modos de cifrado (CBC, OFB, CFB y ECB). Perfilaremos el uso normal de cada uno de estos modos. Para una mejor referencia y una discusión más completa ver Applied Cryptography by Schneier (ISBN 0-471-11709-9).

- ECB (electronic codebook o libro de códigos electrónico) va bien para datos aleatorios, tales como encriptar otras claves. Puesto que los datos son cortos y aleatorios, las desventajas de ECB tienen un efecto negativo favorable.
- CBC (cipher block chaining o cifrado en bloque encadenado) es especialmente útil para encriptar ficheros, donde incrementa significativamente la seguridad por encima de ECB.
- CFB (cipher feedback o cifrado realimentado) es el mejor modo de encriptar flujos de bytes donde cada byte debe ser encriptado.
- OFB (output feedback o salida realimentada) es comparable al CFB, pero puede usarse en aplicaciones donde la propagación de errores no puede tolerarse.

Actualmente PHP no soporta el encriptado/desencriptado de flujos de bits. Por ahora, sólo soporta el manejo de cadenas.

Para una lista completa de los cifrados soportados, ver las definiciones al final de mcrypt.h. La regla general es que se puede acceder al cifrado desde PHP con MCRYPT\_nombredelcifrado.

Aquí hay una pequeña lista de los cifrados que están soportados actualmente por la extensión mcrypt. Si un cifrado no está listado aquí, pero está listado por mcrypt como soportado, puedes asumir con seguridad que esta documentación está caduca.

- MCRYPT\_BLOWFISH
- MCRYPT\_DES
- MCRYPT\_TriplesDES
- MCRYPT\_ThreeWAY
- MCRYPT\_GOST
- MCRYPT\_CRYPT
- MCRYPT\_DES\_COMPAT
- MCRYPT\_SAFER64
- MCRYPT\_SAFER128
- MCRYPT\_CAST128
- MCRYPT\_TEAN
- MCRYPT\_RC2
- MCRYPT\_TWOFISH (para las antiguas versiones mcrypt 2.x)
- MCRYPT\_TWOFISH128 (TWOFISHxxx está disponible en las versiones más nuevas 2.x)
- MCRYPT\_TWOFISH192
- MCRYPT\_TWOFISH256
- MCRYPT\_RC6
- MCRYPT\_IDEA

Debes (en los modos CFB y OFB) o puedes (en el modo CBC) suministrar un vector de inicialización (IV) a la correspondiente función de cifrado. El IV debe ser único y debe ser el mismo cuando descriptas o encriptas. Con datos que son guardados encriptados, puedes cojer la salida de una función de índice bajo la cual los datos son almacenados (ej. la clave MD5 de un fichero). Alternativamente, puedes transmitir el IV junto con los datos encriptados (ver capítulo 9.3 de Applied Cryptography by Schneier (ISBN 0-471-11709-9) para una discusión de éste asunto).

### Tabla de contenidos

[mcrypt\\_get\\_cipher\\_name](#) -- Obtiene el nombre del cifrado especificado

[mcrypt\\_get\\_block\\_size](#) -- Obtiene el tamaño de bloque del cifrado indicado

[mcrypt\\_get\\_key\\_size](#) -- Obtiene el tamaño de la clave de un cifrado

[mcrypt\\_create\\_iv](#) -- Crea un vector de inicialización (IV) a partir de una fuente aleatoria

[mcrypt\\_cbc](#) -- Encripta/descripta datos en modo CBC

[mcrypt\\_cfb](#) -- Encripta/descripta datos en modo CFB

[mcrypt\\_ecb](#) -- Encripta/descripta datos en modo ECB

[mcrypt\\_ofb](#) -- Encripta/descripta datos en modo OFB

---

[Anterior](#)

mcrypt\_event\_add\_attribute

[Inicio](#)

[Subir](#)

[Siguiete](#)

mcrypt\_get\_cipher\_name

## XLIX. Funciones Hash

Estas funciones han sido realizadas para trabajar con [mhash](#).

Esta es una interfaz con al libreria mhash. mhash soporta una amplia variedad de algoritmos hash como MD5, SHA1, GOST, y muchos otros.

Para usarla, hay que descargar la distribucion desde [su sitio web](#) y seguir las intrucciones de instalacion. Se necesita compilar PHP con el parametr `--with-mhash` para activar esta extension.

mhash puede ser usado para crear checksums, message digests, y mas.

### Ejemplo 1. Generar una clave SHA1 e imprimirla en hexadecimal

```
<?php
$input = "Let us meet at 9 o' clock at the secret place.";
$hash = mhash(MHASH_SHA1, $input);

print "The hash is ".bin2hex($hash)."\n";

?>
```

Esto generara:

```
The hash is d3b85d710d8f6e4e5efd4d5e67d041f9cecedafe
```

Para una lista compera de hash soportados, refierase a la documentacion de mhash. La regla general es que se puede acceder a los algoritmos hash desde PHP con `MHASH_HASHNAME`. Como ejemplo, para acceder a `HVAL` se debe usar la constante de PHP llamada `MHASH_HVAL`.

Aqui hay una lista de hashes que esta actualmente soportada por mhash. Si un hash no esta en



dicha lista pero aparece como soportado por mhash, entonces se asume con plena seguridad que esta documentacion esta desfasada.

- MHASH\_MD5
- MHASH\_SHA1
- MHASH\_HAVAL
- MHASH\_RIPEMD160
- MHASH\_RIPEMD128
- MHASH\_SNEFRU
- MHASH\_TIGER
- MHASH\_GOST
- MHASH\_CRC32
- MHASH\_CRC32B

### **Tabla de contenidos**

[mhash\\_get\\_hash\\_name](#) -- Conseguir el nombre de un hash especifico

[mhash\\_get\\_block\\_size](#) -- Conseguir el tamaño de bloque de el hash especificado

[mhash\\_count](#) -- Obtener el valor mayor del id hash disponible

[mhash](#) -- Calcular el hash

---

[Anterior](#)

mcrypt\_ofb

[Inicio](#)

[Subir](#)

[Siguiete](#)

mhash\_get\_hash\_name

# L. Funciones de Microsoft SQL Server

## Tabla de contenidos

[mssql\\_close](#) -- cierra una conexión con MS SQL Server

[mssql\\_connect](#) -- abre una conexión con MS SQL server

[mssql\\_data\\_seek](#) -- mueve el puntero interno de las filas

[mssql\\_fetch\\_array](#) -- Captura la fila en un array

[mssql\\_fetch\\_field](#) -- obtiene la información de los campos

[mssql\\_fetch\\_object](#) -- captura la fila como un objeto

[mssql\\_fetch\\_row](#) -- obtiene la fila como un array numerado

[mssql\\_field\\_seek](#) -- set field offset

[mssql\\_free\\_result](#) -- libera de la memoria el resultado de una consulta

[mssql\\_num\\_fields](#) -- obtiene el número de campos de la consulta

[mssql\\_num\\_rows](#) -- obtiene el número de filas de la consulta

[mssql\\_pconnect](#) -- abre una conexión persistente con MS SQL

[mssql\\_query](#) -- envia una consulta MS SQL

[mssql\\_result](#) -- get result data

[mssql\\_select\\_db](#) -- selecciona una base de datos MS SQL

# LI. Ming functions for Flash

## Aviso

This module is *EXPERIMENTAL*. That means, that the behaviour of these functions, these function names, in concreto ANYTHING documented here can change in a future release of PHP WITHOUT NOTICE. Be warned, and use this module at your own risk.

## Introduction

Ming is an open-source (LGPL) library which allows you to create SWF ("Flash") format movies. Ming supports almost all of Flash 4's features, including: shapes, gradients, bitmaps (pngs and jpegs), morphs ("shape tweens"), text, buttons, actions, sprites ("movie clips"), streaming mp3, and color transforms--the only thing that's missing is sound events.

Ming is not an acronym.

Note that all values specifying length, distance, size, etc. are in "twips", twenty units per pixel. That's pretty much arbitrary, though, since the player scales the movie to whatever pixel size is specified in the embed/object tag, or the entire frame if not embedded.

Ming offers a number of advantages over the existing PHP/libswf module. You can use Ming anywhere you can compile the code, whereas libswf is closed-source and only available for a few platforms, Windows not one of them. Ming provides some insulation from the mundane details of the SWF file format, wrapping the movie elements in PHP objects. Also, Ming is still being maintained; if there's a feature that you want to see, just let us know [ming@opaque.net](mailto:ming@opaque.net).

Ming was added in PHP 4.0.5.

## Installation

To use Ming with PHP, you first need to build and install the Ming library. Source code and installation instructions are available at the Ming home page : <http://www.opaque.net/ming/> along with examples, a small tutorial, and the latest news.

Download the ming archive. Unpack the archive. Go in the Ming directory. make. make install.

This will build `libming.so` and install it into `/usr/lib/`, and copy `ming.h` into `/usr/include/`. Edit the `PREFIX=` line in the `Makefile` to change the installation directory.

## built into php (unix)

```
mkdir <phpdir>/ext/ming
cp php_ext/* <phpdir>/ext/ming
cd <phpdir>
./buildconf
./configure --with-ming <other config options>
```

Build and install php as usual, Restart web server if necessary

## built into php (unix)

download `php_ming.so.gz`. uncompress it and copy it to your php modules directory. (you can find your php module directory by running **`php-config --extension-dir`**). Now either just add `extension=php_ming.so` to your `php.ini` file, or put `dl('php_ming.so');` at the head of all of your Ming scripts.

## How to use Ming

Ming introduces 13 new object in PHP, all with matching methods and attributes. To use them, you need to know about [objects](#).

- [swfmovie\(\)](#).
- [swfshape\(\)](#).

- [swfdisplayitem\(\)](#).
- [swfgradient\(\)](#).
- [swfbitmap\(\)](#).
- [swffill\(\)](#).
- [swfmorph\(\)](#).
- [swftext\(\)](#).
- [swffont\(\)](#).
- [swftextfield\(\)](#).
- [swfsprite\(\)](#).
- [swfbutton\(\)](#).
- [swfaction\(\)](#).

## Tabla de contenidos

[SWFMovie](#) -- Creates a new movie object, representing an SWF version 4 movie.

[SWFMovie->output](#) -- Dumps your lovingly prepared movie out.

[SWFMovie->save](#) -- Saves your movie in a file.

[SWFMovie->add](#) -- Adds any type of data to a movie.

[SWFMovie->remove](#) -- Removes the object instance from the display list.

[SWFMovie->setbackground](#) -- Sets the background color.

[SWFMovie->setrate](#) -- Sets the animation's frame rate.

[SWFMovie->setdimension](#) -- Sets the movie's width and height.

[SWFMovie->setframes](#) -- Sets the total number of frames in the animation.

[SWFMovie->nextframe](#) -- Moves to the next frame of the animation.

[SWFMovie->streammp3](#) -- Streams a MP3 file.

[SWFDisplayItem](#) -- Creates a new displayitem object.

[SWFDisplayItem->moveTo](#) -- Moves object in global coordinates.

[SWFDisplayItem->move](#) -- Moves object in relative coordinates.

[SWFDisplayItem->scaleTo](#) -- Scales the object in global coordinates.

[SWFDisplayItem->scale](#) -- Scales the object in relative coordinates.

[SWFDisplayItem->rotateTo](#) -- Rotates the object in global coordinates.

[SWFDisplayItem->Rotate](#) -- Rotates in relative coordinates.

[SWFDisplayItem->skewXTo](#) -- Sets the X-skew.

[SWFDisplayItem->skewX](#) -- Sets the X-skew.

[SWFDisplayItem->skewYTo](#) -- Sets the Y-skew.

[SWFDisplayItem->skewY](#) -- Sets the Y-skew.

[SWFDisplayItem->setDepth](#) -- Sets z-order

[SWFDisplayItem->remove](#) -- Removes the object from the movie

[SWFDisplayItem->setName](#) -- Sets the object's name  
[SWFDisplayItem->setRatio](#) -- Sets the object's ratio.  
[SWFDisplayItem->addColor](#) -- Adds the given color to this item's color transform.  
[SWFDisplayItem->multColor](#) -- Multiplies the item's color transform.  
[SWFShape](#) -- Creates a new shape object.  
[SWFShape->setLine](#) -- Sets the shape's line style.  
[SWFShape->addFill](#) -- Adds a solid fill to the shape.  
[SWFShape->setLeftFill](#) -- Sets left rasterizing color.  
[SWFShape->setRightFill](#) -- Sets right rasterizing color.  
[SWFShape->movePenTo](#) -- Moves the shape's pen.  
[SWFShape->movePen](#) -- Moves the shape's pen (relative).  
[SWFShape->drawLineTo](#) -- Draws a line.  
[SWFShape->drawLine](#) -- Draws a line (relative).  
[SWFShape->drawCurveTo](#) -- Draws a curve.  
[SWFShape->drawCurve](#) -- Draws a curve (relative).  
[SWFGradient](#) -- Creates a gradient object  
[SWFGradient->addEntry](#) -- Adds an entry to the gradient list.  
[SWFBitmap](#) -- Loads Bitmap object  
[SWFBitmap->getWidth](#) -- Returns the bitmap's width.  
[SWFBitmap->getHeight](#) -- Returns the bitmap's height.  
[SWFFill](#) -- Loads SWFFill object  
[SWFFill->moveTo](#) -- Moves fill origin  
[SWFFill->scaleTo](#) -- Sets fill's scale  
[SWFFill->rotateTo](#) -- Sets fill's rotation  
[SWFFill->skewXTo](#) -- Sets fill x-skew  
[SWFFill->skewYTo](#) -- Sets fill y-skew  
[SWFMorph](#) -- Creates a new SWFMorph object.  
[SWFMorph->getshape1](#) -- Gets a handle to the starting shape  
[SWFMorph->getshape2](#) -- Gets a handle to the ending shape  
[SWFText](#) -- Creates a new SWFText object.  
[SWFText->setFont](#) -- Sets the current font  
[SWFText->setHeight](#) -- Sets the current font height  
[SWFText->setSpacing](#) -- Sets the current font spacing  
[SWFText->setColor](#) -- Sets the current font color  
[SWFText->moveTo](#) -- Moves the pen  
[SWFText->addString](#) -- Draws a string

[SWFText->getWidth](#) -- Computes string's width  
[SWFFont](#) -- Loads a font definition  
[swffont->getWidth](#) -- Returns the string's width  
[SWFTextField](#) -- Creates a text field object  
[SWFTextField->setFont](#) -- Sets the text field font  
[SWFTextField->setbounds](#) -- Sets the text field width and height  
[SWFTextField->align](#) -- Sets the text field alignment  
[SWFTextField->setHeight](#) -- Sets the font height of this text field font.  
[SWFTextField->setLeftMargin](#) -- Sets the left margin width of the text field.  
[SWFTextField->setrightMargin](#) -- Sets the right margin width of the text field.  
[SWFTextField->setMargins](#) -- Sets the margins width of the text field.  
[SWFTextField->setindentation](#) -- Sets the indentation of the first line.  
[SWFTextField->setLineSpacing](#) -- Sets the line spacing of the text field.  
[SWFTextField->setcolor](#) -- Sets the color of the text field.  
[SWFTextField->setname](#) -- Sets the variable name  
[SWFTextField->addstring](#) -- Concatenates the given string to the text field  
[SWFSprite](#) -- Creates a movie clip (a sprite)  
[SWFSprite->add](#) -- Adds an object to a sprite  
[SWFSprite->remove](#) -- Removes an object to a sprite  
[SWFSprite->setframes](#) -- Sets the total number of frames in the animation.  
[SWFSprite->nextframe](#) -- Moves to the next frame of the animation.  
[SWFbutton](#) -- Creates a new Button.  
[SWFbutton->addShape](#) -- Adds a shape to a button  
[SWFbutton->setUp](#) -- Alias for addShape(shape, SWFBUTTON\_UP)  
[SWFbutton->setOver](#) -- Alias for addShape(shape, SWFBUTTON\_OVER)  
[SWFbutton->setdown](#) -- Alias for addShape(shape, SWFBUTTON\_DOWN))  
[SWFbutton->setHit](#) -- Alias for addShape(shape, SWFBUTTON\_HIT)  
[SWFbutton->addAction](#) -- Adds an action  
[SWFbutton->setAction](#) -- Sets the action  
[SWFAction](#) -- Creates a new Action.

---

[Anterior](#)

mssql\_select\_db

[Inicio](#)

[Subir](#)

[Siguiete](#)

SWFMovie

## LII. Miscelánea de funciones

Estas funciones están colocadas aquí debido a que no parecen ajustarse a ninguna otra categoría.

### Tabla de contenidos

[connection\\_aborted](#) -- Devuelve **TRUE** si el cliente está desconectado

[connection\\_status](#) -- Devuelve el estado de la conexión en un campo de bits

[connection\\_timeout](#) -- Devuelve **TRUE** si el script ha alcanzado su time out

[define](#) -- Define una constante con nombre.

[defined](#) -- Comprueba que una constante con nombre dada existe.

[die](#) -- Envía a la salida un mensaje y finaliza el script actual

[eval](#) -- Evalúa una cadena de caracteres como código PHP

[exit](#) -- Finaliza el script actual

[get\\_browser](#) -- Informa sobre lo que es capaz de hacer el navegador (browser) del usuario.

[ignore\\_user\\_abort](#) -- Establece si la desconexión de un cliente debe suspender la ejecución del script

[iptcparse](#) -- Divide un bloque binario IPTC <http://www.xe.net/iptc/> en su tags (etiquetas) individuales.

[leak](#) -- Filtra memoria

[pack](#) -- empaqueta datos en una cadena binaria

[serialize](#) -- genera una representación almacenable de un valor

[sleep](#) -- Ejecución retardada

[uniqid](#) -- Genera un id único.

[unpack](#) -- desempaqueta datos de una cadena binaria

[unserialize](#) -- crea un valor PHP de una representación almacenada

[usleep](#) -- Retrasa la ejecución, en microsegundos



## LIII. mnoGoSearch Functions

These functions allow you to access mnoGoSearch (former UdmSearch) free search engine. In order to have these functions available, you must compile php with mnogosearch support by using the `--with-mnogosearch` option. If you use this option without specifying the path to mnogosearch, php will look for mnogosearch under `/usr/local/mnogosearch` path by default. If you installed mnogosearch at other path you should specify it: `--with-mnogosearch=DIR`.

mnoGoSearch is a full-featured search engine software for intranet and internet servers, distributed under the GNU license. mnoGoSearch has number of unique features, which makes it appropriate for a wide range of application from search within your site to a specialized search system such as cooking recipes or newspaper search, ftp archive search, news articles search, etc. It offers full-text indexing and searching for HTML, PDF, and text documents. mnoGoSearch consists of two parts. The first is an indexing mechanism (indexer). The purpose of indexer is to walk through HTTP, FTP, NEWS servers or local files, recursively grabbing all the documents and storing meta-data about that documents in a SQL database in a smart and effective manner. After every document is referenced by its corresponding URL, meta-data collected by indexer is used later in a search process. The search is performed via Web interface. C CGI, PHP and Perl search front ends are included.

**Nota:** php contains built-in mysql access library, which can be used to access mysql. It is known that mnoGoSearch is not compatible with this built-in library and can work only with generic mysql libraries. Thus, if you use mnoGoSearch with mysql, during php configuration you have to indicate directory of mysql installation, that was used during mnoGoSearch configuration, i.e. for example: `--with-mnogosearch --with-mysql=/usr`.

You need at least 3.1.10 version of mnoGoSearch installed to use these functions.

More information about mnoGoSearch can be found at <http://www.mnogosearch.ru/>.

**Tabla de contenidos**

[udm\\_add\\_search\\_limit](#) -- Add various search limits  
[udm\\_alloc\\_agent](#) -- Allocate mnoGoSearch session  
[udm\\_api\\_version](#) -- Get mnoGoSearch API version.  
[udm\\_cat\\_path](#) -- Get the path to the current category.  
[udm\\_cat\\_list](#) -- Get all the categories on the same level with the current one.  
[udm\\_clear\\_search\\_limits](#) -- Clear all mnoGoSearch search restrictions  
[udm\\_errno](#) -- Get mnoGoSearch error number  
[udm\\_error](#) -- Get mnoGoSearch error message  
[udm\\_find](#) -- Perform search  
[udm\\_free\\_agent](#) -- Free mnoGoSearch session  
[udm\\_free\\_ispell\\_data](#) -- Free memory allocated for ispell data  
[udm\\_free\\_res](#) -- Free mnoGoSearch result  
[udm\\_get\\_doc\\_count](#) -- Get total number of documents in database.  
[udm\\_get\\_res\\_field](#) -- Fetch mnoGoSearch result field  
[udm\\_get\\_res\\_param](#) -- Get mnoGoSearch result parameters  
[udm\\_load\\_ispell\\_data](#) -- Load ispell data  
[udm\\_set\\_agent\\_param](#) -- Set mnoGoSearch agent session parameters

---

[Anterior](#)

usleep

[Inicio](#)

[Subir](#)

[Siguiete](#)

[udm\\_add\\_search\\_limit](#)

# LIV. funciones mSQL

## Tabla de contenidos

[msql](#) -- ejecuta una consulta mSQL

[msql\\_affected\\_rows](#) -- devuelve el número de filas involucradas

[msql\\_close](#) -- cierra una conexión mSQL

[msql\\_connect](#) -- abre una conexión mSQL

[msql\\_create\\_db](#) -- crea una base de datos mSQL

[msql\\_createdb](#) -- crea una base de datos mSQL

[msql\\_data\\_seek](#) -- desplaza el puntero interno de fila

[msql\\_dbname](#) -- obtiene el nombre de la base de datos mSQL actual

[msql\\_drop\\_db](#) -- elimina (suprime) una base de datos mSQL

[msql\\_dropdb](#) -- elimina (suprime) una base de datos mSQL

[msql\\_error](#) -- devuelve el mensaje de error de la última llamada msql

[msql\\_fetch\\_array](#) -- recupera una fila como un array

[msql\\_fetch\\_field](#) -- obtiene información de campo

[msql\\_fetch\\_object](#) -- recupera una fila como un objeto

[msql\\_fetch\\_row](#) -- obtiene una fila como un array enumerado

[msql\\_fieldname](#) -- obtiene el nombre del campo

[msql\\_field\\_seek](#) -- establece el desplazamiento del campo

[msql\\_fieldtable](#) -- obtiene el nombre de la tabla de un campo

[msql\\_fieldtype](#) -- obtiene el tipo del campo

[msql\\_fieldflags](#) -- obtiene los flags del campo

[msql\\_fieldlen](#) -- obtiene la longitud del campo

[msql\\_free\\_result](#) -- libera la memoria del resultado

[msql\\_freeresult](#) -- libera la memoria del resultado

[msql\\_list\\_fields](#) -- lista los campos del resultado

[msql\\_listfields](#) -- lista los campos del resultado

[msql\\_list\\_dbs](#) -- lista las bases de datos mSQL en el servidor

[msql\\_listdbs](#) -- lista las bases de datos mSQL en el servidor

[msql\\_list\\_tables](#) -- lista las tablas de una base de datos mSQL

[msql\\_listtables](#) -- lista las tablas de una base de datos mSQL

[mysql\\_num\\_fields](#) -- obtiene el número de campos de un resultado  
[mysql\\_num\\_rows](#) -- obtiene el número de filas de un resultado  
[mysql\\_numfields](#) -- obtiene el número de campos de un resultado  
[mysql\\_numrows](#) -- obtiene el número de filas en el resultado  
[mysql\\_pconnect](#) -- abre una conexión mSQL persistente  
[mysql\\_query](#) -- envía una consulta mSQL  
[mysql\\_regcase](#) -- construye una expresión regular para una búsqueda que no distinga mayúsculas/minúsculas  
[mysql\\_result](#) -- obtiene datos resultado  
[mysql\\_select\\_db](#) -- selecciona una base de datos mSQL  
[mysql\\_selectdb](#) -- selecciona una base de datos mSQL  
[mysql\\_tablename](#) -- obtiene el nombre de la tabla de un campo

---

[Anterior](#)

udm\_set\_agent\_param

[Inicio](#)

[Subir](#)

[Siguiete](#)

mysql

# LV. Funciones MySQL

Estas funciones le permiten acceder a servidores de bases de datos MySQL.

Puede encontrar más información sobre MySQL en <http://www.mysql.com/>.

## Tabla de contenidos

[mysql\\_affected\\_rows](#) -- Devuelve el número de filas afectadas de la última operación MySQL

[mysql\\_change\\_user](#) -- Cambia el usuario conectado en la conexión activa

[mysql\\_close](#) -- cierra el enlace con MySQL

[mysql\\_connect](#) -- Abre una conexión a un servidor MySQL

[mysql\\_create\\_db](#) -- Crea una base MySQL

[mysql\\_data\\_seek](#) -- Mueve el puntero interno

[mysql\\_db\\_query](#) -- Envía una sentencia MySQL al servidor

[mysql\\_drop\\_db](#) -- Borra una base de datos MySQL

[mysql\\_errno](#) -- Devuelve el número del mensaje de error de la última operación MySQL

[mysql\\_error](#) -- Devuelve el texto del mensaje de error de la última operación MySQL

[mysql\\_fetch\\_array](#) -- Extrae la fila de resultado como una matriz asociativa

[mysql\\_fetch\\_field](#) -- Extrae la información de una columna y la devuelve como un objeto.

[mysql\\_fetch\\_lengths](#) -- Devuelve la longitud de cada salida en un resultado

[mysql\\_fetch\\_object](#) -- Extrae una fila de resultado como un objeto

[mysql\\_fetch\\_row](#) -- Devuelve una fila de resultado como matriz

[mysql\\_field\\_name](#) -- Devuelve el nombre del campo especificado en un resultado

[mysql\\_field\\_seek](#) -- Asigna el puntero del resultado al offset del campo especificado

[mysql\\_field\\_table](#) -- Devuelve el nombre de la tabla donde esta el campo especificado

[mysql\\_field\\_type](#) -- Devuelve el tipo del campo especificado en un resultado

[mysql\\_field\\_flags](#) -- Devuelve los flags asociados con el campo especificado en un resultado

[mysql\\_field\\_len](#) -- Devuelve la longitud del campo especificado

[mysql\\_free\\_result](#) -- Libera la memoria del resultado

[mysql\\_insert\\_id](#) -- Devuelve el identificador generado en la última llamada a INSERT

[mysql\\_list\\_fields](#) -- Lista los campos del resultado de MySQL

[mysql\\_list\\_dbs](#) -- Lista las bases de datos disponibles en el servidor MySQL  
[mysql\\_list\\_tables](#) -- Lista las tablas en una base de datos MySQL  
[mysql\\_num\\_fields](#) -- devuelve el numero de campos de un resultado  
[mysql\\_num\\_rows](#) -- Devuelve el numero de filas de un resultado  
[mysql\\_pconnect](#) -- Abre una conexión persistente al servidor MySQL  
[mysql\\_query](#) -- Envía una sentencia SQL a MySQL  
[mysql\\_result](#) -- Devuelve datos de un resultado  
[mysql\\_select\\_db](#) -- Selecciona un base de datos MySQL  
[mysql\\_tablename](#) -- Devuelve el nombre de la tabla de un campo

---

[Anterior](#)

mysql\_tablename

[Inicio](#)

[Subir](#)

[Siguiete](#)

mysql\_affected\_rows

# LVI. Funciones de Red

## Tabla de contenidos

[checkdnsrr](#) -- Comprueba registros DNS correspondientes a nombres de máquinas en Internet o direcciones IP.

[closelog](#) -- cierra la conexión con el logger del sistema

[debugger\\_off](#) -- deshabilita el depurador interno de PHP

[debugger\\_on](#) -- habilita el depurador interno de PHP

[fsockopen](#) -- Abre una conexión de dominio Internet o Unix via sockets.

[gethostbyaddr](#) -- Obtiene el nombre de una máquina en Internet mediante su dirección IP.

[gethostbyname](#) -- Obtiene la dirección IP correspondiente al nombre de una máquina conectada a Internet.

[gethostbyname1](#) -- Obtiene una lista de direcciones IP correspondiente a los nombres de máquinas conectadas a Internet.

[getmxrr](#) -- Obtiene registros MX correspondientes a una máquina conectada a Internet.

[getprotobyname](#) -- Obtiene el número asociado al nombre del protocolo

[getprotobynumber](#) -- obtiene el nombre asociado al número de protocolo

[getservbyname](#) -- obtiene el número del puerto asociado al servicio Internet especificado

[getservbyport](#) -- obtiene el servicio Internet que correspondiente al puerto del protocolo especificado

[openlog](#) -- abre una conexión con el logger del sistema

[pfsockopen](#) -- Abre conexiones persistentes de dominio Internet o Unix.

[set\\_socket\\_blocking](#) -- Set blocking/non-blocking modo de un socket

[syslog](#) -- genera un mensaje de sistema

## LVII. ODBC functions

### Tabla de contenidos

- [odbc\\_autocommit](#) -- Interruptor de comportamiento de auto-entrega
- [odbc\\_binmode](#) -- Manejo de campos de datos binarios
- [odbc\\_close](#) -- Cierra una conexion ODBC
- [odbc\\_close\\_all](#) -- Cierra todas las conexiones ODBC
- [odbc\\_commit](#) -- Entrega una transaccion ODBC
- [odbc\\_connect](#) -- Conecta a una fuente de datos
- [odbc\\_cursor](#) -- Toma un nombre de cursor
- [odbc\\_do](#) -- sinonimo de [odbc\\_exec\(\)](#)
- [odbc\\_exec](#) -- Prepara o ejecuta una declaracion SQL
- [odbc\\_execute](#) -- ejecuta una declaracion preparada
- [odbc\\_fetch\\_into](#) -- Busca un registro de resultados dentro de un vector
- [odbc\\_fetch\\_row](#) -- Busca un registro
- [odbc\\_field\\_name](#) -- Devuelve el nombre de campo
- [odbc\\_field\\_type](#) -- Tipo de datos de un campo
- [odbc\\_field\\_len](#) -- Da la longitud de un campo
- [odbc\\_free\\_result](#) -- recursos libres asociados con un resultado
- [odbc\\_longreadlen](#) -- manejo de LONGITUD de columnas
- [odbc\\_num\\_fields](#) -- numero de campos de un resultado
- [odbc\\_pconnect](#) -- Abre una conexion permanente de base de datos
- [odbc\\_prepare](#) -- Prepara una declaracion para su ejecucion
- [odbc\\_num\\_rows](#) -- Numero de campos en un resultado
- [odbc\\_result](#) -- coge informacion de un campo
- [odbc\\_result\\_all](#) -- Print result as HTML table
- [odbc\\_rollback](#) -- Volver a pasar una transacion
- [odbc\\_setoption](#) -- Ajusta la configuracion de ODBC. Devuelve **FALSE** en caso de error, en otro caso **TRUE**.



syslog

[Subir](#)

odbc\_autocommit

## LVIII. Funciones de Oracle 8

Estas funciones permiten acceder a bases de datos Oracle8 y Oracle7. Estas usan la Oracle8 Call-Interface (OCI8). Necesitará las librerías clientes de Oracle8 para usar esta extensión.

Esta extensión es más flexible que las estándar de Oracle. Soporta el enlace de variables locales y globales de PHP con placeholders de Oracle, tiene soporte completo para LOB, FILE y ROWID y le permiten usar las variables definidas por el usuario.

### Tabla de contenidos

[OCIDefineByName](#) -- Usa una variable de PHP para el define-step durante una sentencia SELECT

[OCIBindByName](#) -- Enlaza una variable PHP a un Placeholder de Oracle

[OCILogon](#) -- Establece la conexión con Oracle

[OCIPLogon](#) -- Conecta con una base de datos Oracle usando una conexión persistente. Devuelve una nueva sesión.

[OCINLogon](#) -- Conecta con una base de datos Oracle usando una nueva conexión. Devuelve una nueva sesión.

[OCILogOff](#) -- Termina la conexión con Oracle

[OCIExecute](#) -- Ejecuta una sentencia

[OCICommit](#) -- Confirma transacciones pendientes

[OCIRollback](#) -- Restablece todas las transacciones sin confirmar

[OCINewDescriptor](#) -- Inicializa un nuevo descriptor vacío LOB/FILE (LOB por defecto)

[OCIRowCount](#) -- Obtiene el número de filas afectadas

[OCINumCols](#) -- Devuelve el número de columnas resultantes en una sentencia

[OCIResult](#) -- Devuelve el valor de una columna en la fila buscada

[OCIFetch](#) -- Busca la siguiente fila en el result-buffer

[OCIFetchInto](#) -- Busca la siguiente fila dentro del result-array

[OCIFetchStatement](#) -- Busca todas las filas de un resultset dentro de un array.

[OCIColumnIsNULL](#) -- comprueba si una columna es **NULL**

[OCIColumnSize](#) -- devuelve el tamaño de la columna

[OCI\\_SERVER\\_VERSION](#) -- Devuelve una cadena conteniendo información acerca de la versión del servidor.

[OCIStatementType](#) -- Devuelve el tipo de una sentencia OCI.

[OCINewCursor](#) -- devuelve un cursor nuevo (Statement-Handle) - use esto para enlazar ref-cursors!

[OCIFreeStatement](#) -- Libera todos los recursos asociados con una sentencia.

[OCIFreeCursor](#) -- Libera todos los recursos asociados con cursor.

[OCIColumnName](#) -- Devuelve el nombre de una columna.

[OCIColumnType](#) -- Devuelve el tipo de dato de una columna.

[OCIParse](#) -- Analiza una consulta y devuelve una sentencia

[OCIError](#) -- Devuelve el último error de stmt|conn|global. Si no ocurre ningún error devuelve falso.

[OCIInternalDebug](#) -- Habilita o deshabilita la salida del depurador interno. Por defecto este está deshabilitado

---

[Anterior](#)

odbc\_setoption

[Inicio](#)

[Subir](#)

[Siguiente](#)

OCIDefineByName

# LIX. OpenSSL functions

## Aviso

This module is *EXPERIMENTAL*. That means, that the behaviour of these functions, these function names, in concreto ANYTHING documented here can change in a future release of PHP WITHOUT NOTICE. Be warned, and use this module at your own risk.

## Introduction

This module uses the functions of [OpenSSL](#) for generation and verification of signatures and for sealing (encrypting) and opening (decrypting) data. PHP-4.0.4p11 requires OpenSSL  $\geq$  0.9.6, but PHP-4.0.5 and greater with also work with OpenSSL  $\geq$  0.9.5.

**Nota:** Please keep in mind that this extension is still considered experimental!

OpenSSL offers many features that this module currently doesn't support. Some of these may be added in the future.

## Key/Certificate parameters

Quite a few of the openssl functions require a key or a certificate parameter. PHP 4.0.5 and earlier have to use a key or certificate resource returned by one of the openssl\_get\_XXX functions. Later versions may use one of the following methods:

- Certificates
  1. An X.509 resource returned from openssl\_x509\_read
  2. A string having the format `file://path/to/cert.pem`; the named file must contain a PEM encoded certificate
  3. A string containing the content of a certificate, PEM encoded

- Public/Private Keys

1. A key resource returned from [openssl\\_get\\_publickey\(\)](#) or [openssl\\_get\\_privatekey\(\)](#)
2. For public keys only: an X.509 resource
3. A string having the format `file://path/to/file.pem` - the named file must contain a PEM encoded certificate/private key (it may contain both)
4. A string containing the content of a certificate/key, PEM encoded
5. For private keys, you may also use the syntax `array($key, $passphrase)` where `$key` represents a key specified using the `file://` or textual content notation above, and `$passphrase` represents a string containing the passphrase for that private key

## Certificate Verification

When calling a function that will verify a signature/certificate, the *cainfo* parameter is an array containing file and directory names that specify the locations of trusted CA files. If a directory is specified, then it must be a correctly formed hashed directory as the **openssl** command would use.

## PKCS7 Flags/Constants

The S/MIME functions make use of flags which are specified using a bitfield which can include one or more of the following values:

**Tabla 1. PKCS7 CONSTANTS**

Constant	Description
PKCS7_TEXT	adds text/plain content type headers to encrypted/signed message. If decrypting or verifying, it strips those headers from the output - if the decrypted or verified message is not of MIME type text/plain then an error will occur.

PKCS7_BINARY	normally the input message is converted to "canonical" format which is effectively using CR and LF as end of line: as required by the S/MIME specification. When this options is present, no translation occurs. This is useful when handling binary data which may not be in MIME format.
PKCS7_NOINTERN	when verifying a message, certificates (if any) included in the message are normally searched for the signing certificate. With this option only the certificates specified in the <i>extracerts</i> parameter of <a href="#">openssl_pkcs7_verify()</a> are used. The supplied certificates can still be used as untrusted CAs however.
PKCS7_NOVERIFY	do not verify the signers certificate of a signed message.
PKCS7_NOCHAIN	do not chain verification of signers certificates: that is don't use the certificates in the signed message as untrusted CAs.
PKCS7_NOCERTS	when signing a message the signer's certificate is normally included - with this option it is excluded. This will reduce the size of the signed message but the verifier must have a copy of the signers certificate available locally (passed using the <i>extracerts</i> to <a href="#">openssl_pkcs7_verify()</a> for example.
PKCS7_NOATTR	normally when a message is signed, a set of attributes are included which include the signing time and the supported symmetric algorithms. With this option they are not included.
PKCS7_DETACHED	When signing a message, use cleartext signing with the MIME type multipart/signed. This is the default if the <i>flags</i> parameter to <a href="#">openssl_pkcs7_sign()</a> if you do not specify any flags. If you turn this option off, the message will be signed using opaque signing, which is more resistant to translation by mail relays but cannot be read by mail agents that do not support S/MIME.
PKCS7_NOSIGS	Don't try and verify the signatures on a message

**Nota:** These constants were added in 4.0.6.

### Tabla de contenidos

[openssl\\_error\\_string](#) -- Return openssl error message

[openssl\\_free\\_key](#) -- Free key resource

[openssl\\_get\\_privatekey](#) -- Prepare a PEM formatted private key for use

[openssl\\_get\\_publickey](#) -- Extract public key from certificate and prepare it for use

[openssl\\_open](#) -- Open sealed data

[openssl\\_seal](#) -- Seal (encrypt) data  
[openssl\\_sign](#) -- Generate signature  
[openssl\\_verify](#) -- Verify signature  
[openssl\\_pkcs7\\_decrypt](#) -- Decrypts an S/MIME encrypted message  
[openssl\\_pkcs7\\_encrypt](#) -- Encrypt an S/MIME message  
[openssl\\_pkcs7\\_sign](#) -- sign an S/MIME message  
[openssl\\_pkcs7\\_verify](#) -- Verifies the signature of an S/MIME signed message  
[openssl\\_x509\\_checkpurpose](#) -- Verifies if a certificate can be used for a particular purpose  
[openssl\\_x509\\_free](#) -- Free certificate resource  
[openssl\\_x509\\_parse](#) -- Parse an X509 certificate and return the information as an array  
[openssl\\_x509\\_read](#) -- Parse an X.509 certificate and return a resource identifier for it

---

[Anterior](#)

OCInternalDebug

[Inicio](#)

[Subir](#)

[Siguiete](#)

openssl\_error\_string

# LX. Funciones Oracle

## Tabla de contenidos

[Ora\\_Bind](#) -- Vincula una variable PHP a un parámetro Oracle

[Ora\\_Close](#) -- Cierra un cursor Oracle

[Ora\\_ColumnName](#) -- toma el nombre de una columna de resultados Oracle

[Ora\\_ColumnType](#) -- toma el tipo de una columna de resultados Oracle

[Ora\\_Commit](#) -- realiza una transacción Oracle

[Ora\\_CommitOff](#) -- deshabilita el modo automatico de ejecucion de tareas (autocommit)

[Ora\\_CommitOn](#) -- Habilita la ejecucion automática de tareas (autocommit)

[Ora\\_Error](#) -- toma los mensajes de error de Oracle

[Ora\\_ErrorCode](#) -- captura el código de error Oracle

[Ora\\_Exec](#) -- ejecuta las declaraciones interpretadas en un cursor Oracle

[Ora\\_Fetch](#) -- extrae una fila de datos a partir de un cursor

[Ora\\_GetColumn](#) -- toma datos de la fila extraída

[Ora\\_Logoff](#) -- cierra una conexión Oracle

[Ora\\_Logon](#) -- Abre una conexión Oracle

[Ora\\_Open](#) -- abre un cursor Oracle

[Ora\\_Parse](#) -- interpreta una declaración SQL

[Ora\\_Rollback](#) -- retrocede en la lista de transacciones (hace un roll back)



# LXI. Ovrimos SQL functions

Ovrimos SQL Server, is a client/server, transactional RDBMS combined with Web capabilities and fast transactions.

Ovrimos SQL Server is available at [www.ovrimos.com](http://www.ovrimos.com). To enable ovrimos support in PHP just compile php with the '--with-ovrimos' parameter to configure script. You'll need to install the sqlcli library available in the Ovrimos SQL Server distribution.

## Ejemplo 1. Connect to Ovrimos SQL Server and select from a system table

```
<?php
$conn = ovrimos_connect ("server.domain.com", "8001", "admin", "password");
if ($conn != 0) {
    echo ("Connection ok!");
    $res = ovrimos_exec ($conn, "select table_id, table_name from sys.tables");
    if ($res != 0) {
        echo "Statement ok!";
        ovrimos_result_all ($res);
        ovrimos_free_result ($res);
    }
    ovrimos_close($conn);
}
?>
```

This will just connect to SQL Server.

### Tabla de contenidos

[ovrimos\\_connect](#) -- Connect to the specified database

[ovrimos\\_close](#) -- Closes the connection to ovrimos

[ovrimos\\_close\\_all](#) -- Closes all the connections to ovrimos

[ovrimos\\_longreadlen](#) -- Specifies how many bytes are to be retrieved from long datatypes

[ovrimos\\_prepare](#) -- Prepares an SQL statement

[ovrimos\\_execute](#) -- Executes a prepared SQL statement

[ovrimos\\_cursor](#) -- Returns the name of the cursor

[ovrimos\\_exec](#) -- Executes an SQL statement

[ovrimos\\_fetch\\_into](#) -- Fetches a row from the result set

[ovrimos\\_fetch\\_row](#) -- Fetches a row from the result set

[ovrimos\\_result](#) -- Retrieves the output column

[ovrimos\\_result\\_all](#) -- Prints the whole result set as an HTML table

[ovrimos\\_num\\_rows](#) -- Returns the number of rows affected by update operations

[ovrimos\\_num\\_fields](#) -- Returns the number of columns

[ovrimos\\_field\\_name](#) -- Returns the output column name

[ovrimos\\_field\\_type](#) -- Returns the (numeric) type of the output column

[ovrimos\\_field\\_len](#) -- Returns the length of the output column  
[ovrimos\\_field\\_num](#) -- Returns the (1-based) index of the output column  
[ovrimos\\_free\\_result](#) -- Frees the specified result\_id  
[ovrimos\\_commit](#) -- Commits the transaction  
[ovrimos\\_rollback](#) -- Rolls back the transaction

---

[Anterior](#)

Ora\_Rollback

[Inicio](#)

[Subir](#)

[Siguiete](#)

ovrimos\_connect

## LXII. Output Control Functions

The Output Control functions allow you to control when output is sent from the script. This can be useful in several different situations, especially if you need to send headers to the browser after your script has began outputting data. The Output Control functions do not affect headers sent using [header\(\)](#) or [setcookie\(\)](#), only functions such as [echo\(\)](#) and data between blocks of PHP code.

### Ejemplo 1. Output Control example

```
<?php
ob_start();
echo "Hello\n";

setcookie ("cookiename", "cookiedata");

ob_end_flush();

?>
```

In the above example, the output from [echo\(\)](#) would be stored in the output buffer until [ob\\_end\\_flush\(\)](#) was called. In the mean time, the call to [setcookie\(\)](#) successfully stored a cookie without causing an error. (You can not normally send headers to the browser after data has already been sent.)

See also [header\(\)](#) and [setcookie\(\)](#).

### Tabla de contenidos

[flush](#) -- Flush the output buffer

[ob\\_start](#) -- Turn on output buffering

[ob\\_get\\_contents](#) -- Return the contents of the output buffer

[ob\\_get\\_length](#) -- Return the length of the output buffer

[ob\\_end\\_flush](#) -- Flush (send) the output buffer and turn off output buffering

[ob\\_end\\_clean](#) -- Clean (erase) the output buffer and turn off output buffering

[ob\\_implicit\\_flush](#) -- Turn implicit flush on/off

---

[Anterior](#)

ovrimos\_rollback

[Inicio](#)

[Subir](#)

[Siguiente](#)

flush

# LXIII. Object property and method call overloading

## Aviso

This module is *EXPERIMENTAL*. That means, that the behaviour of these functions, these function names, in concreto ANYTHING documented here can change in a future release of PHP WITHOUT NOTICE. Be warned, and use this module at your own risk.

The purpose of this extension is to allow overloading of object property access and method calls. Only one function is defined in this extension, [overload\(\)](#) which takes the name of the class that should have this functionality enabled. The class named has to define appropriate methods if it wants to have this functionality: `__get()`, `__set()` and `__call()` respectively for getting/setting a property, or calling a method. This way overloading can be selective. Inside these handler functions the overloading is disabled so you can access object properties normally.

Some simple examples on using the [overload\(\)](#) function:

### Ejemplo 1. Overloading a PHP class

```
<?php
class OO
{
    var $a = 111;
    var $elem = array('b' => 9, 'c' => 42);

    // Callback method for getting a property
    function __get($prop_name, &$prop_value)
    {
        if (isset($this->elem[$prop_name])) {
            $prop_value = $this->elem[$prop_name];
            return true;
        }
    }
}
```

```

        } else {
            return false;
        }
    }

    // Callback method for setting a property
    function __set($prop_name, $prop_value)
    {
        $this->elem[$prop_name] = $prop_value;
        return true;
    }
}

// Here we overload the OO object
overload('OO');

$o = new OO;
print "\$o->a: $o->a\n"; // print: $o->a:
print "\$o->b: $o->b\n"; // print: $o->b: 9
print "\$o->c: $o->c\n"; // print: $o->c: 42
print "\$o->d: $o->d\n"; // print: $o->d:

// add a new item to the $elem array in OO
$o->x = 56;

// instantiate stdClass (it is built-in in PHP 4)
// $val is not overloaded!
$val = new stdClass;
$val->prop = 555;

// Set "a" to be an array with the $val object in it
// But __set() will put this in the $elem array
$o->a = array($val);
var_dump($o->a[0]->prop);

?>

```

As this is an experimental extension, not all things work. There is no `__call()` support currently, you can only overload the get and set operations for properties. You cannot invoke the original overloading handlers of the class, and `__set()` only works to one level of property access.

## Tabla de contenidos

[overload](#) -- Enable property and method call overloading for a class

---

[Anterior](#)

ob\_implicit\_flush

[Inicio](#)

[Subir](#)

[Siguiete](#)

overload

## LXIV. PDF functions

You can use the PDF functions in PHP to create PDF files if you have the PDF library by Thomas Merz (available at <http://www.pdflib.com/pdflib/index.html>; you will also need [the JPEG library](#) and [the TIFF library](#) to compile this. These two libs also quite often make problems when configuring php. Follow the messages of configure to fix possible problems. If you use pdflib 2.01 check how the lib was installed. There should be file or link libpdf.so. Version 2.01 just creates a lib with the name libpdf2.01.so which cannot be found when linking the test program in configure. You will have to create a symbolic link from libpdf.so to libpdf2.01.so.).

Version 2.20 of pdflib has introduced more changes to its API and support for chinese and japanese fonts. This unfortunately causes some changes of the pdf module of php4 (not php3). If you use pdflib 2.20 handle the in memory generation of PDF documents with care. Until pdflib 3.0 is released it might be unstable. The encoding parameter of [pdf\\_set\\_font\(\)](#) has changed to a string. This means that instead of e.g. 4 you have to use 'winansi'.

If you use pdflib 2.30 the [pdf\\_set\\_text\\_matrix\(\)](#) will have gone. It is not supported any more. In general it is a good advise to consult the release notes of the used version of pdflib for possible changes.

Since version 3.0 of pdflib you should configure pdflib with the option `--enable-shared-pdflib`.

Any version of PHP4 after March, 9th 2000 do not support versions of pdflib older than 3.0. PHP3 on the other hand should not be used with version newer than 2.01.

Please consult the excellent documentation for pdflib shipped with the source distribution of pdflib. It provides a very good overview of what pdflib capable of doing. Most of the functions in pdflib and the PHP module have the same name. The parameters are also identical. You should also understand some of the concepts of PDF or Postscript to efficiently use this module. All lengths and coordinates are measured in Postscript points. There are generally 72 PostScript points to an inch, but this depends on the output resolution.

There is another PHP module for pdf document creation based on [FastIO's](#). ClibPDF. It has a slightly different API. Check the [ClibPDF functions](#) section for details.

Currently all versions of pdflib are supported. It is recommended that you use the newest version since it has more features and fixes some problems which required a patch for the old version. Unfortunately, the changes of the pdflib API in 2.x compared to 0.6 have been so severe that even some PHP functions had to be altered. Here is a list of changes:

- The Info structure does not exist anymore. Therefore the function [pdf\\_get\\_info\(\)](#) is obsolete and the functions [pdf\\_set\\_info\\_creator\(\)](#), [pdf\\_set\\_info\\_title\(\)](#), [pdf\\_set\\_info\\_author\(\)](#), [pdf\\_set\\_info\\_subject\(\)](#) and [pdf\\_set\\_info\\_keywords\(\)](#) do not take the info structure as the first parameter but the pdf document. This also means that the pdf document must be opened before these functions can be called. The above functions can and should also be replaced by [pdf\\_set\\_info\(\)](#)
- The way a new document is opened has changed. The function [pdf\\_open\(\)](#) takes only one parameter which



is the file handle of a file opened with [fopen\(\)](#).

There were some more changes with the release 2.01 of pdflib which should be covered by PHP. Some functions are not required anymore (e.g. [pdf\\_put\\_image\(\)](#)). You will get a warning so don't be shocked.

The pdf module introduces two new types of variables (if pdflib 2.x is used it is only one new type). They are called *pdfdoc* and *pdfinfo* (*pdfinfo* is not existent if pdflib 2.x is used. *pdfdoc* is a pointer to a pdf document and almost all functions need it as its first parameter. *pdfinfo* contains meta data about the PDF document. It has to be set before [pdf\\_open\(\)](#) is called.

**Nota:** The following is only **TRUE** for pdflib 0.6. Read the pdflib manual for newer version

In order to output text into a PDF document you will need to provide the afm file for each font. Afm files contain font metrics for a Postscript font. By default these afm files are searched for in a directory named 'fonts' relative to the directory where the PHP script is located. (Again, this was **TRUE** for pdflib 0.6, newer versions do not necessarily need the afm files.)

Most of the functions are fairly easy to use. The most difficult part is probably to create a very simple pdf document at all. The following example should help to get started. It uses the PHP functions for pdflib 0.6. It creates the file test.pdf with one page. The page contains the text "Times-Roman" in an outlined 30pt font. The text is also underlined.

### Ejemplo 1. Creating a PDF document with pdflib 0.6

```
<?php
$fp = fopen("test.pdf", "w");
$info = PDF_get_info();
pdf_set_info_author($info, "Uwe Steinmann");
PDF_set_info_title($info, "Test for PHP wrapper of PDFlib 0.6");
PDF_set_info_author($info, "Name of Author");
pdf_set_info_creator($info, "See Author");
pdf_set_info_subject($info, "Testing");
$pdf = PDF_open($fp, $info);
PDF_begin_page($pdf, 595, 842);
PDF_add_outline($pdf, "Page 1");
pdf_set_font($pdf, "Times-Roman", 30, 4);
pdf_set_text_rendering($pdf, 1);
PDF_show_xy($pdf, "Times Roman outlined", 50, 750);
pdf_moveto($pdf, 50, 740);
pdf_lineto($pdf, 330, 740);
pdf_stroke($pdf);
PDF_end_page($pdf);
PDF_close($pdf);
fclose($fp);
echo "<A HREF=getpdf.php3>finished</A>";
?>
```

The PHP script `getpdf.php3` just outputs the pdf document.

```
<?php
$fp = fopen("test.pdf", "r");
header("Content-type: application/pdf");
fpassthru($fp);
fclose($fp);
?>
```

Doing the same with `pdflib 2.x` looks like the following:

### **Ejemplo 2. Creating a PDF document with `pdflib 2.x`**

```
<?php
$fp = fopen("test.pdf", "w");
$pdf = PDF_open($fp);
pdf_set_info_author($pdf, "Uwe Steinmann");
PDF_set_info_title($pdf, "Test for PHP wrapper of PDFlib 2.0");
PDF_set_info_author($pdf, "Name of Author");
pdf_set_info_creator($pdf, "See Author");
pdf_set_info_subject($pdf, "Testing");
PDF_begin_page($pdf, 595, 842);
PDF_add_outline($pdf, "Page 1");
pdf_set_font($pdf, "Times-Roman", 30, 4);
pdf_set_text_rendering($pdf, 1);
PDF_show_xy($pdf, "Times Roman outlined", 50, 750);
pdf_moveto($pdf, 50, 740);
pdf_lineto($pdf, 330, 740);
pdf_stroke($pdf);
PDF_end_page($pdf);
PDF_close($pdf);
fclose($fp);
echo "<A HREF=getpdf.php3>finished</A>";
?>
```

The PHP script `getpdf.php3` is the same as above.

The `pdflib` distribution contains a more complex example which creates a series of pages with an analog clock. This example converted into PHP using `pdflib 2.x` looks as the following (you can see the same example in the documentation for the [`clibpdf` module](#)):

### Ejemplo 3. pdfclock example from pdflib 2.x distribution

```
<?php
$pdffilename = "clock.pdf";
$radius = 200;
$margin = 20;
$pagecount = 40;

$fp = fopen($pdffilename, "w");
$pdf = pdf_open($fp);
pdf_set_info_creator($pdf, "pdf_clock.php3");
pdf_set_info_author($pdf, "Uwe Steinmann");
pdf_set_info_title($pdf, "Analog Clock");

while($pagecount-- > 0) {
    pdf_begin_page($pdf, 2 * ($radius + $margin), 2 * ($radius + $margin));

    pdf_set_transition($pdf, 4); /* wipe */
    pdf_set_duration($pdf, 0.5);

    pdf_translate($pdf, $radius + $margin, $radius + $margin);
    pdf_save($pdf);
    pdf_setrgbcolor($pdf, 0.0, 0.0, 1.0);

    /* minute strokes */
    pdf_setlinewidth($pdf, 2.0);
    for ($alpha = 0; $alpha < 360; $alpha += 6) {
        pdf_rotate($pdf, 6.0);
        pdf_moveto($pdf, $radius, 0.0);
        pdf_lineto($pdf, $radius-$margin/3, 0.0);
        pdf_stroke($pdf);
    }

    pdf_restore($pdf);
    pdf_save($pdf);

    /* 5 minute strokes */
    pdf_setlinewidth($pdf, 3.0);
    for ($alpha = 0; $alpha < 360; $alpha += 30) {
        pdf_rotate($pdf, 30.0);
        pdf_moveto($pdf, $radius, 0.0);
        pdf_lineto($pdf, $radius-$margin, 0.0);
        pdf_stroke($pdf);
    }

    $ltime = getdate();

    /* draw hour hand */
    pdf_save($pdf);
    pdf_rotate($pdf, -((($ltime['minutes']/60.0)+$ltime['hours']-3.0)*30.0));
    pdf_moveto($pdf, -$radius/10, -$radius/20);
```

```

pdf_lineto($pdf, $radius/2, 0.0);
pdf_lineto($pdf, -$radius/10, $radius/20);
pdf_closepath($pdf);
pdf_fill($pdf);
pdf_restore($pdf);

/* draw minute hand */
pdf_save($pdf);
pdf_rotate($pdf, -(($ltime['seconds']/60.0)+$ltime['minutes']-15.0)*6.0);
pdf_moveto($pdf, -$radius/10, -$radius/20);
pdf_lineto($pdf, $radius * 0.8, 0.0);
pdf_lineto($pdf, -$radius/10, $radius/20);
pdf_closepath($pdf);
pdf_fill($pdf);
pdf_restore($pdf);

/* draw second hand */
pdf_setrgbcolor($pdf, 1.0, 0.0, 0.0);
pdf_setlinewidth($pdf, 2);
pdf_save($pdf);
pdf_rotate($pdf, -(($ltime['seconds'] - 15.0) * 6.0));
pdf_moveto($pdf, -$radius/5, 0.0);
pdf_lineto($pdf, $radius, 0.0);
pdf_stroke($pdf);
pdf_restore($pdf);

/* draw little circle at center */
pdf_circle($pdf, 0, 0, $radius/30);
pdf_fill($pdf);

pdf_restore($pdf);

pdf_end_page($pdf);
}

$pdf = pdf_close($pdf);
fclose($fp);
echo "<A HREF=getpdf.php3?filename=".$pdffilename.">finished</A>";
?>

```

The PHP script `getpdf.php3` just outputs the pdf document.

```
<?php
$fp = fopen($filename, "r");
header("Content-type: application/pdf");
fpassthru($fp);
fclose($fp);
?>
```

## Tabla de contenidos

[PDF\\_get\\_info](#) -- Returns an empty info structure for a pdf document

[PDF\\_set\\_info](#) -- Fills a field of the document information

[PDF\\_open](#) -- Opens a new pdf document

[PDF\\_close](#) -- Closes a pdf document

[PDF\\_begin\\_page](#) -- Starts new page

[PDF\\_end\\_page](#) -- Ends a page

[PDF\\_show](#) -- Output text at current position

[PDF\\_show\\_boxed](#) -- Output text in a box

[PDF\\_show\\_xy](#) -- Output text at given position

[PDF\\_set\\_font](#) -- Selects a font face and size

[PDF\\_set\\_leading](#) -- Sets distance between text lines

[PDF\\_set\\_parameter](#) -- Sets certain parameters

[PDF\\_get\\_parameter](#) -- Gets certain parameters

[PDF\\_set\\_value](#) -- Sets certain numerical value

[PDF\\_get\\_value](#) -- Gets certain numerical value

[PDF\\_set\\_text\\_rendering](#) -- Determines how text is rendered

[PDF\\_set\\_horiz\\_scaling](#) -- Sets horizontal scaling of text

[PDF\\_set\\_text\\_rise](#) -- Sets the text rise

[PDF\\_set\\_text\\_matrix](#) -- Sets the text matrix

[PDF\\_set\\_text\\_pos](#) -- Sets text position

[PDF\\_set\\_char\\_spacing](#) -- Sets character spacing

[PDF\\_set\\_word\\_spacing](#) -- Sets spacing between words

[PDF\\_skew](#) -- Skews the coordinate system

[PDF\\_continue\\_text](#) -- Outputs text in next line

[PDF\\_stringwidth](#) -- Returns width of text using current font

[PDF\\_save](#) -- Saves the current environment

[PDF\\_restore](#) -- Restores formerly saved environment

[PDF\\_translate](#) -- Sets origin of coordinate system

[PDF\\_scale](#) -- Sets scaling

[PDF\\_rotate](#) -- Sets rotation

[PDF\\_setflat](#) -- Sets flatness

[PDF\\_setlinejoin](#) -- Sets linejoin parameter

[PDF\\_setlinecap](#) -- Sets linecap parameter

[PDF\\_setmiterlimit](#) -- Sets miter limit

[PDF\\_setlinewidth](#) -- Sets line width  
[PDF\\_setdash](#) -- Sets dash pattern  
[PDF\\_moveto](#) -- Sets current point  
[PDF\\_curveto](#) -- Draws a curve  
[PDF\\_lineto](#) -- Draws a line  
[PDF\\_circle](#) -- Draws a circle  
[PDF\\_arc](#) -- Draws an arc  
[PDF\\_rect](#) -- Draws a rectangle  
[PDF\\_closepath](#) -- Closes path  
[PDF\\_stroke](#) -- Draws line along path  
[PDF\\_closepath\\_stroke](#) -- Closes path and draws line along path  
[PDF\\_fill](#) -- Fills current path  
[PDF\\_fill\\_stroke](#) -- Fills and strokes current path  
[PDF\\_closepath\\_fill\\_stroke](#) -- Closes, fills and strokes current path  
[PDF\\_endpath](#) -- Ends current path  
[PDF\\_clip](#) -- Clips to current path  
[PDF\\_setgray\\_fill](#) -- Sets filling color to gray value  
[PDF\\_setgray\\_stroke](#) -- Sets drawing color to gray value  
[PDF\\_setgray](#) -- Sets drawing and filling color to gray value  
[PDF\\_setrgbcolor\\_fill](#) -- Sets filling color to rgb color value  
[PDF\\_setrgbcolor\\_stroke](#) -- Sets drawing color to rgb color value  
[PDF\\_setrgbcolor](#) -- Sets drawing and filling color to rgb color value  
[PDF\\_add\\_outline](#) -- Adds bookmark for current page  
[PDF\\_set\\_transition](#) -- Sets transition between pages  
[PDF\\_set\\_duration](#) -- Sets duration between pages  
[PDF\\_open\\_gif](#) -- Opens a GIF image  
[PDF\\_open\\_png](#) -- Opens a PNG image  
[PDF\\_open\\_memory\\_image](#) -- Opens an image created with PHP's image functions  
[PDF\\_open\\_jpeg](#) -- Opens a JPEG image  
[PDF\\_close\\_image](#) -- Closes an image  
[PDF\\_place\\_image](#) -- Places an image on the page  
[PDF\\_put\\_image](#) -- Stores an image in the PDF for later use  
[PDF\\_execute\\_image](#) -- Places a stored image on the page  
[pdf\\_add\\_annotation](#) -- Adds annotation  
[PDF\\_set\\_border\\_style](#) -- Sets style of border around links and annotations  
[PDF\\_set\\_border\\_color](#) -- Sets color of border around links and annotations  
[PDF\\_set\\_border\\_dash](#) -- Sets dash style of border around links and annotations

# LXV. Verisign Payflow Pro functions

This extension allows you to process credit cards and other financial transactions using Verisign Payment Services, formerly known as Signio (<http://www.verisign.com/payment/>).

These functions are only available if PHP has been compiled with the `--with-pfpro[=DIR]` option. You will require the appropriate SDK for your platform, which may be downloaded [from within the manager interface](#) once you have registered.

Once you have downloaded the SDK you should copy the files from the `lib` directory of the distribution. Copy the header file `pfpro.h` to `/usr/local/include` and the library file `libpfpro.so` to `/usr/local/lib`.

When using these functions, you may omit calls to [pfpro\\_init\(\)](#) and [pfpro\\_cleanup\(\)](#) as this extension will do so automatically if required. However the functions are still available in case you are processing a number of transactions and require fine control over the library. You may perform any number of transactions using [pfpro\\_process\(\)](#) between the two.

These functions have been added in PHP 4.0.2.

**Nota:** These functions only provide a link to Verisign Payment Services. Be sure to read the Payflow Pro Developers Guide for full details of the required parameters.

## Tabla de contenidos

[pfpro\\_init](#) -- Initialises the Payflow Pro library

[pfpro\\_cleanup](#) -- Shuts down the Payflow Pro library

[pfpro\\_process](#) -- Process a transaction with Payflow Pro

[pfpro\\_process\\_raw](#) -- Process a raw transaction with Payflow Pro

[pfpro\\_version](#) -- Returns the version of the Payflow Pro software

PDF\_set\_border\_dash

[Subir](#)

pfpro\_init



# LXVI. opciones e información de PHP

## Tabla de contenidos

[extension\\_loaded](#) -- averigua si una extensión ha sido cargada

[getenv](#) -- Obtiene el valor de una variable de entorno

[get\\_cfg\\_var](#) -- Obtiene el valor de una opción de configuración de PHP.

[get\\_current\\_user](#) -- Obtiene el nombre del propietario del script PHP actual.

[get\\_magic\\_quotes\\_gpc](#) -- Obtiene el valor de la configuración activa actual de las comillas mágicas gpc.

[get\\_magic\\_quotes\\_runtime](#) -- Obtiene el valor de la configuración activa actual de magic\_quotes\_runtime.

[getlastmod](#) -- Recupera la fecha/hora de la última modificación de la página.

[getmyinode](#) -- Recupera el inodo del script actual.

[getmypid](#) -- Obtiene el ID de proceso de PHP.

[getmyuid](#) -- Obtiene el UID del propietario del script PHP.

[getrusage](#) -- Obtiene el consumo actual de recursos.

[phpinfo](#) -- Recupera gran cantidad de información de PHP.

[phpversion](#) -- Obtiene la versión actual de PHP.

[php\\_logo\\_guid](#) -- Obtiene el guid logo

[putenv](#) -- Establece el valor de una variable de entorno.

[set\\_magic\\_quotes\\_runtime](#) -- Establece el valor de la configuración activa actual de magic\_quotes\_runtime.

[set\\_time\\_limit](#) -- limita el tiempo máximo de ejecución

[zend\\_logo\\_guid](#) -- Obtiene el guid zend

# LXVII. Funciones POSIX

Este módulo contiene una interfaz a aquellas funciones definidas en el documento estandar IEEE 1003.1 (POSIX.1) que no son accesibles de otra manera. POSIX.1 por ejemplo definió las funciones `open()`, `read()`, `write()` y `close()`, las cuales han sido parte de PHP durante mucho tiempo. Algunas funciones específicas del sistema no habian estado disponibles antes, aunque con este módulo se intenta remediar esto ofreciendo un acceso fácil a esas funciones.

## Tabla de contenidos

[posix\\_kill](#) -- Manda una señal a un proceso

[posix\\_getpid](#) -- Devuelve el identificador del proceso actual

[posix\\_getppid](#) -- Devuelve el identificador del proceso padre

[posix\\_getuid](#) -- Devuelve el ID de usuario real del proceso actual

[posix\\_geteuid](#) -- Devuelve el ID de usuario efectivo del proceso actual

[posix\\_getgid](#) -- Devuelve el ID de grupo real del proceso actual

[posix\\_getegid](#) -- Devuelve el ID de grupo efectivo del proceso actual

[posix\\_setuid](#) -- Asigna el UID efectivo del proceso actual

[posix\\_setgid](#) -- Asigna el GID efectivo del proceso actual

[posix\\_getgroups](#) -- Devuelve el conjunto de grupos del proceso actual

[posix\\_getlogin](#) -- Devuelve el nombre de usuario

[posix\\_getpgrp](#) -- Devuelve el identificador de grupo del proceso actual

[posix\\_setsid](#) -- Convierte el proceso actual en lider de sesión

[posix\\_setpgid](#) -- Asigna el id de grupo de procesos para el control de trabajos

[posix\\_getpgid](#) -- Recoge el id del grupo de procesos para el control de trabajo

[posix\\_getsid](#) -- Consigue el sid actual del proceso

[posix\\_uname](#) -- Consigue el nombre del sistema

[posix\\_times](#) -- Recoge el tiempo de los procesos

[posix\\_ctermid](#) -- Recoge el nombre de ruta de la terminal de control

[posix\\_ttyname](#) -- Determina el nombre del dispositivo terminal

[posix\\_isatty](#) -- Determina si un descriptor de fichero esta en una terminal interactiva

[posix\\_getcwd](#) -- Nombre de ruta del directorio actual

[posix\\_mkfifo](#) -- Crea un fichero especial fifo (los llamados pipe o tuberias)

[posix\\_getgrnam](#) -- Devuelve información sobre un grupo a través del nombre  
[posix\\_getgrgid](#) -- Devuelve información sobre un grupo a través del id de grupo  
[posix\\_getpwnam](#) -- Devuelve información sobre un usuario a través del nombre de usuario  
[posix\\_getpwuid](#) -- Devuelve información sobre un usuario a través de su id  
[posix\\_getrlimit](#) -- Devuelve información sobre los límites de recursos del sistema

---

[Anterior](#)

zend\_logo\_guid

[Inicio](#)

[Subir](#)

[Siguiete](#)

posix\_kill

## LXVIII. Funciones de PostgreSQL

Postgres, desarrollado originalmente en el UC Berkeley Computer Science Department, ha sido pionero en muchos de los conceptos relacionales/orientados a objeto que ahora están empezando a estar disponibles en algunas bases de datos comerciales. Tiene soporte de lenguaje SQL92/SQL3, integridad transaccional, y extensibilidad de tipos. PostgreSQL es un descendiente de dominio público, más concretamente open source, del código original de Berkeley.

PostgreSQL se encuentra disponible sin coste alguno. La versión actual la tienes a tu disposición en [www.PostgreSQL.org](http://www.PostgreSQL.org).

Desde la versión 6.3 (02/03/1998) PostgreSQL usa sockets tipo Unix. Abajo se da una tabla con las diferentes posibilidades. El socket se encuentra en el fichero `/tmp/.s.PGSQL.5432`. Esta opción se controla mediante el flag `'-i'` del **postmaster** y cuando se incluye significa "escuchar sockets TCP/IP además de los de dominio Unix" ya que si no se le dice nada solo escucha sockets tipo Unix.

**Tabla 1. Postmaster y PHP**

Postmaster	PHP	Estado
postmaster &	<code>pg_connect("", "", "", "", "dbname");</code>	OK
postmaster -i &	<code>pg_connect("", "", "", "", "dbname");</code>	OK

postmaster &	pg_connect("localhost", "", "", "", "dbname");	Unable to connect to PostgreSQL server: connectDB() failed: Is the postmaster running and accepting TCP/IP (with -i) connection at 'localhost' on port '5432'? in /path/to/file.php3 on line 20. (Imposible conectar al servidor PostgreSQL, la llamada connectDB() ha fallado: ¿Está funcionando el postmaster aceptando conexiones TCP/IP (con -i) en 'localhost' en el puerto '5432'? en /path/to/file.php3 en linea 20.
postmaster -i &	pg_connect("localhost", "", "", "", "dbname");	OK

Uno puede establecer una conexión con el siguiente comando:

Para usar el interface de objetos grandes (large object o lo), es necesario encapsularlo en un bloque de transacción. Un bloque de transacción empieza con un **begin** y si la transacción fue valida termina con **commit** y **end**. Si la transacción falla debe ser cerrada con **abort** y **rollback**.

### Ejemplo 1. Usando Objetos Grandes (lo)

```

<?php
$database = pg_Connect ("", "", "", "", "jacarta");
pg_exec ($database, "begin");
    $oid = pg_locreate ($database);
    echo ("$oid\n");
    $handle = pg_loopen ($database, $oid, "w");
    echo ("$handle\n");
    pg_lowrite ($handle, "gaga");
    pg_loclose ($handle);
pg_exec ($database, "commit")
pg_exec ($database, "end")
?>

```

## Tabla de contenidos

[pg\\_Close](#) -- Cierra una conexión PostgreSQL

[pg\\_cmdTuples](#) -- Devuelve el número de tuplas afectadas

[pg\\_Connect](#) -- Abre una conexión

[pg\\_DBname](#) -- Nombre de la base de datos

[pg\\_ErrorMessage](#) -- mensaje de error

[pg\\_Exec](#) -- Ejecuta una consulta (query)

[pg\\_Fetch\\_Array](#) -- obtiene una fila en la forma de un array

[pg\\_Fetch\\_Object](#) -- obtener una fila en forma de objeto

[pg\\_Fetch\\_Row](#) -- obtiene la fila como un array enumerado

[pg\\_FieldIsNull](#) -- Comprueba si un campo es NULO

[pg\\_FieldName](#) -- Devuelve el nombre de un campo

[pg\\_FieldNum](#) -- Devuelve el número de una columna

[pg\\_FieldPrtLen](#) -- Devuelve la longitud impresa

[pg\\_FieldSize](#) -- Devuelve el tamaño de almacenamiento interno de un campo en concreto

[pg\\_FieldType](#) -- Devuelve el nombre del tipo de dato correspondiente al campo cuyo número pasamos como parámetro

[pg\\_FreeResult](#) -- Libera memoria

[pg\\_GetLastOid](#) -- Devuelve el identificador del último objeto insertado

[pg\\_Host](#) -- Devuelve el nombre del host

[pg\\_loclose](#) -- Cierra un objeto grande (large object)

[pg\\_locreate](#) -- Crea un objeto grande

[pg\\_loopen](#) -- Abre un objeto grande

[pg\\_loread](#) -- lee un large object (objeto grande)  
[pg\\_loreadall](#) -- Lee un objeto grande entero  
[pg\\_lounlink](#) -- borra un large object  
[pg\\_lowrite](#) -- escribe en un objeto grande  
[pg\\_NumFields](#) -- Devuelve el número de campos  
[pg\\_NumRows](#) -- Devuelve el número de filas  
[pg\\_Options](#) -- Devuelve opciones  
[pg\\_pConnect](#) -- Crea una conexión persistente con una base de datos  
[pg\\_Port](#) -- Devuelve el número de puerto  
[pg\\_Result](#) -- Devuelve valores a partir de un identificador de resultado  
[pg\\_tty](#) -- Devuelve el nombre del tty

---

[Anterior](#)

posix\_getrlimit

[Inicio](#)

[Subir](#)

[Siguiete](#)

pg\_Close

# LXIX. Funciones de ejecución de programas

## Tabla de contenidos

[escapeshellcmd](#) -- enmascara los metacaracteres del intérprete de ordenes

[exec](#) -- Ejecuta un programa externo

[passthru](#) -- Ejecuta un programa externo y muestra su salida literal

[system](#) -- Ejecuta un programa externo y muestra su salida



# LXX. Printer functions

These functions are only available under Windows 9.x, ME, NT4 and 2000. They have been added in PHP 4 (4.0.4).

## Tabla de contenidos

- [printer\\_open](#) -- Open connection to a printer
- [printer\\_abort](#) -- Deletes the printer's spool file
- [printer\\_close](#) -- Close an open printer connection
- [printer\\_write](#) -- Write data to the printer
- [printer\\_list](#) -- Return an array of printers attached to the server
- [printer\\_set\\_option](#) -- Configure the printer connection
- [printer\\_get\\_option](#) -- Retrieve printer configuration data
- [printer\\_create\\_dc](#) -- Create a new device context
- [printer\\_delete\\_dc](#) -- Delete a device context
- [printer\\_start\\_doc](#) -- Start a new document
- [printer\\_end\\_doc](#) -- Close document
- [printer\\_start\\_page](#) -- Start a new page
- [printer\\_end\\_page](#) -- Close active page
- [printer\\_create\\_pen](#) -- Create a new pen
- [printer\\_delete\\_pen](#) -- Delete a pen
- [printer\\_select\\_pen](#) -- Select a pen
- [printer\\_create\\_brush](#) -- Create a new brush
- [printer\\_delete\\_brush](#) -- Delete a brush
- [printer\\_select\\_brush](#) -- Select a brush
- [printer\\_create\\_font](#) -- Create a new font
- [printer\\_delete\\_font](#) -- Delete a font
- [printer\\_select\\_font](#) -- Select a font
- [printer\\_logical\\_fontheight](#) -- Get logical font height
- [printer\\_draw\\_roundrect](#) -- Draw a rectangle with rounded corners
- [printer\\_draw\\_rectangle](#) -- Draw a rectangle
- [printer\\_draw\\_elipse](#) -- Draw an ellipse

[printer\\_draw\\_text](#) -- Draw text  
[printer\\_draw\\_line](#) -- Draw a line  
[printer\\_draw\\_chord](#) -- Draw a chord  
[printer\\_draw\\_pie](#) -- Draw a pie  
[printer\\_draw\\_bmp](#) -- Draw a bmp

---

[Anterior](#)  
system

[Inicio](#)  
[Subir](#)

[Siguiete](#)  
printer\_open

# LXXI. Pspell Functions

The `pspell()` functions allow you to check the spelling of a word and offer suggestions.

You need the `aspell` and `pspell` libraries, available from <http://aspell.sourceforge.net/> and <http://pspell.sourceforge.net/> respectively, and add the `--with-pspell[=dir]` option when compiling `php`.

## Tabla de contenidos

[pspell\\_new](#) -- Load a new dictionary

[pspell\\_check](#) -- Check a word

[pspell\\_suggest](#) -- Suggest spellings of a word

## LXXII. GNU Readline

Las funciones [readline\(\)](#) implementan una interfaz con la librería GNU Readline. Un ejemplo de la manera de funcionar podría ser la forma en que el Bash permite usar las flechas de dirección para insertar caracteres o desplazarse a través del historial de comandos. Debido a la naturaleza interactiva de esta librería, tendrá un uso muy reducido en la escritura de aplicaciones Web, aunque puede ser útil para scripts que han de ser ejecutados desde la consola.

La página principal del proyecto GNU Readline es <http://cnswww.cns.cwru.edu/~chet/readline/rltop.html>. Está actualizada por Chet Ramey, quien además es el autor de Bash.

### Tabla de contenidos

[readline](#) -- Lee una línea

[readline\\_add\\_history](#) -- Añade una línea al historial

[readline\\_clear\\_history](#) -- Borra el historial

[readline\\_completion\\_function](#) -- Registra una función de completitud

[readline\\_info](#) -- Establece/Obtiene diversas variables internas de readline

[readline\\_list\\_history](#) -- Lista el historial

[readline\\_read\\_history](#) -- Lee un historial

[readline\\_write\\_history](#) -- Escribe el historial

## LXXIII. Funciones GNU Recode

Este modulo contiene un interfaz para la biblioteca GNU Recode version 3.5. Para poder usar las funciones definidas en este modulo, debereis de compilar el interprete PHP con la opcion --with-recode. Para poder hacer esto debereis tener instalado en vuestro sistema GNU Recode 3.5 o superior.

La biblioteca GNU Recode convierte entre ficheros con diferentes codigos de caracteres y codificacion. Cuando esto no puede realizarse exactamente, puede desahacerse de los caracteres problematicos o crear una aproximacion. La biblioteca reconoce o produce alrededor de 150 codigos de caracteres y puede convertir ficheros entre casi todos los pares posibles. La gran mayoria de de codigos de caracteres RFC 1345 estan soportados.

### Tabla de contenidos

[recode\\_string](#) -- Recodifica una cadena literal segun una peticion de recodificacion.

[recode\\_file](#) -- Recodifica de fichero a fichero segun una peticion de recodificacion.

# LXXIV. Funciones de expresiones regulares compatibles con Perl

La sintaxis, para los patrones usados en estas funciones, es muy semejante al Perl. Las expresiones estarán encerradas por delimitadores, por ejemplo una barra de dividir (/). Cualquier carácter puede ser usado para delimitar incluso los que no son caracteres alfanuméricos o la barra invertida (\). Si el carácter delimitador ha sido usado en la propia expresión, es necesario que sea precedido por una barra inversa.

El delimitador de fin puede ser seguido por varios modificadores que afectarán al resultado. Examina [Modificadores de Patrones](#).

## Ejemplo 1. Ejemplos de patrones válidos

- `/<\w+>/`
- `|(\d{3})-\d+|Sm`
- `/(?i)php[34]/`

## Ejemplo 2. Ejemplos de patrones no válidos

- `/href='(.*)'` - falta el delimitador de fin
- `^\w+\s*\w+/J` - el modificador 'J' es desconocido
- `1-\d3-\d3-\d4|` - falta el delimitador de inicio

**Nota:** Para las funciones de expresiones compatibles con Perl se necesita PHP 4 o PHP 3.0.9 o superior.

## Tabla de contenidos

[preg\\_match](#) -- Realiza un emparejamiento dada una expresión

[preg\\_match\\_all](#) -- Realiza un completo emparejamiento de expresiones

[preg\\_replace](#) -- Lleva a cabo la búsqueda de una expresión y su sustitución

[preg\\_split](#) -- Divide una cadena dada una expresión

[preg\\_quote](#) -- Prepara los caracteres de expresiones

[preg\\_grep](#) -- Devuelve un array con los elementos que casen con el patrón

[Modificadores de Patrones](#) -- describe los modificadores posibles en los patrones de expresiones regulares (regex)

[Sintaxis de los Patrones](#) -- describe la sintaxis de PCRE regex

---

[Anterior](#)

recode\_file

[Inicio](#)

[Subir](#)

[Siguiete](#)

preg\_match

# LXXV. Funciones para expresiones regulares

Las expresiones regulares se usan en PHP para manipular cadenas complejas. Las funciones que soportan expresiones regulares son:

- [ereg\(\)](#)
- [ereg\\_replace\(\)](#)
- [eregi\(\)](#)
- [eregi\\_replace\(\)](#)
- [split\(\)](#)

En todas estas funciones, el primer argumento es una expresión regular. PHP utiliza las expresiones regulares extendidas de POSIX, definidas en POSIX 1003.2. Para una descripción completa de las expresiones regulares POSIX, ver las páginas de manual de regex incluidas en el directorio regex de la distribución de PHP. Están en formato de página de manual, por lo que se deben leer con una orden como **man /usr/local/src/regex/regex.7**.

## Ejemplo 1. Ejemplos de expresiones regulares

```
ereg("abc", $string);
/* Devuelve true si "abc"
   se encuentra en $string. */

ereg("^abc", $string);
/* Devuelve true si "abc"
   se encuentra al comienzo de $string. */

ereg("abc$", $string);
/* Devuelve true si "abc"
   se encuentra al final de $string. */

eregi("(ozilla.[23]|MSIE.3)", $HTTP_USER_AGENT);
```



```
/* Devuelve true si el navegador cliente
   es Netscape 2, 3 o MSIE 3. */

ereg("([[:alnum:]]+) ([[:alnum:]]+) ([[:alnum:]]+)",
     $string,$regs);
/* Pone tres palabras separadas por espacios
   en $regs[1], $regs[2] y $regs[3]. */

$string = ereg_replace("^","<BR>",$string);
/* Coloca la etiqueta <BR> al comienzo de $string. */

$string = ereg_replace("$","<BR>",$string);
/* Coloca la etiqueta <BR> al final de $string. */

$string = ereg_replace("\n","",$string);
/* Elimina los caracteres fin-de-línea de $string. */
```

## Tabla de contenidos

[ereg](#) -- Coincidencia de expresiones regulares

[ereg\\_replace](#) -- reemplaza expresiones regulares

[eregi](#) -- coincidencia de expresiones regulares sin diferenciar mayúsculas y minúsculas

[eregi\\_replace](#) -- reemplaza expresiones regulares sin diferenciar mayúsculas y minúsculas

[split](#) -- divide la cadena en elementos de un array según una expresión regular

[sql\\_regcase](#) -- construye una expresión regular para buscar coincidencias sin diferenciar mayúsculas y minúsculas

---

[Anterior](#)

Sintaxis de los Patrones

[Inicio](#)

[Subir](#)

[Siguiete](#)

ereg

# LXXVI. Satellite CORBA client extension

## Aviso

This module is *EXPERIMENTAL*. That means, that the behaviour of these functions, these function names, in concreto ANYTHING documented here can change in a future release of PHP WITHOUT NOTICE. Be warned, and use this module at your own risk.

The Satellite extension is used for accessing CORBA objects. You will need to set the `idl_directory=` entry in `php.ini` to a path where you store all IDL files you use.

## Tabla de contenidos

[OrbitObject](#) -- Access CORBA objects

[OrbitEnum](#) -- Use CORBA enums

[OrbitStruct](#) -- Use CORBA structs

[satellite\\_caught\\_exception](#) -- See if an exception was caught from the previous function

[satellite\\_exception\\_id](#) -- Get the repository id for the latest excetpion.

[satellite\\_exception\\_value](#) -- Get the exception struct for the latest exception

# LXXVII. Funciones Semáforo y de memoria compartida

Este módulo provee funciones semáforo utilizando los semaforos de System V. Los semáforos pueden usarse para obtener acceso exclusivo a algun recurso del ordenador en cuestión, o para limitar el número de procesos que pueden usar un recurso simultaneamente.

Este módulo provee tambien funciones de memoria compartida, usando el compartimiento de memoria de System V. La memoria compartida puede usarse para proveer acceso a variables globales. Los diferentes demonios http e incluso otros programas, (como Perl, C, ...) son capaces de utilizar estos datos, para intercambiarlos de modo global. Recuerde que, la memoria compartida NO es segura para los accesos simultáneos. Use los semáforos para obtener sincronismo.

**Tabla 1. Limites de la memoria compartida del SO Unix**

SHMMAX	máximo tamaño de memoria compartida, normalmente 131072 bytes
SHMMIN	minimo tamaño de memoria compartida, por lo general 1 byte
SHMMNI	máxima cantidad de segmentos de memoria compartida, normalmente 100
SHMSEG	máximo de memoria compartida por proceso, normalmente 6

## Tabla de contenidos

[sem\\_get](#) -- obtiene la identificacion de un semáforo (semaphore id)

[sem\\_acquire](#) -- adquiere un semáforo, lo toma para sí

[sem\\_release](#) -- release a semaphore

[shm\\_attach](#) -- Crea o abre un segmento de memoria compartida

[shm\\_detach](#) -- Finaliza conexión con un segmento de memoria compartida

[shm\\_remove](#) -- Elimina memoria compartida del sistema Unix

[shm\\_put\\_var](#) -- Inserta o actualiza una variable en la memoria compartida

[shm\\_get\\_var](#) -- Devuelve una variable de la memoria compartida

[shm\\_remove\\_var](#) -- Elimina una variable de la memoria compartida

---

[Anterior](#)

satellite\_exception\_value

[Inicio](#)

[Subir](#)

[Siguiete](#)

sem\_get

## LXXVIII. SESAM database functions

SESAM/SQL-Server is a mainframe database system, developed by Fujitsu Siemens Computers, Germany. It runs on high-end mainframe servers using the operating system BS2000/OSD.

In numerous productive BS2000 installations, SESAM/SQL-Server has proven ...

- the ease of use of Java-, Web- and client/server connectivity,
- the capability to work with an availability of more than 99.99%,
- the ability to manage tens and even hundreds of thousands of users.

Now there is a PHP3 SESAM interface available which allows database operations via PHP-scripts.

**Configuration notes:** There is no standalone support for the PHP SESAM interface, it works only as an integrated Apache module. In the Apache PHP module, this [SESAM interface is configured](#) using Apache directives.

**Tabla 1. SESAM Configuration directives**

Directive	Meaning
<code>php3_sesam_oml</code>	<p>Name of BS2000 PLAM library containing the loadable SESAM driver modules. Required for using SESAM functions.</p> <p>Example:</p> <pre>php3_sesam_oml \$.SYSLNK.SESAM-SQL.030</pre>
<code>php3_sesam_configfile</code>	<p>Name of SESAM application configuration file. Required for using SESAM functions.</p> <p>Example:</p> <pre>php3_sesam_configfile \$SESAM.SESAM.CONF.AW</pre> <p>It will usually contain a configuration like (see SESAM reference manual):</p>

	CNF=B NAM=K NOTYPE
<code>php3_sesam_messagecatalog</code>	Name of SESAM message catalog file. In most cases, this directive is not necessary. Only if the SESAM message file is not installed in the system's BS2000 message file table, it can be set with this directive.  Example:  <pre>php3_sesam_messagecatalog \$.SYSMES.SESAM-SQL.030</pre>

In addition to the configuration of the PHP/SESAM interface, you have to configure the SESAM-Database server itself on your mainframe as usual. That means:

- starting the SESAM database handler (DBH), and
- connecting the databases with the SESAM database handler

To get a connection between a PHP script and the database handler, the CNF and NAM parameters of the selected SESAM configuration file must match the id of the started database handler.

In case of distributed databases you have to start a SESAM/SQL-DCN agent with the distribution table including the host and database names.

The communication between PHP (running in the POSIX subsystem) and the database handler (running outside the POSIX subsystem) is realized by a special driver module called SQLSCI and SESAM connection modules using common memory. Because of the common memory access, and because PHP is a static part of the web server, database accesses are very fast, as they do not require remote accesses via ODBC, JDBC or UTM.

Only a small stub loader (SESMOD) is linked with PHP, and the SESAM connection modules are pulled in from SESAM's OML PLAM library. In the [configuration](#), you must tell PHP the name of this PLAM library, and the file link to use for the SESAM configuration file (As of SESAM V3.0, SQLSCI is available in the SESAM Tool Library, which is part of the standard distribution).

Because the SQL command quoting for single quotes uses duplicated single quotes (as opposed to a single quote preceded by a backslash, used in some other databases), it is advisable to set the PHP configuration directives [php3\\_magic\\_quotes\\_gpc](#) and [php3\\_magic\\_quotes\\_sybase](#) to On for all PHP scripts using the SESAM interface.

**Runtime considerations:** Because of limitations of the BS2000 process model, the driver can be loaded only after the Apache server has forked off its server child processes. This will slightly slow down the initial SESAM request of each child, but subsequent accesses will respond at full speed.

When explicitly defining a Message Catalog for SESAM, that catalog will be loaded each time the driver is loaded (i.e., at the initial SESAM request). The BS2000 operating system prints a message after successful load of the message catalog, which will be sent to Apache's error\_log file. BS2000 currently does not allow suppression of this message, it will slowly fill up the log.

Make sure that the SESAM OML PLAM library and SESAM configuration file are readable by the user id running the web server. Otherwise, the server will be unable to load the driver, and will not allow to call any SESAM functions. Also, access to the database must be granted to the user id under which the Apache server is running. Otherwise, connections to the SESAM database handler will fail.

**Cursor Types:** The result cursors which are allocated for SQL "select type" queries can be either "sequential" or "scrollable". Because of the larger memory overhead needed by "scrollable" cursors, the default is "sequential".

When using "scrollable" cursors, the cursor can be freely positioned on the result set. For each "scrollable" query, there are global default values for the scrolling type (initialized to: SESAM\_SEEK\_NEXT) and the scrolling offset which can either be set once by [sesam\\_seek\\_row\(\)](#) or each time when fetching a row using [sesam\\_fetch\\_row\(\)](#). When fetching a row using a "scrollable" cursor, the following post-processing is done for the global default values for the scrolling type and scrolling offset:

**Tabla 2. Scrolled Cursor Post-Processing**

Scroll Type	Action
SESAM_SEEK_NEXT	none
SESAM_SEEK_PRIOR	none
SESAM_SEEK_FIRST	set scroll type to SESAM_SEEK_NEXT
SESAM_SEEK_LAST	set scroll type to SESAM_SEEK_PRIOR
SESAM_SEEK_ABSOLUTE	Auto-Increment internal offset value
SESAM_SEEK_RELATIVE	none. (maintain global default <i>offset</i> value, which allows for, e.g., fetching each 10th row backwards)

**Porting note:** Because in the PHP world it is natural to start indexes at zero (rather than 1), some adaptations have been made to the SESAM interface: whenever an indexed array is starting with index 1 in the native SESAM interface, the PHP interface uses index 0 as a starting point. E.g., when retrieving columns with [sesam\\_fetch\\_row\(\)](#), the first column has the index 0, and the subsequent columns have indexes up to (but not including) the column count (`$array["count"]`). When porting SESAM applications from other high level languages to PHP, be aware of this changed interface. Where appropriate, the description of the respective php sesam functions include a note that the index is zero based.

**Security concerns:** When allowing access to the SESAM databases, the web server user should only have as little privileges as possible. For most databases, only read access privilege should be granted. Depending on your usage scenario, add more access rights as you see fit. Never allow full control to any database for any user from the 'net! Restrict access to php scripts which must administer the database by using password control and/or SSL security.

**Migration from other SQL databases:** No two SQL dialects are ever 100% compatible. When porting SQL applications from other database interfaces to SESAM, some adaptation may be required. The

following typical differences should be noted:

- Vendor specific data types

Some vendor specific data types may have to be replaced by standard SQL data types (e.g., TEXT could be replaced by VARCHAR(max. size)).

- Keywords as SQL identifiers

In SESAM (as in standard SQL), such identifiers must be enclosed in double quotes (or renamed).

- Display length in data types

SESAM data types have a precision, not a display length. Instead of int(4) (intended use: integers up to '9999'), SESAM requires simply int for an implied size of 31 bits. Also, the only datetime data types available in SESAM are: DATE, TIME(3) and TIMESTAMP(3).

- SQL types with vendor-specific unsigned, zerofill, or auto\_increment attributes

Unsigned and zerofill are not supported. Auto\_increment is automatic (use "INSERT ... VALUES(\*, ...)" instead of "... VALUES(0, ...)" to take advantage of SESAM-implied auto-increment.

- **int ... DEFAULT '0000'**

Numeric variables must not be initialized with string constants. Use **DEFAULT 0** instead. To initialize variables of the datetime SQL data types, the initialization string must be prefixed with the respective type keyword, as in: CREATE TABLE exmpl ( xtime timestamp(3) DEFAULT TIMESTAMP '1970-01-01 00:00:00.000' NOT NULL );

- **\$count = xxxx\_num\_rows();**

Some databases promise to guess/estimate the number of the rows in a query result, even though the returned value is grossly incorrect. SESAM does not know the number of rows in a query result before actually fetching them. If you REALLY need the count, try **SELECT COUNT(...) WHERE ...**, it will tell you the number of hits. A second query will (hopefully) return the results.

- **DROP TABLE thename;**

In SESAM, in the **DROP TABLE** command, the table name must be either followed by the keyword RESTRICT or CASCADE. When specifying RESTRICT, an error is returned if there are dependent objects (e.g., VIEWS), while with CASCADE, dependent objects will be deleted along with the specified table.

**Notes on the use of various SQL types:** SESAM does not currently support the BLOB type. A future version of SESAM will have support for BLOB.

At the PHP interface, the following type conversions are automatically applied when retrieving SQL fields:

**Tabla 3. SQL to PHP Type Conversions**

SQL Type	PHP Type
SMALLINT, INTEGER	"integer"
NUMERIC, DECIMAL, FLOAT, REAL, DOUBLE	"double"



DATE, TIME, TIMESTAMP	"string"
VARCHAR, CHARACTER	"string"

When retrieving a complete row, the result is returned as an array. Empty fields are not filled in, so you will have to check for the existence of the individual fields yourself (use [isset\(\)](#) or [empty\(\)](#) to test for empty fields). That allows more user control over the appearance of empty fields (than in the case of an empty string as the representation of an empty field).

**Support of SESAM's "multiple fields" feature:** The special "multiple fields" feature of SESAM allows a column to consist of an array of fields. Such a "multiple field" column can be created like this:

### Ejemplo 1. Creating a "multiple field" column

```
CREATE TABLE multi_field_test (
    pkey CHAR(20) PRIMARY KEY,
    multi(3) CHAR(12)
)
```

and can be filled in using:

### Ejemplo 2. Filling a "multiple field" column

```
INSERT INTO multi_field_test (pkey, multi(2..3) )
VALUES ('Second', <'first_val', 'second_val'>)
```

Note that (like in this case) leading empty sub-fields are ignored, and the filled-in values are collapsed, so that in the above example the result will appear as multi(1..2) instead of multi(2..3).

When retrieving a result row, "multiple columns" are accessed like "inlined" additional columns. In the example above, "pkey" will have the index 0, and the three "multi(1..3)" columns will be accessible as indices 1 through 3.

For specific SESAM details, please refer to [the SESAM/SQL-Server documentation \(english\)](#) or [the SESAM/SQL-Server documentation \(german\)](#), both available online, or use the respective manuals.

### Tabla de contenidos

- [sesam\\_connect](#) -- Open SESAM database connection
- [sesam\\_disconnect](#) -- Detach from SESAM connection
- [sesam\\_settransaction](#) -- Set SESAM transaction parameters
- [sesam\\_commit](#) -- Commit pending updates to the SESAM database
- [sesam\\_rollback](#) -- Discard any pending updates to the SESAM database
- [sesam\\_execimm](#) -- Execute an "immediate" SQL-statement
- [sesam\\_query](#) -- Perform a SESAM SQL query and prepare the result
- [sesam\\_num\\_fields](#) -- Return the number of fields/columns in a result set
- [sesam\\_field\\_name](#) -- Return one column name of the result set
- [sesam\\_diagnostic](#) -- Return status information for last SESAM call
- [sesam\\_fetch\\_result](#) -- Return all or part of a query result
- [sesam\\_affected\\_rows](#) -- Get number of rows affected by an immediate query

[sesam\\_errormsg](#) -- Returns error message of last SESAM call  
[sesam\\_field\\_array](#) -- Return meta information about individual columns in a result  
[sesam\\_fetch\\_row](#) -- Fetch one row as an array  
[sesam\\_fetch\\_array](#) -- Fetch one row as an associative array  
[sesam\\_seek\\_row](#) -- Set scrollable cursor mode for subsequent fetches  
[sesam\\_free\\_result](#) -- Releases resources for the query

---

[Anterior](#)

shm\_remove\_var

[Inicio](#)

[Subir](#)

[Siguiete](#)

sesam\_connect

## LXXIX. Session handling functions

Session support in PHP consists of a way to preserve certain data across subsequent accesses. This enables you to build more customized applications and increase the appeal of your web site.

If you are familiar with the session management of PHPLIB, you will notice that some concepts are similar to PHP's session support.

A visitor accessing your web site is assigned an unique id, the so-called session id. This is either stored in a cookie on the user side or is propagated in the URL.

The session support allows you to register arbitrary numbers of variables to be preserved across requests. When a visitor accesses your site, PHP will check automatically (if `session.auto_start` is set to 1) or on your request (explicitly through [session\\_start\(\)](#) or implicitly through [session\\_register\(\)](#)) whether a specific session id has been sent with the request. If this is the case, the prior saved environment is recreated.

All registered variables are serialized after the request finishes. Registered variables which are undefined are marked as being not defined. On subsequent accesses, these are not defined by the session module unless the user defines them later.

The [track\\_vars](#) and `register_globals` configuration settings influence how the session variables get stored and restored.

**Nota:** As of PHP 4.0.3, [track\\_vars](#) is always turned on.

If [track\\_vars](#) is enabled and `register_globals` is disabled, only members of the global associative array `$HTTP_SESSION_VARS` can be registered as session variables. The restored session variables will only be available in the array `$HTTP_SESSION_VARS`.

### Ejemplo 1. Registering a variable with [track\\_vars](#) enabled

```
<?php
session_register("count");
$HTTP_SESSION_VARS["count"]++;
?>
```

If `register_globals` is enabled, then all global variables can be registered as session variables and the session variables will be restored to corresponding global variables.

### **Ejemplo 2. Registering a variable with `register_globals` enabled**

```
<?php
session_register("count");
$count++;
?>
```

If both [track\\_vars](#) and `register_globals` are enabled, then the global variables and the `$HTTP_SESSION_VARS` entries will reference the same value.

There are two methods to propagate a session id:

- Cookies
- URL parameter

The session module supports both methods. Cookies are optimal, but since they are not reliable (clients are not bound to accept them), we cannot rely on them. The second method embeds the session id directly into URLs.

PHP is capable of doing this transparently when compiled with `--enable-trans-sid`. If you enable this option, relative URIs will be changed to contain the session id automatically. Alternatively, you can use the constant `SID` which is defined, if the client did not send the appropriate cookie. `SID` is either of the form `session_name=session_id` or is an empty string.

The following example demonstrates how to register a variable, and how to link correctly to another page using `SID`.

### **Ejemplo 3. Counting the number of hits of a single user**

```

<?php
session_register ("count");
$count++;
?>

Hello visitor, you have seen this page <?php echo $count; ?> times.<p>

<php?
# the <?=SID?> is necessary to preserve the session id
# in the case that the user has disabled cookies
?>

To continue, <A HREF="nextpage.php?<?=SID?>">click here</A>

```

The <?=SID?> is not necessary, if `--enable-trans-sid` was used to compile PHP.

**Nota:** Non-relative URLs are assumed to point to external sites and hence don't append the SID, as it would be a security risk to leak the SID to a different server.

To implement database storage, or any other storage method, you will need to use [session\\_set\\_save\\_handler\(\)](#) to create a set of user-level storage functions.

The session management system supports a number of configuration options which you can place in your `php.ini` file. We will give a short overview.

- `session.save_handler` defines the name of the handler which is used for storing and retrieving data associated with a session. Defaults to `files`.
- `session.save_path` defines the argument which is passed to the save handler. If you choose the default files handler, this is the path where the files are created. Defaults to `/tmp`.

#### Aviso

If you leave this set to a world-readable directory, such as `/tmp` (the default), other users on the server may be able to hijack sessions by getting the list of files in that directory.

- `session.name` specifies the name of the session which is used as cookie name. It should only contain alphanumeric characters. Defaults to `PHPSESSID`.
- `session.auto_start` specifies whether the session module starts a session automatically on request startup. Defaults to `0` (disabled).
- `session.cookie_lifetime` specifies the lifetime of the cookie in seconds which is sent to the browser. The value `0` means "until the browser is closed." Defaults to `0`.
- `session.serialize_handler` defines the name of the handler which is used to serialize/deserialize data. Currently, a PHP internal format (name `php`) and `WDDX` is supported (name `wddx`). `WDDX` is only available, if PHP is compiled with [WDDX support](#). Defaults to

php.

- `session.gc_probability` specifies the probability that the gc (garbage collection) routine is started on each request in percent. Defaults to 1.
- `session.gc_maxlifetime` specifies the number of seconds after which data will be seen as 'garbage' and cleaned up.
- `session.referer_check` contains the substring you want to check each HTTP Referer for. If the Referer was sent by the client and the substring was not found, the embedded session id will be marked as invalid. Defaults to the empty string.
- `session.entropy_file` gives a path to an external resource (file) which will be used as an additional entropy source in the session id creation process. Examples are `/dev/random` or `/dev/urandom` which are available on many Unix systems.
- `session.entropy_length` specifies the number of bytes which will be read from the file specified above. Defaults to 0 (disabled).
- `session.use_cookies` specifies whether the module will use cookies to store the session id on the client side. Defaults to 1 (enabled).
- `session.cookie_path` specifies path to set in `session_cookie`. Defaults to `/`.
- `session.cookie_domain` specifies domain to set in `session_cookie`. Default is none at all.
- `session.cache_limiter` specifies cache control method to use for session pages (`nocache/private/private_no_expire/public`). Defaults to `nocache`.
- `session.cache_expire` specifies time-to-live for cached session pages in minutes, this has no effect for `nocache` limiter. Defaults to 180.
- `session.use_trans_sid` whether transient sid support is enabled or not if enabled by compiling with `--enable-trans-sid`. Defaults to 1 (enabled).
- `url_rewriter.tags` specifies which html tags are rewritten to include session id if transient sid support is enabled. Defaults to `a=href,area=href,frame=src,input=src,form=fakeentry`

**Nota:** Session handling was added in PHP 4.0.

## Tabla de contenidos

[session\\_start](#) -- Initialize session data

[session\\_destroy](#) -- Destroys all data registered to a session

[session\\_name](#) -- Get and/or set the current session name

[session\\_module\\_name](#) -- Get and/or set the current session module

[session\\_save\\_path](#) -- Get and/or set the current session save path

[session\\_id](#) -- Get and/or set the current session id

[session\\_register](#) -- Register one or more variables with the current session

[session\\_unregister](#) -- Unregister a variable from the current session

[session\\_unset](#) -- Free all session variables

[session\\_is\\_registered](#) -- Find out if a variable is registered in a session

[session\\_get\\_cookie\\_params](#) -- Get the session cookie parameters

[session\\_set\\_cookie\\_params](#) -- Set the session cookie parameters

[session\\_decode](#) -- Decodes session data from a string

[session\\_encode](#) -- Encodes the current session data as a string

[session\\_set\\_save\\_handler](#) -- Sets user-level session storage functions

[session\\_cache\\_limiter](#) -- Get and/or set the current cache limiter

[session\\_write\\_close](#) -- Write session data and end session

---

[Anterior](#)

sesam\_free\_result

[Inicio](#)

[Subir](#)

[Siguiete](#)

session\_start

## LXXX. Shared Memory Functions

Shmop is an easy to use set of functions that allows php to read, write, create and delete UNIX shared memory segments. The functions will not work on windows, as it does not support shared memory. To use shmop you will need to compile php with the `--enable-shmop` parameter in your configure line.

**Nota:** The functions explained in the chapter begin all with `shm_()` in PHP 4.0.3, but in PHP 4.0.4 and later versions these names are changed to begin with `shmop_()`.

### Ejemplo 1. Shared Memory Operations Overview

```
<?php

// Create 100 byte shared memory block with system id if 0xff3
$shm_id = shmop_open(0xff3, "c", 0644, 100);
if(!$shm_id) {
    echo "Couldn't create shared memory segment\n";
}

// Get shared memory block's size
$shm_size = shmop_size($shm_id);
echo "SHM Block Size: ".$shm_size. " has been created.\n";

// Lets write a test string into shared memory
$shm_bytes_written = shmop_write($shm_id, "my shared memory block", 0);
if($shm_bytes_written != strlen("my shared memory block")) {
    echo "Couldn't write the entire length of data\n";
}

// Now lets read the string back
$my_string = shmop_read($shm_id, 0, $shm_size);
if(!$my_string) {
    echo "Couldn't read from shared memory block\n";
}
echo "The data inside shared memory was: ".$my_string."\n";

//Now lets delete the block and close the shared memory segment
if(!shmop_delete($shm_id)) {
    echo "Couldn't mark shared memory block for deletion.
}
shmop_close($shm_id);
```



?>

## Tabla de contenidos

[shmop\\_open](#) -- Create or open shared memory block

[shmop\\_read](#) -- Read data from shared memory block

[shmop\\_write](#) -- Write data into shared memory block

[shmop\\_size](#) -- Get size of shared memory block

[shmop\\_delete](#) -- Delete shared memory block

[shmop\\_close](#) -- Close shared memory block

---

[Anterior](#)

session\_write\_close

[Inicio](#)

[Subir](#)

[Siguiete](#)

shmop\_open

# LXXXI. Shockwave Flash functions

PHP offers the ability to create Shockwave Flash files via Paul Haeberli's libswf module. You can download libswf at <http://reality.sgi.com/grafica/flash/>. Once you have libswf all you need to do is to configure `--with-swf[=DIR]` where DIR is a location containing the directories include and lib. The include directory has to contain the swf.h file and the lib directory has to contain the libswf.a file. If you unpack the libswf distribution the two files will be in one directory. Consequently you will have to copy the files to the proper location manually.

Once you've successfully installed PHP with Shockwave Flash support you can then go about creating Shockwave files from PHP. You would be surprised at what you can do, take the following code:

## Ejemplo 1. SWF example

```
<?php
swf_openfile ("test.swf", 256, 256, 30, 1, 1, 1);
swf_ortho2 (-100, 100, -100, 100);
swf_defineline (1, -70, 0, 70, 0, .2);
swf_definerect (4, 60, -10, 70, 0, 0);
swf_definerect (5, -60, 0, -70, 10, 0);
swf_addcolor (0, 0, 0, 0);

swf_definefont (10, "Mod");
swf_fontsize (5);
swf_fontslant (10);
swf_definetext (11, "This be Flash wit PHP!", 1);

swf_pushmatrix ();
swf_translate (-50, 80, 0);
swf_placeobject (11, 60);
swf_popmatrix ();

for ($i = 0; $i < 30; $i++) {
```

```

    $p = $i/(30-1);
    swf_pushmatrix ();
    swf_scale (1-($p*.9), 1, 1);
    swf_rotate (60*$p, 'z');
    swf_translate (20+20*$p, $p/1.5, 0);
    swf_rotate (270*$p, 'z');
    swf_addcolor ($p, 0, $p/1.2, -$p);
    swf_placeobject (1, 50);
    swf_placeobject (4, 50);
    swf_placeobject (5, 50);
    swf_popmatrix ();
    swf_showframe ();
}

for ($i = 0; $i < 30; $i++) {
    swf_removeobject (50);
    if (($i%4) == 0) {
        swf_showframe ();
    }
}

swf_startdoaction ();
swf_actionstop ();
swf_enddoaction ();

swf_closefile ();
?>

```

It will produce the animation found at the following [url](#).

**Nota:** SWF support was added in PHP4 RC2.

### Tabla de contenidos

[swf\\_openfile](#) -- Open a new Shockwave Flash file

[swf\\_closefile](#) -- Close the current Shockwave Flash file

[swf\\_labelframe](#) -- Label the current frame

[swf\\_showframe](#) -- Display the current frame

[swf\\_setframe](#) -- Switch to a specified frame

[swf\\_getframe](#) -- Get the frame number of the current frame

[swf\\_mulcolor](#) -- Sets the global multiply color to the rgba value specified  
[swf\\_addcolor](#) -- Set the global add color to the rgba value specified  
[swf\\_placeobject](#) -- Place an object onto the screen  
[swf\\_modifyobject](#) -- Modify an object  
[swf\\_removeobject](#) -- Remove an object  
[swf\\_nextid](#) -- Returns the next free object id  
[swf\\_startdoaction](#) -- Start a description of an action list for the current frame  
[swf\\_actiongotoframe](#) -- Play a frame and then stop  
[swf\\_actiongeturl](#) -- Get a URL from a Shockwave Flash movie  
[swf\\_actionnextframe](#) -- Go forward one frame  
[swf\\_actionprevframe](#) -- Go backwards one frame  
[swf\\_actionplay](#) -- Start playing the flash movie from the current frame  
[swf\\_actionstop](#) -- Stop playing the flash movie at the current frame  
[swf\\_actiontogglequality](#) -- Toggle between low and high quality  
[swf\\_actionwaitforframe](#) -- Skip actions if a frame has not been loaded  
[swf\\_actionsettarget](#) -- Set the context for actions  
[swf\\_actiongotolabel](#) -- Display a frame with the specified label  
[swf\\_enddoaction](#) -- End the current action  
[swf\\_defineline](#) -- Define a line  
[swf\\_definerect](#) -- Define a rectangle  
[swf\\_definepoly](#) -- Define a polygon  
[swf\\_startshape](#) -- Start a complex shape  
[swf\\_shapelinesolid](#) -- Set the current line style  
[swf\\_shapefilloff](#) -- Turns off filling  
[swf\\_shapefillsolid](#) -- Set the current fill style to the specified color  
[swf\\_shapefillbitmapclip](#) -- Set current fill mode to clipped bitmap  
[swf\\_shapefillbitmaptile](#) -- Set current fill mode to tiled bitmap  
[swf\\_shapemoveto](#) -- Move the current position  
[swf\\_shapelineto](#) -- Draw a line  
[swf\\_shapecurveto](#) -- Draw a quadratic bezier curve between two points  
[swf\\_shapecurveto3](#) -- Draw a cubic bezier curve  
[swf\\_shapearc](#) -- Draw a circular arc  
[swf\\_endshape](#) -- Completes the definition of the current shape  
[swf\\_definefont](#) -- Defines a font  
[swf\\_setfont](#) -- Change the current font  
[swf\\_fontsize](#) -- Change the font size

[swf\\_fontslant](#) -- Set the font slant  
[swf\\_fontracking](#) -- Set the current font tracking  
[swf\\_getfontinfo](#) -- The height in pixels of a capital A and a lowercase x  
[swf\\_definetext](#) -- Define a text string  
[swf\\_textwidth](#) -- Get the width of a string  
[swf\\_definebitmap](#) -- Define a bitmap  
[swf\\_getbitmapinfo](#) -- Get information about a bitmap  
[swf\\_startsymbol](#) -- Define a symbol  
[swf\\_endsymbol](#) -- End the definition of a symbol  
[swf\\_startbutton](#) -- Start the definition of a button  
[swf\\_addbuttonrecord](#) -- Controls location, appearance and active area of the current button  
[swf\\_oncondition](#) -- Describe a transition used to trigger an action list  
[swf\\_endbutton](#) -- End the definition of the current button  
[swf\\_viewport](#) -- Select an area for future drawing  
[swf\\_ortho](#) -- Defines an orthographic mapping of user coordinates onto the current viewport  
[swf\\_ortho2](#) -- Defines 2D orthographic mapping of user coordinates onto the current viewport  
[swf\\_perspective](#) -- Define a perspective projection transformation  
[swf\\_polarview](#) -- Define the viewer's position with polar coordinates  
[swf\\_lookat](#) -- Define a viewing transformation  
[swf\\_pushmatrix](#) -- Push the current transformation matrix back unto the stack  
[swf\\_popmatrix](#) -- Restore a previous transformation matrix  
[swf\\_scale](#) -- Scale the current transformation  
[swf\\_translate](#) -- Translate the current transformations  
[swf\\_rotate](#) -- Rotate the current transformation  
[swf\\_posround](#) -- Enables or Disables the rounding of the translation when objects are placed or moved

---

[Anterior](#)  
shmop\_close

[Inicio](#)  
[Subir](#)

[Siguiete](#)  
swf\_openfile

## LXXXII. Funciones SNMP

Para usar las funciones SNMP en Unix necesita instalar el paquete [UCD SNMP](#). En Windows estas funciones están solamente disponibles en NT y no en Win95/98.

Importante: Para usar el paquete UCD SNMP, necesita definir `NO_ZEROLENGTH_COMMUNITY` a 1 antes de compilarlo. Después de configurar UCD SNMP, edite `config.h` y busque `NO_ZEROLENGTH_COMMUNITY`. Descomente la línea `#define`. Debería de verse como sigue:

```
#define NO_ZEROLENGTH_COMMUNITY 1
```

Si ve faltas de segmentación desconocidas en combinación con los comandos SNMP, no siga las siguientes instrucciones. Si no desea recompilar UCD SNMP, puede compilar PHP con la opción `--enable-ucd-snmp-hack` la cual trabajará entorno a las mismas características.

### Tabla de contenidos

[snmpget](#) -- Va a buscar un objeto SNMP

[snmpset](#) -- Va a buscar un objeto SNMP

[snmpwalk](#) -- Búsqueda por un árbol de información acerca de un entidad de red

[snmpwalkoid](#) -- Búsqueda por un árbol de información acerca de un entidad de red

[snmp\\_get\\_quick\\_print](#) -- Va a buscar el valor actual de la biblioteca UCD estableciendo `quick_print`

[snmp\\_set\\_quick\\_print](#) -- Establece el valor de `quick_print` con el de la biblioteca UCD SNMP.

## LXXXIII. Socket functions

### Aviso

This module is *EXPERIMENTAL*. That means, that the behaviour of these functions, these function names, in concreto ANYTHING documented here can change in a future release of PHP WITHOUT NOTICE. Be warned, and use this module at your own risk.

The socket extension implements a low-level interface to the socket communication functions, providing the possibility to act as a socket server as well as a client.

For a more generic client-side socket interface, see [fsockopen\(\)](#) and [pfsockopen\(\)](#).

When using the socket functions described here, it is important to remember that while many of them have identical names to their C counterparts, they often have different declarations. Please be sure to read the descriptions to avoid confusion.

That said, those unfamiliar with socket programming can still find a lot of useful material in the appropriate Unix man pages, and there is a great deal of tutorial information on socket programming in C on the web, much of which can be applied, with slight modifications, to socket programming in PHP.

### Ejemplo 1. Socket example: Simple TCP/IP server

This example shows a simple talkback server. Change the `address` and `port` variables to suit your setup and execute. You may then connect to the server with a command similar to: **telnet 192.168.1.53 10000** (where the address and port match your setup). Anything you type will then be output on the server side, and echoed back to you. To disconnect, enter 'quit'.

```
<?php
error_reporting(E_ALL);

/* Allow the script to hang around waiting for connections. */
set_time_limit(0);

$address = '192.168.1.53';
$port = 10000;

if (($sock = socket(AF_INET, SOCK_STREAM, 0)) < 0) {
    echo "socket() failed: reason: " . strerror($sock) . "\n";
}

if (($ret = bind($sock, $address, $port)) < 0) {
    echo "bind() failed: reason: " . strerror($ret) . "\n";
}
```

```

if (($ret = listen($sock, 5)) < 0) {
    echo "listen() failed: reason: " . strerror($ret) . "\n";
}

do {
    if (($msgsock = accept_connect($sock)) < 0) {
        echo "accept_connect() failed: reason: " . strerror($msgsock) . "\n";
        break;
    }
    do {
        $buf = '';
        $ret = read($msgsock, $buf, 2048);
        if ($ret < 0) {
            echo "read() failed: reason: " . strerror($ret) . "\n";
            break 2;
        }
        if ($ret == 0) {
            break 2;
        }
        $buf = trim($buf);
        if ($buf == 'quit') {
            close($msgsock);
            break 2;
        }
        $talkback = "PHP: You said '$buf'.\n";
        write($msgsock, $talkback, strlen($talkback));
        echo "$buf\n";
    } while (true);
    close($msgsock);
} while (true);

close($sock);
?>

```

## Ejemplo 2. Socket example: Simple TCP/IP client

This example shows a simple, one-shot HTTP client. It simply connects to a page, submits a HEAD request, echoes the reply, and exits.



```

<?php
error_reporting(E_ALL);

echo "<h2>TCP/IP Connection</h2>\n";

/* Get the port for the WWW service. */
$service_port = getservbyname('www', 'tcp');

/* Get the IP address for the target host. */
$address = gethostbyname('www.php.net');

/* Create a TCP/IP socket. */
$socket = socket(AF_INET, SOCK_STREAM, 0);
if ($socket < 0) {
    echo "socket() failed: reason: " . strerror($socket) . "\n";
} else {
    "socket() successful: " . strerror($socket) . "\n";
}

echo "Attempting to connect to '$address' on port '$service_port'...";
$result = connect($socket, $address, $service_port);
if ($result < 0) {
    echo "connect() failed.\nReason: ($result) " . strerror($result) . "\n";
} else {
    echo "OK.\n";
}

$in = "HEAD / HTTP/1.0\r\n\r\n";
$out = '';

echo "Sending HTTP HEAD request...";
write($socket, $in, strlen($in));
echo "OK.\n";

echo "Reading response:\n\n";
while (read($socket, $out, 2048)) {
    echo $out;
}

echo "Closing socket...";
close($socket);
echo "OK.\n\n";
?>

```

## Tabla de contenidos

[accept](#) [connect](#) -- Accepts a connection on a socket.

[bind](#) -- Binds a name to a socket.

[connect](#) -- Initiates a connection on a socket.

[listen](#) -- Listens for a connection on a socket.

[socket](#) -- Create a socket (endpoint for communication).

[strerror](#) -- Return a string describing a socket error.

---

[Anterior](#)

snmp\_set\_quick\_print

[Inicio](#)

[Subir](#)

[Siguiete](#)

accept\_connect

# LXXXIV. Funciones de cadenas

Todas estas funciones manipulan cadenas de varias maneras. En las secciones sobre expresiones regulares y manejo de URL se pueden encontrar secciones más especializadas.

## Tabla de contenidos

[AddCSlashes](#) -- Marca una cadena con barras al estilo del C

[AddSlashes](#) -- Marca una cadena con barras

[bin2hex](#) -- Convierte datos binarios en su representación hexadecimal

[chop](#) -- Elimina espacios sobrantes al final

[chr](#) -- Devuelve un caracter específico

[chunk\\_split](#) -- Divide una cadena en trozos más pequeños

[convert\\_cyr\\_string](#) -- Convierte de un juego de caracteres Cirílico a otro

[count\\_chars](#) -- Devuelve información sobre los caracteres usados en una cadena

[crc32](#) -- Calcula el polinomio crc32 de una cadena

[crypt](#) -- Encripta una cadena mediante DES

[echo](#) -- Da salida a una o más cadenas

[explode](#) -- Divide una cadena por otra

[get\\_html\\_translation\\_table](#) -- Devuelve la tabla de traducción utilizada por

[htmlspecialchars\(\)](#) y [htmlentities\(\)](#)

[get\\_meta\\_tags](#) -- Extrae todas las etiquetas meta de un archivo y retorna una matriz

[hebrew](#) -- Convierte Hebreo lógico a texto visual

[hebrevc](#) -- Convierte Hebreo lógico a texto visual con conversión de saltos de línea

[htmlentities](#) -- Convierte todos los caracteres aplicables a entidades HTML

[htmlspecialchars](#) -- Convierte caracteres especiales a entidades HTML

[implode](#) -- Unir elementos de una matriz mediante una cadena

[join](#) -- Une elementos de una tabla mediante una cadena

[levenshtein](#) -- Calcula la distancia Levenshtein entre dos cadenas

[ltrim](#) -- Elimina el espacio en blanco del principio de una cadena

[md5](#) -- Calcula el hash md5 de una cadena

[metaphone](#) -- Calcula la clave "metáfono" de una cadena

[nl2br](#) -- Convierte nuevas líneas a saltos de línea HTML

[ord](#) -- Devuelve el valor ASCII de un caracter

[parse\\_str](#) -- Divide la cadena en variables

[print](#) -- Emite una cadena

[printf](#) -- Emite una cadena con formato

[quoted\\_printable\\_decode](#) -- Convierte una cadena con marcación imprimible a una cadena de 8 bits

[quotemeta](#) -- Quote meta characters

[rtrim](#) -- Elimina espacios en blanco al final de la cadena.

[sscanf](#) -- Trocea la entrada desde una cadena según un formato dado

[setlocale](#) -- Fija la información de localidad

[similar\\_text](#) -- Calcula la similitud entre dos cadenas

[soundex](#) -- Calcula la clave soundex de una cadena

[sprintf](#) -- Devuelve una cadena con formato

[strcasecmp](#) -- Comparación de cadenas insensible a mayúsculas y minúsculas y segura en modo binario

[strchr](#) -- Encuentra la primera aparición de un caracter

[strcmp](#) -- Comparación de cadenas con seguridad binaria

[strcspn](#) -- Encuentra la longitud del elemento inicial que no coincide con la máscara

[strip\\_tags](#) -- Elimina marcas HTML y PHP de una cadena

[stripslashes](#) -- Desmarca la cadena marcada con [addslashes\(\)](#)

[stripslashes](#) -- Desmarca la cadena marcada con [addslashes\(\)](#)

[stristr](#) -- [stristr\(\)](#) sin tener en cuenta mayúsculas o minúsculas

[strlen](#) -- Obtiene la longitud de la cadena

[strnatcmp](#) -- Compara cadenas usando un algoritmo de "orden natural"

[strnatcasecmp](#) -- Comparación de cadenas insensible a mayúsculas y minúsculas usando un algoritmo de "orden natural"

[strncmp](#) -- Comparación de los n primeros caracteres de cadenas, con seguridad binaria

[str\\_pad](#) -- Rellena una cadena con otra hasta una longitud dada

[strpos](#) -- Encuentra la posición de la primera aparición de una cadena

[strrchr](#) -- Encuentra la última aparición de un caracter en una cadena

[str\\_repeat](#) -- Repite una cadena

[strrev](#) -- Invierte una cadena

[strrpos](#) -- Encuentra la posición de la última aparición de un caracter en una cadena

[strspn](#) -- Encuentra la longitud del segmento inicial que coincide con la máscara

[strstr](#) -- Encuentra la primera aparición de una cadena

[strtok](#) -- Divide una cadena en elementos

[strtolower](#) -- Pasa a minúsculas una cadena

[strtoupper](#) -- Pasa a mayúsculas una cadena

[str\\_replace](#) -- Sustituye todas las apariciones de la aguja en el pajar por la cadena

[strtr](#) -- Traduce ciertos caracteres

[substr](#) -- Devuelve parte de una cadena

[substr\\_count](#) -- Cuenta el número de apariciones de la subcadena

[substr\\_replace](#) -- Sustituye texto en una parte de una cadena

[trim](#) -- Elimina espacios del principio y final de una cadena

[ucfirst](#) -- Pasar a mayúsculas el primer caracter de una cadena

[ucwords](#) -- Pone en mayúsculas el primer caracter de cada palabra de una cadena

[wordwrap](#) -- Corta una cadena en un número dado de caracteres usando un caracter de ruptura de cadenas.

---

[Anterior](#)

strerror

[Inicio](#)

[Subir](#)

[Siguiete](#)

AddCSlashes

# LXXXV. Funciones de Sybase

## Tabla de contenidos

[sybase\\_affected\\_rows](#) -- obtiene el número de filas afectadas por la última consulta

[sybase\\_close](#) -- cierra una conexión Sybase

[sybase\\_connect](#) -- abre una conexión con un servidor Sybase

[sybase\\_data\\_seek](#) -- mueve el puntero interno de la fila

[sybase\\_fetch\\_array](#) -- carga una fila como un array

[sybase\\_fetch\\_field](#) -- obtiene la información del campo

[sybase\\_fetch\\_object](#) -- carga una fila como un objeto

[sybase\\_fetch\\_row](#) -- obtiene una fila como un array enumerado

[sybase\\_field\\_seek](#) -- establece el offset de un campo

[sybase\\_free\\_result](#) -- libera el resultado de la memoria

[sybase\\_num\\_fields](#) -- obtiene el número de campos de un resultado

[sybase\\_num\\_rows](#) -- obtiene el número de filas de un resultado

[sybase\\_pconnect](#) -- abre una conexión con Sybase persistente

[sybase\\_query](#) -- envía una consulta a Sybase

[sybase\\_result](#) -- obtiene datos de un resultado

[sybase\\_select\\_db](#) -- selecciona una base de datos Sybase

# LXXXVI. Funciones URL

## Tabla de contenidos

[base64\\_decode](#) -- decodifica datos cifrados con MIME base64

[base64\\_encode](#) -- Codifica datos en MIME base64

[parse\\_url](#) -- Analiza una URL y devuelve sus componentes

[urldecode](#) -- decodifica URL-cifradas en una cadena de texto

[urlencode](#) -- Codifica una URL en una cadena de texto

---

# LXXXVII. Funciones sobre variables

## Tabla de contenidos

[doubleval](#) -- Obtiene el valor double (decimal) de una variable.

[empty](#) -- Determina si una variable está definida

[gettype](#) -- Obtiene el tipo de una variable.

[intval](#) -- Obtiene el valor entero de una variable.

[is\\_array](#) -- Averigua si una variable es un array.

[is\\_double](#) -- Averigua si una variable es un valor double (número decimal).

[is\\_float](#) -- Averigua si una variable es un flotante.

[is\\_int](#) -- Averigua si una variable es un valor entero.

[is\\_integer](#) -- Averigua si una variable es un valor entero.

[is\\_long](#) -- Averigua si una variable es un valor entero.

[is\\_object](#) -- Averigua si una variable es un objeto.

[is\\_real](#) -- Averigua si una variable es un número real.

[is\\_string](#) -- Averigua si una variable es una cadena de caracteres (string).

[isset](#) -- Determina si una variable está definida

[settype](#) -- Establece el tipo de una variable.

[strval](#) -- Obtiene una cadena de caracteres a partir de una variable.

[unset](#) -- Destruye una variable dada



## LXXXVIII. Funciones WDDX

Estas funciones permiten el uso de [WDDX](#).

Debe saber que todas las funciones que serializan variables usan el primer elemento de un array para determinar si este ha de serializarse en forma de array o como estructura. Si el primer elemento esta indexado por una cadena, se serializa como estructura, y en caso contrario, como array.

### Ejemplo 1. Serializacion de un valor simple

```
<?php
print wddx_serialize_value("Ejemplo de PHP a paquete WDDX", "Paquete PHP");
?>
```

Este ejemplo producira:

```
<wddxPacket version='0.9'><header comment='Paquete PHP'></header><data>
<string>Ejemplo de PHP a paquete WDDX</string></data></wddxPacket>
```

### Ejemplo 2. Uso de paquetes incrementales

```
<?php
$pi = 3.1415926;
$packet_id = wddx_packet_start("PHP");
wddx_add_vars($packet_id, "pi");

/* Suponga que $ciudades se ha obtenido de una base de datos */
$ciudades = array("Austin", "Novato", "Seattle");
wddx_add_vars($packet_id, "ciudades");

$packet = wddx_packet_end($packet_id);
print $packet;
?>
```

Este ejemplo producira:

```
<wddxPacket version='0.9'><header comment='PHP' /><data><struct>  
<var name='pi'><number>3.1415926</number></var><var name='ciudades'>  
<array length='3'><string>Austin</string><string>Novato</string>  
<string>Seattle</string></array></var></struct></data></wddxPacket>
```

## Tabla de contenidos

[wddx\\_serialize\\_value](#) -- Serializa un valor simple en un paquete WDDX

[wddx\\_serialize\\_vars](#) -- Serializa variables en un paquete WDDX

[wddx\\_packet\\_start](#) -- Comienza un nuevo paquete WDDX con una estructura dentro

[wddx\\_packet\\_end](#) -- Finaliza un paquete WDDX con el identificador dado

[wddx\\_add\\_vars](#) -- Finaliza un paquete WDDX con el identificador dado

[wddx\\_deserialize](#) -- Des-serializa un paquete WDDX

---

[Anterior](#)

unset

[Inicio](#)

[Subir](#)

[Siguiete](#)

wddx\_serialize\_value

# LXXXIX. Funciones de intérprete XML

## Introducción

### Acerca de XML

XML (eXtensible Markup Language) es un formato de información para el intercambio de documentos estructurado en la "Web". Es un estándar definido por el consorcio de la "World Wide Web" (W3C). Se puede encontrar información sobre XML y tecnologías relacionadas en <http://www.w3.org/XML/>.

## Instalación

Esta extensión usa expat, que se puede encontrar en <http://www.jclark.com/xml/>. El Makefile que viene con expat no crea una biblioteca por defecto, se puede usar esta regla de make para eso:

```
libexpat.a: $(OBJS)
    ar -rc $@ $(OBJS)
    ranlib $@
```

Se puede conseguir un paquete RPM de expat en <http://sourceforge.net/projects/expat/>.

Nota que si se usa Apache-1.3.7 o posterior, ya tienes la biblioteca requerida expat. Simplemente, configura PHP usando `--with-xml` (sin ninguna ruta adicional) y usará automáticamente la biblioteca expat incluida en Apache.

En UNIX, ejecuta **configure** con la opción `--with-xml`. La biblioteca expat debería ser instalada en algún lugar donde el compilador pueda encontrarlo. Si se compila PHP como un módulo para Apache 1.3.9 o posterior, PHP automáticamente usará la biblioteca integrada expat de Apache. Puede necesitar establecer `CPPFLAGS` y `LDFLAGS` en su entorno antes de ejecutar "configure" si se ha instalado expat en algún lugar exótico.

Compila PHP. ¡*Ta-tam!* Ya debería estar.

## Sobre Esta Extensión

Esta extensión de PHP implementa soporte para expat de James Clarkin en PHP. Este conjunto de herramientas permite interpretar, pero no validar, documentos XML. Soporta tres [codificaciones de caracteres](#) fuente, también proporcionados por PHP: US-ASCII, ISO-8859-1 y UTF-8. UTF-16 no está soportado.

Esta extensión permite [crear intérpretes de XML](#) y definir entonces *gestores* para diferentes eventos XML. Cada intérprete XML tiene también unos cuantos [parámetros](#) que se pueden ajustar.

Los gestores de eventos XML definidos son:

**Tabla 1. Gestores de XML soportados**

Función PHP para establecer gestor	Descripción del evento
<a href="#">xml_set_element_handler()</a>	Los eventos de elemento ("element") se producen cuando el intérprete XML encuentra etiquetas de comienzo o fin. Hay gestores separados para etiquetas de comienzo y etiquetas de fin.
<a href="#">xml_set_character_data_handler()</a>	La información de caracteres es, por definición, todo el contenido no "marcado" de los documentos XML, incluidos los espacios en blanco entre etiquetas. Nota que el intérprete XML no añade o elimina ningún espacio en blanco, depende de la aplicación (de ti) decidir si el espacio en blanco es significativo.
<a href="#">xml_set_processing_instruction_handler()</a>	Los programadores de PHP deberían estar ya familiarizados con las instrucciones de procesado (PI). <code>&lt;?php ?&gt;</code> es una instrucción de procesado, donde <i>php</i> se denomina el "objetivo de procesado". El manejo de éstos es específico a cada aplicación, salvo que todos los objetivos PI que comienzan con "XML" están reservados.
<a href="#">xml_set_default_handler()</a>	Todo lo que no va a otro gestor, va al gestor por defecto. Se tendrán en el gestor por defecto cosas como las declaraciones de tipos de documento y XML.
<a href="#">xml_set_unparsed_entity_decl_handler()</a>	Este gestor se llamará para la declaración de una entidad no analizada (NDATA).
<a href="#">xml_set_notation_decl_handler()</a>	Este gestor se llama para la declaración de una anotación.
<a href="#">xml_set_external_entity_ref_handler()</a>	Este gestor se llama cuando el intérprete XML encuentra una referencia a una entidad general interpretada externa. Puede ser una referencia a un archivo o URL, por ejemplo. Ver <a href="#">el ejemplo de entidad externa</a> para demostración.

## Case Folding

Las funciones manejadoras de elementos pueden tomar sus nombres de elementos "*case-folded*". Case-folding se define en el estándar XML como "un proceso aplicado a una secuencia de caracteres, en el cual aquellos identificados como sin-mayúsculas son reemplazados por sus equivalentes en mayúsculas". En otras palabras, cuando se trata de XML, case-folding simplemente significa poner en mayúsculas.

Por defecto, todos los nombres de elementos que se pasan a las funciones gestoras están "pasados a mayúsculas". Esta conducta puede ser observada y controlada por el analizador XML con las funciones [xml\\_parser\\_get\\_option\(\)](#) y [xml\\_parser\\_set\\_option\(\)](#), respectivamente.

## Códigos de Error

Las siguientes constantes se definen para códigos de error XML (como los devuelve [xml\\_parse\(\)](#)):

XML\_ERROR\_NONE  
XML\_ERROR\_NO\_MEMORY  
XML\_ERROR\_SYNTAX

XML\_ERROR\_NO\_ELEMENTS  
XML\_ERROR\_INVALID\_TOKEN  
XML\_ERROR\_UNCLOSED\_TOKEN  
XML\_ERROR\_PARTIAL\_CHAR  
XML\_ERROR\_TAG\_MISMATCH  
XML\_ERROR\_DUPLICATE\_ATTRIBUTE  
XML\_ERROR\_JUNK\_AFTER\_DOC\_ELEMENT  
XML\_ERROR\_PARAM\_ENTITY\_REF  
XML\_ERROR\_UNDEFINED\_ENTITY  
XML\_ERROR\_RECURSIVE\_ENTITY\_REF  
XML\_ERROR\_ASYNC\_ENTITY  
XML\_ERROR\_BAD\_CHAR\_REF  
XML\_ERROR\_BINARY\_ENTITY\_REF  
XML\_ERROR\_ATTRIBUTE\_EXTERNAL\_ENTITY\_REF  
XML\_ERROR\_MISPLACED\_XML\_PI  
XML\_ERROR\_UNKNOWN\_ENCODING  
XML\_ERROR\_INCORRECT\_ENCODING  
XML\_ERROR\_UNCLOSED\_CDATA\_SECTION  
XML\_ERROR\_EXTERNAL\_ENTITY\_HANDLING

## Codificación de caracteres

La extensión XML de PHP soporta el conjunto de caracteres [Unicode](#) a través de diferentes *codificaciones de caracteres*. Hay dos tipos de codificaciones de caracteres, *codificación de fuente* y *codificación de destino*. La representación interna de PHP del documento está siempre codificada con UTF-8.

La codificación de fuente se hace cuando un documento XML es [interpretado](#). Al [crear un intérprete XML](#), se puede especificar una codificación de fuente (esta codificación no se puede cambiar más tarde durante el tiempo de vida del intérprete XML). Las codificaciones de fuente soportadas son ISO-8859-1, US-ASCII y UTF-8. Las dos primeras son codificaciones de byte-único, lo que significa que cada carácter se representa por un solo byte. UTF-8 puede codificar caracteres compuestos por un número variable de bits (hasta 21) en de uno a cuatro bytes. La codificación fuente por defecto usada por PHP es ISO-8859-1.

La codificación de destino se hace cuando PHP pasa datos a las funciones gestoras XML. Cuando se crea un intérprete XML, la codificación de destino se crea igual a la codificación de fuente, pero se puede cambiar en cualquier momento. La codificación de destino afectará a la información de los caracteres así como a los nombres de las etiquetas y a los objetivos de instrucciones de procesado.

Si el intérprete XML encuentra caracteres fuera del rango que su codificación de fuente es capaz de representar, devolverá un error.

Si PHP encuentra caracteres en el documento XML interpretado que no pueden ser representados en la codificación de destino elegida, los caracteres problema serán "degradados". Actualmente, esto significa que tales caracteres se reemplazan por un signo de interrogación.

## Algunos Ejemplos

Aquí hay algunos ejemplos de archivos de comandos PHP que interpretan documentos XML.

## Ejemplos de Estructuras de Elementos XML

Este primer ejemplo muestra la estructura del elemento inicio en un documento con indentación.

### Ejemplo 1. Muestra la Estructura del Elemento XML

```
$file = "data.xml";
$depth = array();

function startElement($parser, $name, $attrs) {
    global $depth;
    for ($i = 0; $i < $depth[$parser]; $i++) {
        print " ";
    }
    print "$name\n";
    $depth[$parser]++;
}

function endElement($parser, $name) {
    global $depth;
    $depth[$parser]--;
}

$xml_parser = xml_parser_create();
xml_set_element_handler($xml_parser, "startElement", "endElement");
if (!$fp = fopen($file, "r")) {
    die("could not open XML input");
}

while ($data = fread($fp, 4096)) {
    if (!xml_parse($xml_parser, $data, feof($fp))) {
        die(sprintf("XML error: %s at line %d",
            xml_error_string(xml_get_error_code($xml_parser)),
            xml_get_current_line_number($xml_parser)));
    }
}

xml_parser_free($xml_parser);
```

## Ejemplo de Mapeo de Etiquetas XML

## Ejemplo 2. Traduciendo XML a HTML

Este ejemplo transforma etiquetas de un documento XML directamente a etiquetas HTML. Los elementos no encontrados en el "array de traducción ("map array") son ignorados. Por supuesto, este ejemplo solamente funcionará con un tipo de documentos XML específico.

```
$file = "data.xml";
$map_array = array(
    "BOLD"      => "B",
    "EMPHASIS" => "I",
    "LITERAL"  => "TT"
);

function startElement($parser, $name, $attrs) {
    global $map_array;
    if ($htmltag = $map_array[$name]) {
        print "<$htmltag>";
    }
}

function endElement($parser, $name) {
    global $map_array;
    if ($htmltag = $map_array[$name]) {
        print "</$htmltag>";
    }
}

function characterData($parser, $data) {
    print $data;
}

$xml_parser = xml_parser_create();
// usa case-folding para que estemos seguros de encontrar la etiqueta
// en $map_array
xml_parser_set_option($xml_parser, XML_OPTION_CASE_FOLDING, true);
xml_set_element_handler($xml_parser, "startElement", "endElement");
xml_set_character_data_handler($xml_parser, "characterData");
if (!$fp = fopen($file, "r")) {
    die("could not open XML input");
}

while ($data = fread($fp, 4096)) {
    if (!xml_parse($xml_parser, $data, feof($fp))) {
        die(sprintf("XML error: %s at line %d",
            xml_error_string(xml_get_error_code($xml_parser)),
            xml_get_current_line_number($xml_parser)));
    }
}
xml_parser_free($xml_parser);
```

# Ejemplo de Entidad Externa XML

Este ejemplo resalta el código XML. Ilustra cómo usar un gestor de referencia de entidades externas para incluir y analizar otros documentos, así como cuántos PIs pueden ser procesados, y un modo de determinar "confianza" para PIs que contienen código.

Los documentos XML que se pueden usar en este ejemplo se encuentran bajo el ejemplo (xmltest.xml y xmltest2.xml.)

## Ejemplo 3. Ejemplo de Entidades Externas

```
$file = "xmltest.xml";

function trustedFile($file) {
    // solamente confía en archivos locales que nos pertenezcan
    if (!eregi("^([a-z]+)://", $file)
        && fileowner($file) == getmyuid()) {
        return true;
    }
    return false;
}

function startElement($parser, $name, $attribs) {
    print "&lt;<font color=\"#0000cc\">$name</font>";
    if (sizeof($attribs)) {
        while (list($k, $v) = each($attribs)) {
            print " <font color=\"#009900\">$k</font>=<font
                color=\"#990000\">$v</font>\"";
        }
    }
    print "&gt;";
}

function endElement($parser, $name) {
    print "&lt;/<font color=\"#0000cc\">$name</font>&gt;";
}

function characterData($parser, $data) {
    print "<b>$data</b>";
}

function PIHandler($parser, $target, $data) {
    switch (strtolower($target)) {
        case "php":
            global $parser_file;
            // Si el documento analizado es "de confianza", diremos
            // que es seguro ejecutar código PHP en su interior.
            // Si no, en vez de ello mostrará el código.
            if (trustedFile($parser_file[$parser])) {
                eval($data);
            } else {
                printf("Untrusted PHP code: <i>%s</i>",
                    htmlspecialchars($data));
            }
        }
    }
}
```



```

        }
        break;
    }
}

function defaultHandler($parser, $data) {
    if (substr($data, 0, 1) == "&" && substr($data, -1, 1) == ";") {
        printf('<font color="#aa00aa">%s</font>',
            htmlspecialchars($data));
    } else {
        printf('<font size="-1">%s</font>',
            htmlspecialchars($data));
    }
}

function externalEntityRefHandler($parser, $openEntityNames, $base, $systemId,
    $publicId) {
    if ($systemId) {
        if (!list($parser, $fp) = new_xml_parser($systemId)) {
            printf("Could not open entity %s at %s\n", $openEntityNames,
                $systemId);
            return false;
        }
        while ($data = fread($fp, 4096)) {
            if (!xml_parse($parser, $data, feof($fp))) {
                printf("XML error: %s at line %d while parsing entity %s\n",
                    xml_error_string(xml_get_error_code($parser)),
                    xml_get_current_line_number($parser), $openEntityNames);
                xml_parser_free($parser);
                return false;
            }
        }
        xml_parser_free($parser);
        return true;
    }
    return false;
}

function new_xml_parser($file) {
    global $parser_file;

    $xml_parser = xml_parser_create();
    xml_parser_set_option($xml_parser, XML_OPTION_CASE_FOLDING, 1);
    xml_set_element_handler($xml_parser, "startElement", "endElement");
    xml_set_character_data_handler($xml_parser, "characterData");
    xml_set_processing_instruction_handler($xml_parser, "PIHandler");
    xml_set_default_handler($xml_parser, "defaultHandler");
    xml_set_external_entity_ref_handler($xml_parser, "externalEntityRefHandler");

    if (!$fp = @fopen($file, "r")) {
        return false;
    }
    if (!is_array($parser_file)) {

```

```

        settype($parser_file, "array");
    }
    $parser_file[$xml_parser] = $file;
    return array($xml_parser, $fp);
}

if (!(list($xml_parser, $fp) = new_xml_parser($file))) {
    die("could not open XML input");
}

print "<pre>";
while ($data = fread($fp, 4096)) {
    if (!xml_parse($xml_parser, $data, feof($fp))) {
        die(sprintf("XML error: %s at line %d\n",
                    xml_error_string(xml_get_error_code($xml_parser)),
                    xml_get_current_line_number($xml_parser)));
    }
}
print "</pre>";
print "parse complete\n";
xml_parser_free($xml_parser);

?>

```

#### Ejemplo 4. xmltest.xml

```

<?xml version='1.0'?>
<!DOCTYPE chapter SYSTEM "/just/a/test.dtd" [
<!ENTITY plainEntity "FOO entity">
<!ENTITY systemEntity SYSTEM "xmltest2.xml">
]>
<chapter>
<TITLE>Title &plainEntity;</TITLE>
<para>
<informaltable>
<tgroup cols="3">
<tbody>
<row><entry>a1</entry><entry morerows="1">b1</entry><entry>c1</entry></row>
<row><entry>a2</entry><entry>c2</entry></row>
<row><entry>a3</entry><entry>b3</entry><entry>c3</entry></row>
</tbody>
</tgroup>
</informaltable>
</para>
&systemEntity;
<sect1 id="about">
<title>About this Document</title>
<para>
<!-- this is a comment -->
<?php print 'Hi! This is PHP version '.phpversion(); ?>
</para>

```

```
</sect1>
</chapter>
```

Este archivo se incluye desde `xmltest.xml`:

### Ejemplo 5. `xmltest2.xml`

```
<?xml version="1.0"?>
<!DOCTYPE foo [
<!ENTITY testEnt "test entity">
]>
<foo>
  <element attrib="value"/>
  &testEnt;
  <?php print "This is some more PHP code being executed."; ?>
</foo>
```

### Tabla de contenidos

[xml\\_parser\\_create](#) -- crea un analizador de XML

[xml\\_set\\_object](#) -- Usa un analizador XML dentro de un objeto

[xml\\_set\\_element\\_handler](#) -- establece gestores de los elementos principio y fin

[xml\\_set\\_character\\_data\\_handler](#) -- Establece gestores de datos de caracteres

[xml\\_set\\_processing\\_instruction\\_handler](#) -- Establece el gestor de instrucciones de procesado (PI)

[xml\\_set\\_default\\_handler](#) -- set up default handler

[xml\\_set\\_unparsed\\_entity\\_decl\\_handler](#) -- Establece un gestor de declaraciones de entidades no analizadas

[xml\\_set\\_notation\\_decl\\_handler](#) -- Establece gestores de declaraciones de notación

[xml\\_set\\_external\\_entity\\_ref\\_handler](#) -- Establece gestores de referencia de entidades externas

[xml\\_parse](#) -- comienza a analizar un documento XML

[xml\\_get\\_error\\_code](#) -- obtiene el código de error del analizador XML

[xml\\_error\\_string](#) -- obtiene la cadena de error del analizador XML

[xml\\_get\\_current\\_line\\_number](#) -- obtiene el número de línea actual de un analizador XML

[xml\\_get\\_current\\_column\\_number](#) -- Obtiene el número de columna actual para un analizador XML.

[xml\\_get\\_current\\_byte\\_index](#) -- obtiene el índice del byte actual para un analizador XML

[xml\\_parser\\_free](#) -- Libera un analizador XML

[xml\\_parser\\_set\\_option](#) -- establece las opciones de un analizador XML

[xml\\_parser\\_get\\_option](#) -- obtiene las opciones de un analizador XML

[utf8\\_decode](#) -- Convierte una cadena codificada UTF-8 a ISO-8859-1

[utf8\\_encode](#) -- codifica una cadena ISO-8859-1 a UTF-8

---

[Anterior](#)

wddx\_deserialize

[Inicio](#)

[Subir](#)

[Siguiete](#)

xml\_parser\_create

## XC. XSLT functions

### Aviso

This module is *EXPERIMENTAL*. That means, that the behaviour of these functions, these function names, in concreto ANYTHING documented here can change in a future release of PHP WITHOUT NOTICE. Be warned, and use this module at your own risk.

## Introduction

### About XSLT and Sablotron

XSLT (Extensible Stylesheet Language (XSL) Transformations) is a language for transforming XML documents into other XML documents. It is a standard defined by The World Wide Web consortium (W3C). Information about XSLT and related technologies can be found at <http://www.w3.org/TR/xslt>.

## Installation

This extension uses Sablotron and expat, which can both be found at <http://www.gingerall.com/>. Binaries are provided as well as source.

On UNIX, run **configure** with the `--with-sablot` and `--enable-sablot-errors-descriptive` options. The Sablotron library should be installed somewhere your compiler can find it.

## About This Extension

This PHP extension implements support Sablotron from Ginger Alliance in PHP. This toolkit lets you transform XML documents into other documents, including new XML documents, but also into HTML or other target formats. It basically provides a standardized and portable

template mechanism, separating content and design of a website.

## **Tabla de contenidos**

[xslt\\_closelog](#) -- Clear the logfile for a given instance of Sablotron

[xslt\\_create](#) -- Create a new XSL processor.

[xslt\\_errno](#) -- Return the current error number

[xslt\\_error](#) -- Return the current error string

[xslt\\_fetch\\_result](#) -- Fetch a (named) result buffer

[xslt\\_free](#) -- Free XSLT processor

[xslt\\_openlog](#) -- Set a logfile for XSLT processor messages

[xslt\\_output\\_begintransform](#) -- unknown

[xslt\\_output\\_endtransform](#) -- unknown

[xslt\\_output\\_process](#) -- unknown

[xslt\\_run](#) -- Apply a XSLT stylesheet to a file.

[xslt\\_set\\_sax\\_handler](#) -- Set SAX handlers for a XSLT processor

[xslt\\_transform](#) -- unknown

---

[Anterior](#)

utf8\_encode

[Inicio](#)

[Subir](#)

[Siguiete](#)

xslt\_closelog

# XCI. YAZ

The `yaz()` functions wrap the YAZ API. The home page of the project is <http://www.indexdata.dk/yaz/>. Information about the `phpyaz` module can be found at <http://www.indexdata.dk/phpyaz/>.

PHP/YAZ is much simpler to use than the C API for YAZ but less flexible. The intent is to make it easy to build basic client functions. It supports persistent stateless connections very similar to those offered by the various SQL APIs that are available for PHP. This means that sessions are stateless but shared amongst users, thus saving the connect and INIT steps in many cases.

Before compiling PHP with the PHP/YAZ module you'll need the YAZ toolkit. Build YAZ and install it. Build PHP with your favourite modules and add option `--with-yaz`. Your task is roughly the following:

```
gunzip -c yaz-1.6.tar.gz|tar xf -
gunzip -c php-4.0.X.tar.gz|tar xf -
cd yaz-1.6
./configure --prefix=/usr
make
make install
cd ../php-4.0.X
./configure --with-yaz=/usr/bin
make
make install
```

PHP/YAZ keeps track of connections with targets (Z-Associations). A positive integer represents the ID of a particular association.

The script below demonstrates the parallel searching feature of the API. When invoked it either prints a query form (if no arguments are supplied) or if there are arguments (term and one or more hosts) it searches the targets in array host.

## Ejemplo 1. YAZ()

```
$num_hosts = count ($host);
if (empty($term) || count($host) == 0) {
    echo '<form method="get">
    <input type="checkbox"
    name="host[]" value="bagel.indexdata.dk/gils">
        GILS test
    <input type="checkbox"
    name="host[]" value="localhost:9999/Default">
        local test
    <input type="checkbox" checked="1"
    name="host[]" value="z3950.bell-labs.com/books">
        BELL Labs Library
    <br>
    RPN Query:
    <input type="text" size="30" name="term">
    <input type="submit" name="action" value="Search">
    ';
} else {
    echo 'You searched for ' . htmlspecialchars($term) . '<br>';
    for ($i = 0; $i > $num_hosts; $i++) {
        $id[] = yaz_connect($host[$i]);
        yaz_syntax($id[$i], "sutrs");
        yaz_search($id[$i], "rpn", $term);
    }
    yaz_wait();
    for ($i = 0; $i < $num_hosts; $i++) {
        echo '<hr>' . $host[$i] . ":";
        $error = yaz_error($id[$i]);
        if (!empty($error)) {
            echo "Error: $error";
        } else {
            $hits = yaz_hits($id[$i]);
            echo "Result Count $hits";
        }
        echo '<dl>';
        for ($p = 1; $p <= 10; $p++) {
            $rec = yaz_record($id[$i], $p, "string");
            if (empty($rec)) continue;
        }
    }
}
```

```
        echo " <dt><b>$p</b></dt><dd>" ;
        echo ereg_replace("\n", "<br>\n", $rec) ;
        echo "</dd>" ;
    }
    echo '</dl>' ;
}
}
```

## Tabla de contenidos

[yaz\\_addinfo](#) -- Returns additional error information

[yaz\\_close](#) -- Closes a YAZ connection

[yaz\\_connect](#) -- Returns a positive association ID on success; zero on failure

[yaz\\_errno](#) -- Returns error number

[yaz\\_error](#) -- Returns error description

[yaz\\_hits](#) -- Returns number of hits for last search

[yaz\\_range](#) -- Specifies the maximum number of records to retrieve

[yaz\\_record](#) -- Returns a record

[yaz\\_search](#) -- Prepares for a search

[yaz\\_syntax](#) -- Specifies the preferred record syntax for retrieval

[yaz\\_wait](#) -- Executes queries

---

[Anterior](#)

xslt\_transform

[Inicio](#)

[Subir](#)

[Siguiete](#)

yaz\_addinfo



## XCII. NIS funciona

NIS ( anteriormente llamado Paginas Amarillas ) permite la administracion de red de los archivos de administracion importantes (e.g.El archivo de contraseñas). Para mas informacion dirigirse a las paginas de ayuda de NIS y a la direccion. [Introduccion a YP/NIS](#) Hay tambien un libro llamado [gestionando NFS Y NIS](#) por Hal Stern.

Para obtener estas funciones de trabajo, usted tiene que configure PHP con `-- con- yp`.

### Tabla de contenidos

[yp\\_get\\_default\\_domain](#) -- Trae el valor por omision de dominios de maquina NIS.

[yp\\_order](#) -- Retorna el numero de orden para el mapa.

[yp\\_master](#) -- Retorna el nombre del servidor de NIS maestro para el mapa.

[yp\\_match](#) -- Retorna la linea compañera (pareja).

[yp\\_first](#) -- devuelve la primera clave emparejada con el nombrado mapa.

[yp\\_next](#) -- Devuelve la siguiente clave tecleada en el nombre de mapa

[yp\\_errno](#) -- Retorna el codigo de error de la operacion previa.

[yp\\_err\\_string](#) -- devuelve el mensaje de error asociado con la operacion previa.Util que indica el problema exacto.

## XCIII. Zip File Functions (Read Only Access)

This module uses the functions of the [ZZIPLib](#) library by Guido Draheim to transparently read ZIP compressed archives and the files inside them.

Please note that ZZIPLib only provides a subset of functions provided in a full implementation of the ZIP compression algorithm and can only read ZIP file archives. A normal ZIP utility is needed to create the ZIP file archives read by this library.

Zip support in PHP is not enabled by default. You will need to use the `--with-zip` configuration option when compiling PHP to enable zip support. This module requires ZZIPLib version `>= 0.10.6`.

**Nota:** Zip support before PHP 4.1.0 is experimental. This section reflects the Zip extension as it exists in PHP 4.1.0 and later.

### Example Usage

This example opens a ZIP file archive, reads each file in the archive and prints out its contents. The `test2.zip` archive used in this example is one of the test archives in the ZZIPLib source distribution.

#### Ejemplo 1. Zip Usage Example

```
<?php
$zip = zip_open("/tmp/test2.zip");
if ($zip) {
    while ($zip_entry = zip_read($zip)) {
        echo "Name:           " . zip_entry_name($zip_entry) . "\n";
        echo "Actual Filesize:  " . zip_entry_filesize($zip_entry) . "\n";
        echo "Compressed Size:     " . zip_entry_compressedsize($zip_entry) . "\n";
        echo "Compression Method: " . zip_entry_compressionmethod($zip_entry) . "\n";

        if (zip_entry_open($zip, $zip_entry, "r")) {
            echo "File Contents:\n";
            $buf = zip_entry_read($zip_entry, zip_entry_filesize($zip_entry));
            echo "$buf\n";

            zip_entry_close($zip_entry);
        }
        echo "\n";
    }

    zip_close($zip);
}
?>
```

## Tabla de contenidos

[zip\\_close](#) -- Close a Zip File Archive

[zip\\_entry\\_close](#) -- Close a Directory Entry

[zip\\_entry\\_compressedsize](#) -- Retrieve the Compressed Size of a Directory Entry

[zip\\_entry\\_compressionmethod](#) -- Retrieve the Compression Method of a Directory Entry

[zip\\_entry\\_filesize](#) -- Retrieve the Actual File Size of a Directory Entry

[zip\\_entry\\_name](#) -- Retrieve the Name of a Directory Entry

[zip\\_entry\\_open](#) -- Open a Directory Entry for Reading

[zip\\_entry\\_read](#) -- Read From an Open Directory Entry

[zip\\_open](#) -- Open a Zip File Archive

[zip\\_read](#) -- Read Next Entry in a Zip File Archive

---

[Anterior](#)

[Inicio](#)

[Siguiete](#)

yp\_err\_string

[Subir](#)

zip\_close

## XCV. Funciones de Compresión

Este módulo usa la función de [zlib](#) de Jean-loup Gailly y Mark Adler para leer y grabar archivos comprimidos .gz, de un modo transparente. Con este módulo, es requisito usar una versión de zlib igual o posterior a 1.0.9.

Este módulo contiene versiones de la mayoría de las funciones de [Sistema de archivos](#) que funcionan con los archivos comprimidos con gzip (y con los no-comprimidos también, pero no con conectores (sockets)).

### Pequeño código de ejemplo

Abre un archivo temporal y escribe en él, una cadena de prueba, y luego presenta el contenido del archivo dos veces

#### Ejemplo 1. Ejemplo de Zlib

```
<?php
$filename = tempnam('/tmp', 'zlibtest').'.gz';
print "<html>\n<head></head>\n<body>\n<pre>\n";
$s = "Sólo es una prueba, prueba, prueba,prueba, prueba, prueba!\n";
// Abre el archivo para escribirlo con máximo de compresión
$zp = gzopen($filename, "w9");
// Escribe la cadena en él
gzwrite($zp, $s);
// Cierra el fichero
gzclose($zp);
// Abre el fichero para lectura
$zp = gzopen($filename, "r");
// Lee 3 caracteres
print gzread($zp, 3);
// Salida hasta el final del fichero, para cerrarlo luego.
gzpassthru($zp);
print "\n";
// Abre el fichero y muestra su contenido (por segunda vez).
if (readgzfile($filename) != strlen($s)) {
    echo "Error con las funciones zlib!";
}
```

```
unlink($filename);  
print "<pre>\n</h1></body>\n</html>\n";  
>
```

## Tabla de contenidos

[gzclose](#) -- cierra un puntero a archivo-gz abierto

[gzEOF](#) -- prueba el fin-de-archivo de un puntero de archivo-gz

[gzfile](#) -- lee el archivo gz completo en un arreglo

[gzgetc](#) -- toma caracteres de un archivo-gz

[gzgets](#) -- toma una línea del archivo apuntado

[gzgetss](#) -- toma una línea del archivo-gz apuntado y le quita los tags HTML

[gzopen](#) -- open gz-file

[gzpassthru](#) -- Devuelve el remanente de datos de un fichero-gz

[gzputs](#) -- escribe al fichero-gz que se apunta

[gzread](#) -- Lee archivos-gz en modo Binario

[gzrewind](#) -- Reposiciona al puntero de archivo-gz, al inicio de aquel

[gzseek](#) -- Posiciona el puntero del archivo-gz

[gztell](#) -- Indica la posición de lecto-escritura en el archivo

[gzwrite](#) -- Escritura de ficheros gz en modo Binario

[readgzfile](#) -- devuelve el fichero-gz

---

[Anterior](#)

zip\_read

[Inicio](#)

[Subir](#)

[Siguiete](#)

gzclose

# V. PEAR: the PHP Extension and Application Repository

## Tabla de contenidos

24. [About PEAR](#)

25. [PEAR Coding Standards](#)

XCV. [PEAR Reference Manual](#)

# Capítulo 24. About PEAR

## Tabla de contenidos

### [What is PEAR?](#)

PEAR is dedicated to Malin Bakken, born 1999-11-21 (the first PEAR code was written just two hours before she was born).

## What is PEAR?

PEAR is a code repository for PHP extensions and PHP library code inspired by TeX's CTAN and Perl's CPAN.

The purpose of PEAR is:

- to provide a consistent means for library code authors to share their code with other developers
- to give the PHP community an infrastructure for sharing code
- to define standards that help developers write portable and reusable code
- to provide tools for code maintenance and distribution

---

# Capítulo 25. PEAR Coding Standards

## Tabla de contenidos

[Indenting](#)[Control Structures](#)[Function Calls](#)[Function Definitions](#)[Comments](#)[Including Code](#)[PHP Code Tags](#)[Header Comment Blocks](#)[Using CVS](#)[Example URLs](#)[Naming Conventions](#)

**Nota:** The PEAR Coding Standards applies to code that is to become a part of PEAR, either distributed with PHP or available for download via PEAR's install tool.

## Indenting

Use an indent of 4 spaces, with no tabs. If you use Emacs to edit PEAR code, you should set `indent-tabs-mode` to `nil`. Here is an example mode hook that will set up Emacs according to these guidelines (you will need to ensure that it is called when you are editing PHP files):

```
(defun php-mode-hook ()
  (setq tab-width 4
        c-basic-offset 4
        c-hanging-comment-ender-p nil
        indent-tabs-mode
        (not
         (and (string-match "/\\(PEAR\\|pear\\)/" (buffer-file-name))
              (string-match "\\.[php]$" (buffer-file-name))))))
```

Here are vim rules for the same thing:



```
set expandtab
set shiftwidth=4
set tabstop=4
```

---

[Anterior](#)

About PEAR

[Inicio](#)

[Subir](#)

[Siguiete](#)

Control Structures

# XCV. PEAR Reference Manual

This chapter contains reference documentation for PEAR components that are distributed with PHP. It is assumed that you are already familiar with [objects and classes](#).

## Tabla de contenidos

[PEAR](#) -- PEAR base class

[PEAR\\_Error](#) -- PEAR error mechanism base class

# VI. FAQ: Frequently Asked Questions

## Tabla de contenidos

- 26. [General Information](#)
- 27. [Mailing lists](#)
- 28. [Obtaining PHP](#)
- 29. [Database issues](#)
- 30. [Installation](#)
- 31. [Build Problems](#)
- 32. [Using PHP](#)
- 33. [PHP and HTML](#)
- 34. [PHP and COM](#)
- 35. [PHP and other languages](#)
- 36. [Migrating from PHP 2 to PHP 3](#)
- 37. [Migrating from PHP 3 to PHP 4](#)
- 38. [Miscellaneous Questions](#)

# Capítulo 26. General Information

This section holds the most general questions about PHP: what it is and what it does.

1. [What is PHP?](#)
2. [What is the relation between the versions?](#)
3. [Can I run several versions of PHP at the same time?](#)
4. [What are the differences between PHP 3 and PHP 4?](#)
5. [I think I found a bug! Who should I tell?](#)

## 1. What is PHP?

From the [preface of the manual](#):

PHP is an HTML-embedded scripting language. Much of its syntax is borrowed from C, Java and Perl with a couple of unique PHP-specific features thrown in. The goal of the language is to allow web developers to write dynamically generated pages quickly.

A nice introduction to PHP by Stig Sæther Bakken can be found [here](#) on the Zend website. Also, much of the [PHP Conference Material](#) is freely available.

## 2. What is the relation between the versions?

PHP/FI 2.0 is an early and no longer supported version of PHP. PHP 3 is the successor to PHP/FI 2.0 and is a lot nicer. PHP 4 is the latest generation of PHP, which uses the [Zend engine](#) under the hood.

## 3. Can I run several versions of PHP at the same time?

Yes. See the INSTALL file that is included in the PHP 4 source distribution. Also, read the related [appendix](#).

## 4. What are the differences between PHP 3 and PHP 4?

There are [a couple of articles](#) written on this by the authors of PHP 4. Here's a list of some of the more important new features:

- Extended API module.
- Generalized build process under UNIX
- Generic web server interface that also supports multi-threaded web servers
- Improved syntax highlighter
- Native HTTP session support
- Output buffering support
- More powerful configuration system
- Reference counting

Please see the [What's new in PHP 4 overview](#) for a detailed explanation of these features and more. If you're migrating from PHP 3 to PHP 4, also read the related [appendix](#).

## **5. I think I found a bug! Who should I tell?**

You should go to the PHP Bug Database and make sure the bug isn't a known bug. If you don't see it in the database, use the reporting form to report the bug. It is important to use the bug database instead of just sending an email to one of the mailing lists because the bug will have a tracking number assigned and it will then be possible for you to go back later and check on the status of the bug. The bug database can be found at <http://bugs.php.net/>.

---

[Anterior](#)

FAQ: Frequently Asked  
Questions

[Inicio](#)

[Subir](#)

[Siguiete](#)

Mailing lists

# Capítulo 27. Mailing lists

This section holds questions about how to get in touch with the PHP community. The best way is the mailing lists.

1. [Are there any PHP mailing lists?](#)
2. [Are there any other communities?](#)
3. [Help! I can't seem to subscribe/unsubscribe to/from one of the mailing lists!](#)
4. [Is there an archive of the mailing lists anywhere?](#)
5. [What can I ask the mailing list?](#)
6. [What information should I include when posting to the mailing list?](#)

## 1. Are there any PHP mailing lists?

Of course! There are many mailing lists for several subjects. A whole list of mailing lists can be found on our [Support](#) page.

The most general mailing list is `php-general`. To subscribe, send mail to [php-general-subscribe@lists.php.net](mailto:php-general-subscribe@lists.php.net). You don't need to include anything special in the subject or body of the message. To unsubscribe, send mail to [php-general-unsubscribe@lists.php.net](mailto:php-general-unsubscribe@lists.php.net).

You can also subscribe and unsubscribe using the web interface on our [Support](#) page.

## 2. Are there any other communities?

There are countless of them around the world. We have links for example to some IRC servers and foreign language mailing lists on our [Support](#) page.

## 3. Help! I can't seem to subscribe/unsubscribe to/from one of the mailing lists!

If you have problems subscribing to or unsubscribing from the `php-general` mailing list, it may be because the mailing list software can't figure out the correct mailing address to use. If your email address was `joeblow@example.com`, you can send your subscription request

to `php-general-subscribe-joeblow@example.com@lists.php.net`, or your unsubscription request to `php-general-unsubscribe-joeblow@example.com@lists.php.net`. Use similar addresses for the other mailing lists.

#### **4. Is there an archive of the mailing lists anywhere?**

Yes, you will find a list of archive sites on the [Support](#) page. The mailing list articles are also archived as news messages. You can access the news server at <news://news.php.net/> with a news client. There is also an experimental web interface for the news server at <http://news.php.net/>

#### **5. What can I ask the mailing list?**

Since PHP is growing more and more popular by the day the traffic has increased on the php-general mailing list and as of now the list gets about 150 to 200 posts a day. Because of this it is in everyones interest that you use the list as a last resort when you have looked everywhere else.

Before you post to the list please have a look in this FAQ and the manual to see if you can find the help there. If there is nothing to be found there try out the mailing list archives (see above). If you're having problem with installing or configuring PHP please read through all included documentation and README's. If you still can't find any information that helps you out you're more than welcome to use the mailing list.

#### **6. What information should I include when posting to the mailing list?**

Posts like "I can't get PHP up and running! Help me! What is wrong?" are of absolutely no use to anyone. If you're having problems getting PHP up and running you must include what operating system you are running on, what version of PHP you're trying to set up, how you got it (pre-compiled, CVS, RPMs and so on), what you have done so far, where you got stuck and the exact error message.

This goes for any other problem as well. You have to include information on what you have done, where you got stuck, what you're trying to do and, if applicable, exact error messages. If you're having problems with your source code you need to include the part of the code that isn't working. Do not include more code than necessary though! It makes the post hard to read and a lot of people might just skip it all together because of this. If you're unsure about how much information to include in the mail it's better that you include to much than to little.

Another important thing to remember is to summarize your problem on the subject line. A subject like "HELP MEEEE!!!" or "What is the problem here?" will be ignored by the majority of the readers.

---

[Anterior](#)

General Information

[Inicio](#)

[Subir](#)

[Siguiete](#)

Obtaining PHP



# Capítulo 28. Obtaining PHP

This section has details about PHP download locations, and OS issues.

1. [Where can I obtain PHP?](#)
2. [Are pre-compiled binary versions available?](#)
3. [Where can I get libraries needed to compile some of the optional PHP extensions?](#)
4. [How do I get these libraries to work?](#)
5. [I got the latest version of the PHP source code from the CVS repository on my Windows machine, what do I need to compile it?](#)
6. [Where do I find the Browser Capabilities File?](#)

## 1. Where can I obtain PHP?

You can download PHP from any of the members of the PHP network of sites. These can be found at <http://www.php.net/>. You can also use anonymous CVS to get the absolute latest version of the source. For more information, go to <http://cvs.php.net/>.

## 2. Are pre-compiled binary versions available?

We only distribute precompiled binaries for Windows systems, as we are not able to compile PHP for every major Linux/Unix platform with every extension combination. Also note, that many Linux distributions come with PHP built in these days. Windows binaries can be downloaded from our [Downloads](#) page, for Linux binaries, please visit your distributions website.

## 3. Where can I get libraries needed to compile some of the optional PHP extensions?

**Nota:** Those marked with \* are not thread-safe libraries, and should not be used with PHP as a server module in the multi-threaded Windows web servers (IIS, Netscape). This does not matter in Unix environments, yet.

- [LDAP \(unix\)](#).

- [LDAP\\* \(unix\)](#).
- [LDAP \(unix/win\)](#) : Netscape Directory (LDAP) SDK 1.1.
- [free LDAP server](#).
- [Berkeley DB2 \(Unix/Win\)](#) : <http://www.sleepycat.com/>.
- [SNMP\\* \(Unix\)](#): .
- [GD\\* \(Unix/Win\)](#).
- [mSQL\\* \(Win\)](#).
- [mSQL\\* \(Unix\)](#).
- [MySQL\\* \(Unix\)](#).
- [IMAP\\* \(Win/Unix\)](#).
- [Sybase-CT\\* \(Linux, libc5\)](#) : Available locally.
- [FreeType \(libtff\)](#): .
- [ZLib \(Unix/Win32\)](#).
- [expat XML parser \(Unix/Win32\)](#).
- [PDFLib](#).
- [mcrypt](#).
- [mhash](#).
- [t1lib](#).
- [dmalloc](#).
- [aspell](#).
- [readline](#).

#### **4. How do I get these libraries to work?**

You will need to follow instructions provided with the library. Some of these libraries are detected automatically when you run the 'configure' script of PHP (such as the GD library), and others you will have to enable using '--with-EXTENSION' options to 'configure'. Run 'configure --help' for a listing of these.

#### **5. I got the latest version of the PHP source code from the CVS repository on my Windows machine, what do I need to compile it?**

First, you will need Microsoft Visual C++ v6 (v5 may do it also, but we do it with v6), and you will need some support files. See the manual section about building PHP from source on Windows.

#### **6. Where do I find the Browser Capabilities File?**

You can find a browscap.ini file at <http://www.cyscape.com/asp/browscap/>.

---

[Anterior](#)

Mailing lists

[Inicio](#)

[Subir](#)

[Siguiete](#)

Database issues

---

# Capítulo 29. Database issues

This section holds common questions about relation between PHP and databases. Yes, PHP can access virtually any database available today.

1. [I heard it's possible to access Microsoft SQL Server from PHP. How?](#)
2. [Can I access Microsoft Access databases?](#)
3. [I upgraded to PHP 4, and now mysql keeps telling me "Warning: MySQL: Unable to save result set in ...". What's up?](#)
4. [After installing shared MySQL support, Apache dumps core as soon as libphp4.so is loaded. Can this be fixed?](#)
5. [Why do I get an error that looks something like this: "Warning: 0 is not a MySQL result index in <file> on line <x>" or "Warning: Supplied argument is not a valid MySQL result resource in <file> on line <x>"?](#)

## 1. I heard it's possible to access Microsoft SQL Server from PHP. How?

On Windows machines, you can simply use the included ODBC support and the correct ODBC driver.

On Unix machines, you can use the Sybase-CT driver to access Microsoft SQL Servers because they are (at least mostly) protocol-compatible. Sybase has made a [free version of the necessary libraries for Linux systems](#). For other Unix operating systems, you need to contact Sybase for the correct libraries. Also see the answer to the next question.

## 2. Can I access Microsoft Access databases?

Yes. You already have all the tools you need if you are running entirely under Windows 9x/Me, or NT/2000, where you can use ODBC and Microsoft's ODBC drivers for Microsoft Access databases.

If you are running PHP on a Unix box and want to talk to MS Access on a Windows box you will need Unix ODBC drivers. [OpenLink Software](#) has Unix-based ODBC drivers that can do this. There is a free pilot program where you can download an evaluation copy that doesn't expire and prices start at \$675 for the commercial supported version.

Another alternative is to use an SQL server that has Windows ODBC drivers and use that to store the data, which you can then access from Microsoft Access (using ODBC) and PHP (using the built in drivers), or to use an intermediary file format that Access and PHP both understand, such as flat files or dBase databases. On this point Tim Hayes from OpenLink software writes:

```
Using another database as an intermediary is not a good idea, when you can use ODBC from PHP straight to your database - i.e. with OpenLink's drivers. If you do need to use an intermediary file format, OpenLink have now released Virtuoso (a virtual database engine) for NT, Linux and other unix platforms. Please visit our website for a free download.
```

One option that has proven successful is to use MySQL and its MyODBC drivers on Windows and synchronizing the databases. Steve Lawrence writes:

- Install MySQL on your platform according to instructions with MySQL. Latest available from [www.mysql.com](http://www.mysql.com) (get it from your mirror!). No special configuration required except when you set up a database, and configure the user account, you should put % in the host field, or the host name of the Windows computer you wish to access MySQL with. Make a note of your server name, username, and password.
- Download the MyODBC for Windows driver from the MySQL site. Latest release is myodbc-2\_50\_19-win95.zip (NT available too, as well as source code). Install it on your Windows machine. You can test the operation with the utilities included with this program.
- Create a user or system dsn in your ODBC administrator, located in the control panel. Make up a dsn name, enter your hostname, user name, password, port, etc for you MySQL database configured in step 1.
- Install Access with a full install, this makes sure you get the proper add-ins.. at the least you will need ODBC support and the linked table manager.
- Now the fun part! Create a new access database. In the table window right click and select Link Tables, or under the file menu option, select Get External Data and then Link Tables. When the file browser box comes up, select files of type: ODBC. Select System dsn and the name of your dsn created in step 3. Select the table to link, press ok, and presto! You can now open the table and add/delete/edit data on your MySQL server! You can also build queries, import/export tables to MySQL, build forms and reports, etc.

Tips and Tricks:

- You can construct your tables in access and export them to MySQL, then link them back in. That makes table creation quick.
- When creating tables in access, you must have a primary key defined in order to have write access to the table in access. Make sure you create a primary key in MySQL before linking in access
- If you change a table in MySQL, you have to re-link it in Access. Go to tools>add-ins>linked table manager, cruise to your ODBC DSN, and select the table to re-link from there. you can also move your dsn source around there, just hit the always prompt for new location checkbox before pressing ok.

### **3. I upgraded to PHP 4, and now mysql keeps telling me "Warning: MySQL: Unable to save result set in ...". What's up?**

Most likely what has happened is, PHP 4 was compiled with the '--with-mysql' option, without specifying the path to MySQL. This means PHP is using its built-in MySQL client library. If your system is running applications, such as PHP 3 as a concurrent Apache module, or auth-mysql, that use other versions of MySQL clients, then there is a conflict between the two differing versions of those clients.

Recompiling PHP 4, and adding the path to MySQL to the flag, '--with-mysql=/your/path/to/mysql' usually solves the problem.

### **4. After installing shared MySQL support, Apache dumps core as soon as libphp4.so is loaded. Can this be fixed?**

If your MySQL libs are linked against pthreads this will happen. Check using ldd. If they are, grab the MySQL tarball and compile from source, or recompile from the source rpm and remove the switch in the spec file that turns on the threaded client code. Either of these suggestions will fix this. Then recompile PHP with the new MySQL libs.

### **5. Why do I get an error that looks something like this: "Warning: 0 is not a MySQL result index in <file> on line <x>" or "Warning: Supplied argument is not a valid MySQL result resource in <file> on line <x>"?**

You are trying to use a result identifier that is 0. The 0 indicates that your query failed for some reason. You need to check for errors after submitting a query and before you attempt to use the returned result identifier. The proper way to do this is with code similar to the following:

```
$result = mysql_query("SELECT * FROM tables_priv");
if (!$result) {
    echo mysql_error();
    exit;
}
```

or

```
$result = mysql_query("SELECT * FROM tables_priv")
    or die("Bad query: ".mysql_error());
```

---

[Anterior](#)  
Obtaining PHP

[Inicio](#)  
[Subir](#)

[Siguiete](#)  
Installation

# Capítulo 30. Installation

This section holds common questions about the way to install PHP. PHP is available for almost any OS (except maybe for MacOS before OSX), and almost any web server.

To install PHP, follow the instructions in the [INSTALL](#) file located in the distribution. Windows users should also read the [install.txt](#) file. There are also some helpful hints for Windows users here.

1. [Unix/Windows: Where should my php.ini file be located?](#)
2. [Unix: I installed PHP, but every time I load a document, I get the message 'Document Contains No Data!' What's going on here?](#)
3. [Unix: I installed PHP using RPMS, but Apache isn't processing the PHP pages! What's going on here?](#)
4. [Unix: I installed PHP 3 using RPMS, but it doesn't compile with the database support I need! What's going on here?](#)
5. [Unix: I patched Apache with the FrontPage extensions patch, and suddenly PHP stopped working. Is PHP incompatible with the Apache FrontPage extensions?](#)
6. [Unix/Windows: I have installed PHP, but when I try to access a PHP script file via my browser, I get a blank screen.](#)
7. [Unix/Windows: I have installed PHP, but when try to access a PHP script file via my browser, I get a server 500 error.](#)
8. [Windows: I have installed PHP, but when I to access a PHP script file via my browser, I get the error:](#)

```
cgi error:
The specified CGI application misbehaved by not
returning a complete set of HTTP headers.
The headers it did return are:
```

9. [Windows: I've followed all the instructions, but still can't get PHP and IIS to work together!](#)

## 1. Unix/Windows: Where should my php.ini file be located?

By default on UNIX it should be in `/usr/local/lib`. Most people will want to change this at compile-time with the `--with-config-file-path` flag. You would, for example, set it with something like:

```
--with-config-file-path=/etc
```

And then you would copy `php.ini-dist` from the distribution to `/etc/php.ini` and edit it to make any local changes you want.

On Windows the default path for the `php.ini` file is the Windows directory.

## **2. Unix: I installed PHP, but every time I load a document, I get the message 'Document Contains No Data'! What's going on here?**

This probably means that PHP is having some sort of problem and is core-dumping. Look in your server error log to see if this is the case, and then try to reproduce the problem with a small test case. If you know how to use 'gdb', it is very helpful when you can provide a backtrace with your bug report to help the developers pinpoint the problem. If you are using PHP as an Apache module try something like:

- Stop your httpd processes
- `gdb httpd`
- Stop your httpd processes
- `> run -X -f /path/to/httpd.conf`
- Then fetch the URL causing the problem with your browser
- `> run -X -f /path/to/httpd.conf`
- If you are getting a core dump, gdb should inform you of this now
- `type: bt`
- You should include your backtrace in your bug report. This should be submitted to <http://bugs.php.net/>

If your script uses the regular expression functions ([ereg\(\)](#) and friends), you should make sure that you compiled PHP and Apache with the same regular expression package. This should happen automatically with PHP and Apache 1.3.x

## **3. Unix: I installed PHP using RPMS, but Apache isn't processing the PHP pages! What's going on here?**

Assuming you installed both Apache and PHP from RPM packages, you need to uncomment or add some or all of the following lines in your `http.conf` file:



```

# Extra Modules
AddModule mod_php.c
AddModule mod_php3.c
AddModule mod_perl.c

# Extra Modules
LoadModule php_module          modules/mod_php.so
LoadModule php3_module         modules/libphp3.so      /* for PHP 3 */
LoadModule php4_module         modules/libphp4.so      /* for PHP 4 */
LoadModule perl_module         modules/libperl.so

```

And add:

```

AddType application/x-httpd-php3 .php3      /* for PHP 3 */
AddType application/x-httpd-php .php        /* for PHP 4 */

```

... to the global properties, or to the properties of the VirtualDomain you want to have PHP support added to.

#### **4. Unix: I installed PHP 3 using RPMS, but it doesn't compile with the database support I need! What's going on here?**

Due to the way PHP 3 built, it is not easy to build a complete flexible PHP RPM. This issue is addressed in PHP 4. For PHP 3, we currently suggest you use the mechanism described in the INSTALL.REDHAT file in the PHP distribution. If you insist on using an RPM version of PHP 3, read on...

The RPM packagers are setting up the RPMS to install without database support to simplify installations *and* because RPMS use /usr/ instead of the standard /usr/local/ directory for files. You need to tell the RPM spec file which databases to support and the location of the top-level of your database server.

This example will explain the process of adding support for the popular MySQL database server, using the mod installation for Apache.

Of course all of this information can be adjusted for any database server that PHP supports. We will assume you installed MySQL and Apache completely with RPMS for this example as well.

- First remove mod\_php3 :

```
rpm -e mod_php3
```

- Then get the source rpm and INSTALL it, NOT --rebuild

```
rpm -Uvh mod_php3-3.0.5-2.src.rpm
```

- Then edit the `/usr/src/redhat/SPECS/mod_php3.spec` file

In the `%build` section add the database support you want, and the path.

For MySQL you would add

```
--with-mysql=/usr \
```

The `%build` section will look something like this:

```
./configure --prefix=/usr \  
--with-apxs=/usr/sbin/apxs \  
--with-config-file-path=/usr/lib \  
--enable-debug=no \  
--enable-safe-mode \  
--with-exec-dir=/usr/bin \  
--with-mysql=/usr \  
--with-system-regex
```

- Once this modification is made then build the binary rpm as follows:

```
rpm -bb /usr/src/redhat/SPECS/mod_php3.spec
```

- Then install the rpm

```
rpm -ivh /usr/src/redhat/RPMS/i386/mod_php3-3.0.5-2.i386.rpm
```

Make sure you restart Apache, and you now have PHP 3 with MySQL support using RPM's. Note that it is probably much easier to just build from the distribution tarball of PHP 3 and follow the instructions in `INSTALL.REDHAT` found in that distribution.

**5. Unix: I patched Apache with the FrontPage extensions patch, and suddenly PHP stopped working. Is PHP incompatible with the Apache FrontPage extensions?**

No, PHP works fine with the FrontPage extensions. The problem is that the FrontPage patch modifies several Apache structures, that PHP relies on. Recompiling PHP (using 'make clean ; make') after the FP patch is applied would solve the problem.

**6. Unix/Windows: I have installed PHP, but when I try to access a PHP script file via my browser, I get a blank screen.**

Do a 'view source' in the web browser and you will probably find that you can see the source code of your PHP script. This means that the web server did not send the script to PHP for interpretation. Something is wrong with the server configuration - double check the server configuration against the PHP installation instructions.

**7. Unix/Windows: I have installed PHP, but when try to access a PHP script file via my browser, I get a server 500 error.**

Something went wrong when the server tried to run PHP. To get to see a sensible error message, from the command line, change to the directory containing the PHP executable (`php.exe` on Windows) and run `php -i`. If PHP has any problems running, then a suitable error message will be displayed which will give you a clue as to what needs to be done next. If you get a screen full of html codes (the output of the [phpinfo\(\)](#) function) then PHP is working, and your problem may be related to your server configuration which you should double check.

**8. Windows: I have installed PHP, but when I to access a PHP script file via my browser, I get the error:**

```
cgi error:  
The specified CGI application misbehaved by not  
returning a complete set of HTTP headers.  
The headers it did return are:
```

This error message means that PHP failed to output anything at all. To get to see a sensible error message, from the command line, change to the directory containing the PHP executable (`php.exe` on Windows) and run `php -i`. If PHP has any problems running, then a suitable error message will be displayed which will give you a clue as to what needs to be done next. If you get a screen full of html codes (the output of the [phpinfo\(\)](#) function) then PHP is working.

Once PHP is working at the command line, try accessing the script via the browser again. If it still fails then it could be one of the following:

- File permissions on your PHP script, `php.exe`, `php4ts.dll`, `php.ini` or any PHP extensions you are trying to load are such that the anonymous internet user `ISUR_<machinename>` cannot access them.
- The script file does not exist (or possibly isn't where you think it is relative to your web root directory). Note that for IIS you can trap this error by ticking the 'check file exists' box when setting up the script mappings in the Internet Services Manager. If a script file does not exist then the server will return a 404 error instead. There is also the additional benefit that IIS will do any authentication required for you based on the NTLM permissions on your script file.

## **9. Windows: I've followed all the instructions, but still can't get PHP and IIS to work together!**

Make sure any user who needs to run a PHP script has the rights to run `php.exe`! IIS uses an anonymous user which is added at the time IIS is installed. This user needs rights to `php.exe`. Also, any authenticated user will also need rights to execute `php.exe`. And for IIS4 you need to tell it that PHP is a script engine.

---

[Anterior](#)

Database issues

[Inicio](#)

[Subir](#)

[Siguiete](#)

Build Problems

# Capítulo 31. Build Problems

This section gathers most common errors that occur at build time.

- [1. I got the latest version of PHP using the anonymous CVS service, but there's no configure script!](#)
- [2. I'm having problems configuring PHP to work with Apache. It says it can't find httpd.h, but it's right where I said it is!](#)
- [3. When I run configure, it says that it can't find the include files or library for GD, gdbm, or some other package!](#)
- [4. When it is compiling the file language-parser.tab.c, it gives me errors that say 'ytname undeclared'.](#)
- [5. When I run 'make', it seems to run fine but then fails when it tries to link the final application complaining that it can't find some files.](#)
- [6. When linking PHP, it complains about a number of undefined references.](#)
- [7. I can't figure out how to build PHP with Apache 1.3.](#)
- [8. I have followed all the steps to install the Apache module version on UNIX, and my PHP scripts show up in my browser or I am being asked to save the file.](#)
- [9. It says to use: --activate-module=src/modules/php4/libphp4.a, but that file doesn't exist, so I changed it to --activate-module=src/modules/php4/libmodphp4.a and it doesn't work!/? What's going on?](#)
- [10. When I try to build Apache with PHP as a static module using --activate-module=src/modules/php4/libphp4.a it tells me that my compiler is not ANSI compliant.](#)
- [11. When I try to build PHP using --with-apxs I get strange error messages.](#)
- [12. During 'make', I get errors in microtime, and a lot of 'RUSAGE\\_' stuff.](#)

## 1. I got the latest version of PHP using the anonymous CVS service, but there's no configure script!

You have to have the GNU autoconf package installed so you can generate the configure script from configure.in. Just run `./buildconf` in the top-level directory after getting the sources from the CVS server. (Also, unless you run configure with the `--enable-maintainer-mode` option, the configure script will not automatically get rebuilt when the configure.in file is updated, so you should make sure to do that manually when you notice configure.in has changed. One symptom of this is finding things like `@VARIABLE@` in your Makefile after configure or config.status is run.)

## 2. I'm having problems configuring PHP to work with Apache. It says it can't find httpd.h, but it's right where I said it is!

You need to tell the configure/setup script the location of the top-level of your Apache source tree. This means that you want to specify `--with-apache=/path/to/apache` and *not* `--with-apache=/path/to/apache/src`.

## 3. When I run configure, it says that it can't find the include files or library for GD, gdbm, or some other package!

You can make the configure script looks for header files and libraries in non-standard locations by specifying additional flags to pass to the C preprocessor and linker, such as:

```
CPPFLAGS=-I/path/to/include LDFLAGS=-L/path/to/library ./configure
```

If you're using a csh-variant for your login shell (why?), it would be:

```
env CPPFLAGS=-I/path/to/include LDFLAGS=-L/path/to/library ./configure
```

#### 4. When it is compiling the file `language-parser.tab.c`, it gives me errors that say 'yytname undeclared'.

You need to update your version of Bison. You can find the latest version at <ftp://ftp.gnu.org/pub/gnu/bison/>.

#### 5. When I run 'make', it seems to run fine but then fails when it tries to link the final application complaining that it can't find some files.

Some old versions of make that don't correctly put the compiled versions of the files in the functions directory into that same directory. Try running "`cp *.o functions`" and then re-running 'make' to see if that helps. If it does, you should really upgrade to a recent version of GNU make.

#### 6. When linking PHP, it complains about a number of undefined references.

Take a look at the link line and make sure that all of the appropriate libraries are being included at the end. Common ones that you might have missed are '-ldl' and any libraries required for any database support you included.

If you're linking with Apache 1.2.x, did you remember to add the appropriate information to the EXTRA\_LIBS line of the Configuration file and re-rerun Apache's Configure script? See the [INSTALL](#) file that comes with the distribution for more information.

Some people have also reported that they had to add '-ldl' immediately following 'libphp3.a' when linking with Apache.

#### 7. I can't figure out how to build PHP with Apache 1.3.

This is actually quite easy. Follow these steps carefully:

- Grab the latest Apache 1.3 distribution from <http://www.apache.org/dist/>.
- Ungzip and untar it somewhere, for example `/usr/local/src/apache-1.3`.
- Compile PHP by first running `./configure --with-apache=<path>/apache-1.3` (substitute <path> for the actual path to your apache-1.3 directory).
- Type 'make' followed by 'make install' to build PHP and copy the necessary files to the Apache distribution tree.
- Change directories into to your `/<path>/apache-1.3/src` directory and edit the Configuration file. At the end of the file, add: `AddModule modules/php3/libphp3.a`.
- Type '`./Configure`' followed by 'make'.
- You should now have a PHP-enabled httpd binary!

*Note:* : You can also use the new Apache `./configure` script. See the instructions in the `README.configure` file which is part of your Apache distribution. Also have a look at the `INSTALL` file in the PHP distribution.

#### 8. I have followed all the steps to install the Apache module version on UNIX, and my PHP scripts show up in my browser or I am being asked to save the file.

This means that the PHP module is not getting invoked for some reason. Three things to check before asking for further help:

- Make sure that the httpd binary you are running is the actual new httpd binary you just built. To do this, try running:  
`/path/to/binary/httpd -l`

If you don't see `mod_php3.c` listed then you are not running the right binary. Find and install the correct binary.

- Make sure you have added the correct Mime Type to one of your Apache `.conf` files. It should be: `AddType application/x-httpd-php3 .php3` (for PHP 3)

or `AddType application/x-httpd-php .php` (for PHP 4)

Also make sure that this AddType line is not hidden away inside a <Virtualhost> or <Directory> block which would prevent it from applying to the location of your test script.

- Finally, the default location of the Apache configuration files changed between Apache 1.2 and Apache 1.3. You should check to make sure that the configuration file you are adding the AddType line to is actually being read. You can put an obvious syntax error into your httpd.conf file or some other obvious change that will tell you if the file is being read correctly.

**9. It says to use: `--activate-module=src/modules/php4/libphp4.a`, but that file doesn't exist, so I changed it to `--activate-module=src/modules/php4/libmodphp4.a` and it doesn't work!? What's going on?**

Well, you decided to try to outsmart the people who wrote those nice step-by-step instructions for you and you have now discovered that these people cannot be outsmarted. The libphp4.a file is not supposed to exist. The Apache build process will create it.

**10. When I try to build Apache with PHP as a static module using `--activate-module=src/modules/php4/libphp4.a` it tells me that my compiler is not ANSI compliant.**

This is a misleading error message from Apache that has been fixed in more recent versions.

**11. When I try to build PHP using `--with-apxs` I get strange error messages.**

There are three things to check here. First, for some reason when Apache builds the apxs Perl script, it sometimes ends up getting built without the proper compiler and flags variables. Edit your apxs (sometimes found in /usr/local/apache/bin/apxs or /usr/sbin/apxs) and check for these lines:

```
my $CFG_CFLAGS_SHLIB = ' ';          # substituted via Makefile.tmpl
my $CFG_LD_SHLIB     = ' ';          # substituted via Makefile.tmpl
my $CFG_LDFLAGS_SHLIB = ' ';        # substituted via Makefile.tmpl
```

If this is what you see, you have found your problem. Change these lines to say:

```
my $CFG_CFLAGS_SHLIB = '-fpic -DSHARED_MODULE';      # substituted via Makefile.tmpl
my $CFG_LD_SHLIB     = 'gcc';                        # substituted via Makefile.tmpl
my $CFG_LDFLAGS_SHLIB = q(-shared);# substituted via Makefile.tmpl
```

The second possible problem should only be an issue on RedHat-6.1/6.2. The apxs script RedHat ships is broken. Look for this line:

```
my $CFG_LIBEXECDIR = 'modules';          # substituted via APACI install
```

If you see the above line, change it to this:

```
my $CFG_LIBEXECDIR = '/usr/lib/apache';      # substituted via APACI install
```

Last, if you reconfigure/reinstall Apache, add a 'make clean' to the process after './configure' and before 'make'.

**12. During 'make', I get errors in microtime, and a lot of 'RUSAGE\_' stuff.**

During the 'make' portion of installation, if you encounter problems that look similar to this:

```
microtime.c: In function `php_if_getrusage':
microtime.c:94: storage size of `usg' isn't known
microtime.c:97: `RUSAGE_SELF' undeclared (first use in this function)
microtime.c:97: (Each undeclared identifier is reported only once
microtime.c:97: for each function it appears in.)
microtime.c:103: `RUSAGE_CHILDREN' undeclared (first use in this function)
make[3]: *** [microtime.lo] Error 1
make[3]: Leaving directory `/home/master/php-4.0.1/ext/standard'
make[2]: *** [all-recursive] Error 1
make[2]: Leaving directory `/home/master/php-4.0.1/ext/standard'
make[1]: *** [all-recursive] Error 1
make[1]: Leaving directory `/home/master/php-4.0.1/ext'
make: *** [all-recursive] Error 1
```

Your system is broken. You need to fix your /usr/include files either by making sure your /usr/include/linux symlink is pointing to the right place in your kernel sources or by installing a glibc-devel package that matches your glibc. This has absolutely nothing to do with PHP. To prove this to yourself, try this simple test:

```
$ cat >test.c <<X
#include <sys/resource.h>
X
$ gcc -E test.c >/dev/null
```

If that spews out errors, you know your include files are messed up.

---

[Anterior](#)  
Installation

[Inicio](#)  
[Subir](#)

[Siguiete](#)  
Using PHP



---

# Capítulo 32. Using PHP

This section gathers most common errors you can face, while writing PHP scripts.

1. [I would like to write a generic PHP script that can handle data coming from any form. How do I know which POST method variables are available?](#)
2. [I need to convert all single-quotes \('\) to a backslash followed by a single-quote. How can I do this with a regular expression?](#)
3. [When I do the following, the output is printed in the wrong order:](#)

```
function myfunc($argument)
{
    echo $argument + 10;
}
$variable = 10;
echo "myfunc($variable) = " . myfunc($variable);
```

[what's going on?](#)

4. [Hey, what happened to my newlines?](#)

```
<pre>
<?php echo "This should be the first line."; ?>
<?php echo "This should show up after the new line above."; ?>
</pre>
```

5. [I need to access information in the request header directly. How can I do this?](#)
6. [When I try to use authentication with IIS I get 'No Input file specified'.](#)
7. [My PHP script works on IE and Lynx, but on Netscape some of my output is missing. When I do a "View Source" I see the content in IE but not in Netscape.](#)
8. [How am I supposed to mix XML and PHP? It complains about my <?xml> tags!](#)
9. [How can I use PHP with FrontPage or some other HTML editor that insists on moving my code around?](#)

10. [Where can I find a complete list of pre-set variables available to me, and why are these not documented in the PHP documentation?](#)
11. [I'm trying to access one of the standard CGI variables \(such as \\$DOCUMENT\\_ROOT or \\$HTTP\\_REFERER\) in a user-defined function, and it can't seem to find it. What's wrong?](#)

**1. I would like to write a generic PHP script that can handle data coming from any form. How do I know which POST method variables are available?**

Make sure that the [track\\_vars](#) feature is enabled in your `php.ini` file. Since PHP 4.0.3, this feature is always on. When `track_vars` is on, it creates some associative arrays, the most important here is: `$HTTP_POST_VARS`. So, to write a generic script to handle POST method variables you would need something similar to the following:

```
foreach ($HTTP_POST_VARS as $var => $value) {  
    echo "$var = $value<br>\n";  
}
```

**2. I need to convert all single-quotes (') to a backslash followed by a single-quote. How can I do this with a regular expression?**

First off, take a look at the [addslashes\(\)](#) function. It will do exactly what you want. You should also have a look at the [magic\\_quotes\\_gpc](#) directive in your `php.ini` file.

**3. When I do the following, the output is printed in the wrong order:**

```
function myfunc($argument)  
{  
    echo $argument + 10;  
}  
$variable = 10;  
echo "myfunc($variable) = " . myfunc($variable);
```

**what's going on?**

To be able to use the results of your function in an expression (such as concatenating it with

other strings in the example above), you need to *return* the value, not [echo\(\)](#) it.

#### 4. Hey, what happened to my newlines?

```
<pre>
<?php echo "This should be the first line."; ?>
<?php echo "This should show up after the new line above."; ?>
</pre>
```

In PHP, the ending for a block of code is either "?>" or "?>\n" (where \n means a newline). So in the example above, the echoed sentences will be on one line, because PHP omits the newlines after the block ending. This means that you need to insert an extra newline after each block of PHP code to make it print out one newline.

Why does PHP do this? Because when formatting normal HTML, this usually makes your life easier because you don't want that newline, but you'd have to create extremely long lines or otherwise make the raw page source unreadable to achieve that effect.

#### 5. I need to access information in the request header directly. How can I do this?

The [getallheaders\(\)](#) function will do this if you are running PHP as an Apache module. So, the following bit of code will show you all the request headers:

```
$headers = getallheaders();
foreach ($headers as $name => $content) {
    echo "headers[$name] = $content<br>\n";
}
```

#### 6. When I try to use authentication with IIS I get 'No Input file specified'.

The security model of IIS is at fault here. This is a problem common to all CGI programs running under IIS. A workaround is to create a plain HTML file (not parsed by PHP) as the entry page into an authenticated directory. Then use a META tag to redirect to the PHP page,

or have a link to the PHP page. PHP will then recognize the authentication correctly. With the ISAPI module, this is not a problem. This should not effect other NT web servers. For more information, see: <http://support.microsoft.com/support/kb/articles/q160/4/22.asp>.

### **7. My PHP script works on IE and Lynx, but on Netscape some of my output is missing. When I do a "View Source" I see the content in IE but not in Netscape.**

Very good question! ;) This is a tricky little issue and it has come up twice in the past month as of this writing. Both times I ended up spending a good 20 minutes trying to figure out what the heck was going on. The answer is that both IE and Lynx ignore any NULs (\0) in the HTML stream. Netscape does not. The best way to check for this is to compile the command-line version of PHP (also known as the CGI version) and run your script from the command line and pipe it through 'od -c' and look for any \0 characters. (If you are on Windows you need to find an editor or some other program that lets you look at binary files) When Netscape sees a NUL in a file it will typically not output anything else on that line whereas both IE and Lynx will. If this issue has bitten you, congratulations! You are not alone.

### **8. How am I supposed to mix XML and PHP? It complains about my <?xml> tags!**

You need to turn off the short tags by setting [short\\_tags](#) to 0 in your `php.ini` file, or by using the appropriate Apache directive. You could even use a `<File>` section to do this selectively.

### **9. How can I use PHP with FrontPage or some other HTML editor that insists on moving my code around?**

One of the easiest things to do is to enable using ASP tags in your PHP code. This allows you to use the ASP-style `<%` and `%>` code delimiters. Some of the popular HTML editors handle those more intelligently (for now). To enable the ASP-style tags, you need to set the [asp\\_tags](#) `php.ini` variable, or use the appropriate Apache directive.

### **10. Where can I find a complete list of pre-set variables available to me, and why are these not documented in the PHP documentation?**

The best way is to stick a `<?php phpinfo(); ?>` part on a page and load it up. This will show you all sorts of information about your PHP setup, including a list of both environment variables and also special variables set by your web server. This list can't really be documented in the PHP documentation because it will change from one server to another.

**11. I'm trying to access one of the standard CGI variables (such as \$DOCUMENT\_ROOT or \$HTTP\_REFERER) in a user-defined function, and it can't seem to find it. What's wrong?**

Environment variables are normal global variables, so you must either declare them as global variables in your function (by using "global \$DOCUMENT\_ROOT;", for example) or by using the global variable array (ie, "\$GLOBALS[ "DOCUMENT\_ROOT" ]").

---

[Anterior](#)

Build Problems

[Inicio](#)

[Subir](#)

[Siguiete](#)

PHP and HTML

# Capítulo 33. PHP and HTML

PHP and HTML interact a lot: PHP generate HTML, and HTML has informations that will be sent to PHP.

1. [What encodings/decodings do I need when I pass a value on via a form? And via an URL?](#)
2. [I'm trying to use an <input type="image"> tag, but the \\$foo.x and \\$foo.y variables aren't available. Where are they?](#)
3. [How do I create arrays in a HTML <form>?](#)
4. [How do I get all the results from a select multiple HTML tag?](#)

## 1. What encodings/decodings do I need when I pass a value on via a form? And via an URL?

There are several stages for which encoding is important. Assuming that you have a [string](#) `$data`, which contains the string you want to pass on in a non-encoded way, these are the relevant stages:

- HTML interpretation. In order to specify a random string, you *must* include it in double quotes, and `htmlspecialchars` the the whole value.
- URL: A URL consists of several parts. If you want your data to be interpreted as one item, you *must* encode it with [urlencode\(\)](#).

### Ejemplo 33-1. A hidden HTML form element

```
<?php
    echo "<input type=hidden value=\"\" . htmlspecialchars($data) . \">\n";
?>
```

**Nota:** It is wrong to [urlencode\(\)](#) `$data`, because it's the browsers repositability to [urlencode\(\)](#) the data. All popular browsers do that correctly. Note that this will happen regardless of the method (i.e., GET or POST). You'll only notice this in case of GET request though, because POST requests are usually hidden.

### Ejemplo 33-2. Data to be edited by the user

```
<?php
    echo "<textarea name=mydata>\n" ;
    echo htmlspecialchars($data) . "\n" ;
    echo "</textarea>" ;
?>
```

**Nota:** The data is shown in the browser as intended, because the browser will interpret the html escaped symbols.

Upon submitting, either via GET or POST, the data will be urlencoded by the browser for transferring, and directly urldecoded by PHP. So in the end, you don't need to do any urlencoding/urldecoding yourself, everything is handled automatically.

### Ejemplo 33-3. In an URL

```
<?php
    echo "<a href=\"\" . htmlspecialchars( "/nepage.php?stage=23&data=" .
        urlencode($data) ) . "\">\n" ;
?>
```

**Nota:** In fact you are faking a HTML GET request, therefore it's necessary to manually [urlencode\(\)](#) the data.

**Nota:** You need to **htmlspecialchars()** the whole URL, because the URL occurs as value of an HTML-attribute. In this case, the browser will first un-**htmlspecialchars()** the value, and then pass the URL on. PHP will understand the URL correctly, because you **urlencoded()** the data.

You'll notice that the & in the URL is replaced by `&amp;`. Although most browsers will recover if you forget this, this isn't always possible. So even if your URL is not dynamic, you *need* to **htmlspecialchars()** the URL.

### 2. I'm trying to use an `<input type="image">` tag, but the `$foo.x` and `$foo.y` variables aren't available. Where are they?

When submitting a form, it is possible to use an image instead of the standard submit button with a tag

like:

```
<input type="image" src="image.gif" name="foo" >
```

When the user clicks somewhere on the image, the accompanying form will be transmitted to the server with two additional variables: foo.x and foo.y.

Because \$foo.x and \$foo.y are invalid variable names in PHP, they are automatically converted to \$foo\_x and \$foo\_y. That is, the periods are replaced with underscores.

### 3. How do I create arrays in a HTML <form>?

To get your <form> result sent as an [array](#) to your PHP script you name the <input>, <select> or <textarea> elements like this:

```
<input name="MyArray[ ]" >  
<input name="MyArray[ ]" >  
<input name="MyArray[ ]" >  
<input name="MyArray[ ]" >
```

Notice the square brackets after the variable name, that's what makes it an array. You can group the elements into different arrays by assigning the same name to different elements:

```
<input name="MyArray[ ]" >  
<input name="MyArray[ ]" >  
<input name="MyOtherArray[ ]" >  
<input name="MyOtherArray[ ]" >
```

This produces two arrays, MyArray and MyOtherArray, that gets sent to the PHP script. It's also possible to assign specific keys to your arrays:

```
<input name="AnotherArray[ ]" >  
<input name="AnotherArray[ ]" >  
<input name="AnotherArray[email]" >  
<input name="AnotherArray[phone]" >
```

The AnotherArray array will now contain the keys 0, 1, email and phone.



**Nota:** Specifying an arrays key is optional in HTML. If you do not specify the keys, the array gets filled in the order the elements appear in the form. Our first example will contain keys 0, 1, 2 and 3.

See also [Array Functions](#) and [Variables from outside PHP](#).

#### 4. How do I get all the results from a select multiple HTML tag?

The select multiple tag in an HTML construct allows users to select multiple items from a list. These items are then passed to the action handler for the form. The problem is that they are all passed with the same widget name. ie.

```
<select name="var" multiple>
```

Each selected option will arrive at the action handler as:

```
var=option1  
var=option2  
var=option3
```

Each option will overwrite the contents of the previous `$var` variable. The solution is to use PHP's "array from form element" feature. The following should be used:

```
<select name="var[]" multiple>
```

This tells PHP to treat `$var` as an array and each assignment of a value to `var[]` adds an item to the array. The first item becomes `$var[0]`, the next `$var[1]`, etc. The [count\(\)](#) function can be used to determine how many options were selected, and the [sort\(\)](#) function can be used to sort the option array if necessary.

Note that if you are using JavaScript the `[]` on the element name might cause you problems when you try to refer to the element by name. Use it's numerical form element id instead, or enclose the variable name in single quotes and use that as the index to the elements array, for example:

```
variable = documents.forms[0].elements['var[]'];
```

Using PHP

[Subir](#)

PHP and COM

# Capítulo 34. PHP and COM

PHP can be used to access COM and DCOM objects on Win32 platforms.

1. [I have built a DLL to calculate something. Is there any way to run this DLL under PHP ?](#)
2. [What does 'Unsupported variant type: xxxx \(0xxxxx\)' mean ?](#)
3. [Is it possible manipulate visual objects in PHP ?](#)
4. [Can I store a COM object in a session ?](#)
5. [How can I trap COM errors ?](#)
6. [Can I generate DLL files from PHP scripts like i can in Perl ?](#)
7. [What does 'Unable to obtain IDispatch interface for CLSID {xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx}' mean ?](#)
8. [How can I run COM object from remote server ?](#)
9. [I get 'DCOM is disabled in C:\path...\scriptname.php on line 6', what can I do ?](#)
10. [Is it possible to load/manipulate an ActiveX object in a page with PHP ?](#)
11. [Is it possible to get a running instance of a component ?](#)
12. [Is there a way to handle an event sent from COM object ?](#)
13. [I'm having problems when trying to invoke a method of a COM object wich exposes more than one interface. What can I do ?](#)
14. [So PHP works with COM, how about COM+ ?](#)
15. [If PHP can manipulate COM objects, can we imagine to use MTS to manage components resources, in conjunction with PHP ?](#)

## **1. I have built a DLL to calculate something. Is there any way to run this DLL under PHP ?**

If this is a simple DLL there is no way yet to run it from PHP. If the DLL contains a COM server you may be able to access it if it implements the IDispatch interface.

## **2. What does 'Unsupported variant type: xxxx (0xxxxx)' mean ?**

There are dozens of VARIANT types and combinations of them. Most of them are already supported but a few still have to be implemented. Arrays are not completely supported. Only

single dimensional indexed only arrays can be passed between PHP and COM. If you find other types that aren't supported, please report them as a bug (if not already reported) and provide as much information as available.

### **3. Is it possible manipulate visual objects in PHP ?**

Generally it is, but as PHP is mostly used as a web scripting language it runs in the web servers context, thus visual objects will never appear on the servers desktop. If you use PHP for application scripting e.g. in conjunction with PHP-GTK there is no limitation in accessing and manipulating visual objects through COM.

### **4. Can I store a COM object in a session ?**

No, you can't. COM instances are treated as resources and therefore they are only available in a single script's context.

### **5. How can I trap COM errors ?**

Currently it's not possible to trap COM errors beside the ways provided by PHP itself (@, track\_errors, ..), but we are thinking of a way to implement this.

### **6. Can I generate DLL files from PHP scripts like i can in Perl ?**

No, unfortunately there is no such tool available for PHP.

### **7. What does 'Unable to obtain IDispatch interface for CLSID {xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx}' mean ?**

This error can have multiple reasons:

- the CLSID is wrong
- the requested DLL is missing
- the requested component doesn't implement the IDispatch interface

### **8. How can I run COM object from remote server ?**

Exactly like you run local objects. You only have to pass the ip of the remote machine as second parameter to the COM constructor.

Make sure that you have set `com.allow_dcom=true` in your `php.ini`.

### **9. I get 'DCOM is disabled in C:\path...\scriptname.php on line 6', what can I do ?**

Edit your `php.ini` and set `com.allow_dcom=true`.

### **10. Is it possible to load/manipulate an ActiveX object in a page with PHP ?**

This has nothing to do with PHP. ActiveX objects are loaded on client side if they are requested by the HTML document. There is no relation to the PHP script and therefore there is no direct server side interaction possible.

### **11. Is it possible to get a running instance of a component ?**

This is possible with the help of monikers. If you want to get multiple references to the same word instance you can create that instance like shown:

```
$word = new COM("C:\docs\word.doc");
```

This will create a new instance if there is no running instance available or it will return a handle to the running instance, if available.

### **12. Is there a way to handle an event sent from COM object ?**

Not yet.

### **13. I'm having problems when trying to invoke a method of a COM object wich exposes more than one interface. What can I do ?**

The answer is as simple as unsatisfying. I don't know exactly but i think you can do nothing. If someone has specific information about this, please let [me](#) know :)

### **14. So PHP works with COM, how about COM+ ?**

COM+ extends COM by a framework for managing components through MTS and MSMQ but there is nothing special that PHP has to support to use such components.

**15. If PHP can manipulate COM objects, can we imagine to use MTS to manage components resources, in conjunction with PHP ?**

PHP itself doesn't handle transactions yet. Thus if an error occurs no rollback is initiated. If you use components that support transactions you will have to implement the transaction management yourself.

---

[Anterior](#)

PHP and HTML

[Inicio](#)

[Subir](#)

[Siguiete](#)

PHP and other languages

# Capítulo 35. PHP and other languages

PHP is the best language for web programming, but what about other languages?

1. [PHP vs. ASP?](#)
2. [Is there an ASP to PHP converter?](#)
3. [PHP vs. Cold Fusion?](#)
4. [PHP vs. Perl?](#)

## 1. PHP vs. ASP?

ASP is not really a language in itself, it's an acronym for Active Server Pages, the actual language used to program ASP with is Visual Basic Script or JScript. The biggest drawback of ASP is that it's a proprietary system that is natively used only on Microsoft Internet Information Server (IIS). This limits its availability to Win32 based servers. There are a couple of projects in the works that allows ASP to run in other environments and webservers: [InstantASP](#) from [Halcyon](#) (commercial), [Chili!Soft ASP](#) from [Chili!Soft](#) (commercial) and [OpenASP from ActiveScripting.org](#) (free). ASP is said to be a slower and more cumbersome language than PHP, less stable as well. Some of the pros of ASP is that since it primarily uses VBScript it's relatively easy to pick up the language if you're already know how to program in Visual Basic. ASP support is also enabled by default in the IIS server making it easy to get up and running. The components built in ASP are really limited, so if you need to use "advanced" features like interacting with FTP servers, you need to buy additional components.

## 2. Is there an ASP to PHP converter?

Yes, [asp2php](#) is the one most often referred to.

## 3. PHP vs. Cold Fusion?

PHP is commonly said to be faster and more efficient for complex programming tasks and trying out new ideas. PHP is generally referred to as more stable and less resource intensive as well. Cold Fusion has better error handling, database abstraction and date parsing although database abstraction is addressed in PHP 4. Another thing that is listed as one of Cold Fusion's strengths is its excellent search engine, but it has been mentioned that a search engine is not something that should be included in a web scripting language. PHP runs on almost every platform there is; Cold Fusion is only available on Win32, Solaris, Linux and HP/UX. Cold Fusion has a good IDE and is generally easier to get started with, whereas PHP initially requires more programming knowledge. Cold Fusion is designed with non-programmers in mind, while PHP is focused on programmers.

A great summary by Michael J Sheldon on this topic has been posted to the PHP mailing list. A copy can be found [here](#).

#### **4. PHP vs. Perl?**

The biggest advantage of PHP over Perl is that PHP was designed for scripting for the web where Perl was designed to do a lot more and can because of this get very complicated. The flexibility / complexity of Perl makes it easier to write code that another author / coder has a hard time reading. PHP has a less confusing and stricter format without losing flexibility. PHP is easier to integrate into existing HTML than Perl. PHP has pretty much all the 'good' functionality of Perl: constructs, syntax and so on, without making it as complicated as Perl can be. Perl is a very tried and true language, it's been around since the late eighties, but PHP is maturing very quickly.

---

[Anterior](#)

PHP and COM

[Inicio](#)

[Subir](#)

[Siguiete](#)

Migrating from PHP 2 to PHP



# Capítulo 36. Migrating from PHP 2 to PHP 3

PHP has already a long history behind him: Legendary PHP 1.0, PHP/FI, PHP 3.0 and PHP 4.0.

## 1. [Migrating from PHP 2 to PHP 3?](#)

### 1. Migrating from PHP 2 to PHP 3?

PHP/FI 2.0 is no longer supported. Please see [appropriate manual section](#) for information about migration from PHP/FI 2.0.

If you are still working with PHP 2, we *strongly* recommend you to upgrade straight to PHP 4.

# Capítulo 37. Migrating from PHP 3 to PHP 4

PHP has already a long history behind him : Legendary PHP 1.0, PHP/FI, PHP 3.0 and PHP 4.0.

1. [Migrating from PHP3 to PHP4](#)
2. [Incompatible functions?](#)

## 1. Migrating from PHP3 to PHP4

PHP 4 was designed to be as compatible with earlier versions of PHP as possible and very little functionality was broken in the process. If you're really unsure about compatibility you should install PHP 4 in a test environment and run your scripts there.

Also see the [appropriate migration appendix](#) of this manual.

## 2. Incompatible functions?

Since PHP 4 is basically a rewrite of the entire PHP engine there was very few functions that were altered and only then some of the more exotic ones.

# Capítulo 38. Miscellaneous Questions

There can be some questions we can't put into other categories. Here you can find them.

1. [Where did the pop-ups go on the website? Can I have the code for that?](#)

## 1. Where did the pop-ups go on the website? Can I have the code for that?

The yellow pop-up windows on the old site were pretty cool, but were very difficult to maintain (since some companies seem to enjoy changing the way their browsers work with every new release).

All the code for previous versions of the website is still available through CVS. Specifically, the last version of shared.inc (that had all the Javascript and DHTML to do the popups) is available [here](#).

## VII. Apéndices

### Tabla de contenidos

- A. [Migrando de PHP/FI 2.0 a PHP 3.0](#)
- B. [Migrating from PHP 3.0 to PHP 4.0](#)
- C. [Desarrollo en PHP](#)
- D. [El debugger de PHP](#)
- E. [PHP reserved words](#)
- F. [PHP Resource Types](#)
- G. [Aliases list](#)
- H. [Using PHP from the command line](#)

# Apéndice A. Migrando de PHP/FI 2.0 a PHP 3.0

## Tabla de contenidos

[Acerca de las incompatibilidades en PHP 3.0](#)

[Tags de inicio y fin](#)

[sintáxis de if..endif](#)

[sintáxis de while \(mientras\)](#)

[Tipos de expresiones](#)

[Cambios en los mensajes de error](#)

[Evaluación booleana por corto-circuito](#)

[Retorno de valores en funciones verdadero/falso](#)

[Otras incompatibilidades](#)

## Acerca de las incompatibilidades en PHP 3.0

PHP 3.0 ha sido reescrito desde cero. Tiene un intérprete propio, que es mucho mas robusto y consistente que el de 2.0. 3.0 es también significativamente mas rápido, y usa menos memoria. De cualquier modo, algunas de estas mejoras no fueron posibles sin alterar la compatibilidad, tanto de sintaxis como de funcionamiento.

Además, los desarrolladores de PHP han intentado clarificar la sintaxis y la semántica de PHP 3.0, y eso ha causado también algunas incompatibilidades. Creemos que, a la larga, estos cambios serán para mejor.

Este capítulo, intentará guiarle a través de las incompatibilidades que encontrará si migra de PHP/FI 2.0 a PHP 3.0, además de ayudarle a resolverlas. No se mencionarán aquí las herramientas nuevas, hasta que sea necesario.

Existe un programa que convierte automáticamente sus viejos guiones PHP/FI 2.0. Puede hallarlo en el subdirectorio `convertor` de la distribución 3.0 de PHP Este programa, solo actúa en referencia a los cambios sintácticos, así que debería leer este capítulo detenidamente, de todos modos.

---

[Anterior](#)

Apéndices

[Inicio](#)

[Subir](#)

[Siguiete](#)

Tags de inicio y fin

# Apéndice B. Migrating from PHP 3.0 to PHP 4.0

## Tabla de contenidos

[What has changed in PHP 4.0](#)

[Running PHP 3 and PHP 4 concurrently](#)

[Migrating Configuration Files](#)

[Parser behavior](#)

[Error reporting](#)

[Initializers](#)

[empty\( "0" \)](#)

[Missing functions](#)

[PHP 3.0 extension](#)

[Variable substitution in strings](#)

[Cookies](#)

## What has changed in PHP 4.0

PHP 4.0 and the integrated Zend engine have greatly improved PHP's performance and capabilities, but great care has been taken to break as little existing code as possible. So migrating your code from PHP 3.0 to 4.0 should be much easier than migrating from PHP/FI 2.0 to PHP 3.0. A lot of existing PHP 3.0 code should be ready to run without changes, but you should still know about the few differences and take care to test your code before switching versions in production environments. The following should give you some hints about what to look for.

# Apéndice C. Desarrollo en PHP

## Tabla de contenidos

[Añadiendo funciones al PHP3](#)

[Llamando a Funciones del Usuario](#)

[Informando de errores](#)

## Añadiendo funciones al PHP3

### Prototipo de Función

Todas las funciones son como esta:

```
void php3_algo(INTERNAL_FUNCTION_PARAMETERS) {  
}
```

Incluso si su función no lleva argumentos, es así como se le llama.

### Argumentos de Función

Los argumentos son siempre de tipo pval. Este tipo contiene una unión que es el tipo actual del argumento. Así, si su función tiene dos argumentos, deberá hacer algo como lo que sigue al principio de la misma:

#### Ejemplo C-1. Extrayendo argumentos de función

```
pval *arg1, *arg2;  
if (ARG_COUNT(ht) != 2 || getParameters(ht, 2, &arg1, &arg2) == FAILURE) {  
    WRONG_PARAM_COUNT;  
}
```

NOTA: Los argumentos pueden pasarse tanto por valor como por referencia. En ambos casos, necesitará pasar `&(pval *)` a `getParameters`. Si desea comprobar si el *n*ésimo parámetro le ha sido enviado o no por referencia, puede utilizar la función `ParameterPassedByReference(ht, n)`. Esta devolverá 1 ó 0, según corresponda.

Cuando cambie alguno de los parámetros pasados, tanto si son enviados por referencia o por valor, puede volver a comenzar con éste llamando la función `pval_destructor` sobre el mismo, o, si es una `ARRAY` a la que quiere añadir algo, puede utilizar funciones similares a las incluidas en `internal_functions.h`, que manipulan el valor `return_value` como si fuera de tipo `ARRAY`.

Además, si cambia un parámetro a `IS_STRING`, asegúrese primero de asignar el valor y el tamaño a la cadena creada por `estrdup()` y sólo entonces cambiar su tipo a `IS_STRING`. Si modifica la cadena de un parámetro que ya es `IS_STRING` o `IS_ARRAY`, deberá primero aplicarle la función `pval_destructor`.



# Argumentos de Función Variables

Una función puede tomar un número variable de argumentos. Si su función puede tomar tanto 2 como 3 argumentos, utilice el siguiente código:

## Ejemplo C-2. Argumentos de función variables

```
pval *arg1, *arg2, *arg3;
int arg_count = ARG_COUNT(ht);

if (arg_count < 2 || arg_count > 3 ||
    getParameters(ht, arg_count, &arg1, &arg2, &arg3) == FAILURE) {
    WRONG_PARAM_COUNT;
}
```

# Usando los Argumentos de Función

El tipo de cada argumento se guarda en el campo type del pval. Este tipo puede ser:

**Tabla C-1. Tipos Internos de PHP**

IS_STRING	Cadena
IS_DOUBLE	Coma flotante de doble precisión
IS_LONG	Entero largo
IS_ARRAY	Matriz
IS_EMPTY	Nada
IS_USER_FUNCTION	??
IS_INTERNAL_FUNCTION	?? (N.D.: si alguno de estos no se puede pasar a una función, bórrese)
IS_CLASS	??
IS_OBJECT	??

Si obtiene un argumento de un tipo y desea utilizarlo como si fuera de otro, o si quiere forzar a que un argumento sea de un tipo determinado, puede usar una de las siguientes funciones de conversión:

```
convert_to_long(arg1);
convert_to_double(arg1);
convert_to_string(arg1);
convert_to_boolean_long(arg1); /* Si la cadena es "" o "0" pasa a ser 0; si no, vale 1 */
convert_string_to_number(arg1); /* Convierte la cadena a LONG o a DOUBLE, dependiendo de su contenido */
```

Estas funciones convierten el valor in-situ. No devuelven nada.

El argumento real es almacenado en una unión cuyos miembros son:

- IS\_STRING: arg1->value.str.val
- IS\_LONG: arg1->value.lval
- IS\_DOUBLE: arg1->value.dval

## Manejo de Memoria en las Funciones

La memoria necesitada por una función deberá ser asignada usando `emalloc()` o `estrdup()`. Estas son funciones abstractas de manejo de memoria que son similares a las funciones normales `malloc()` y `strdup()`. La memoria deberá liberarse con `efree()`.

Hay dos tipos de memoria en este programa: la memoria que se devuelve al troceador (parser) en una variable, y la memoria que se necesita para almacenamiento temporal de datos en sus funciones. Cuando asigne una cadena a una variable que se devolverá al troceador deberá asegurarse previamente de asignar la memoria con `emalloc()` o con `estrdup()`. Esta memoria NUNCA debe ser liberada por usted, salvo si más adelante, en la misma función, sobrescribe la asignación original (aunque este hábito de programación no es bueno).

Para cada trozo de memoria temporal/permanente que precise en sus funciones/librería deberá utilizar las funciones `emalloc()`, `estrdup()`, y `efree()`. Estas se comportan EXACTAMENTE como sus funciones equivalentes. Cualquier cosa que asigne con `emalloc()` o `estrdup()` deberá liberarla con `efree()` en uno u otro momento, salvo que se suponga que deba permanecer activa hasta el final del programa; de otro modo, se producirá una fuga de memoria. El significado de "estas se comportan exactamente como sus funciones equivalentes" es: si llama a `efree()` sobre algo que no ha sido asignado previamente con `emalloc()` o con `estrdup()`, puede provocar un fallo de segmentación. Por ello debe tener cuidado y liberar toda la memoria desperdiciada.

Si compila con "-DDEBUG", el PHP3 mostrará una lista de toda la memoria que fue asignada usando `emalloc()` y `estrdup()`, pero que nunca fue liberada con `efree()`, al terminar de ejecutar el guión especificado.

## Asignando Variables en la Tabla de Símbolos

Están disponibles una serie de macros que hacen más fácil el asignar una variable en la tabla de símbolos:

- SET\_VAR\_STRING(nombre,valor) [\[1\]](#)
- SET\_VAR\_DOUBLE(nombre,valor)
- SET\_VAR\_LONG(nombre,valor)

[\[1\]](#)

Las tablas de símbolos en PHP 3.0 se implementan como tablas hash (con extracto). En todo momento, `&symbol_table` es un puntero a la tabla de símbolos 'principal', mientras que `active_symbol_table` apunta a la tabla de símbolos activa (pueden ser idénticas, al principio de todo, o diferentes, si se está dentro de una función).

Los ejemplos siguientes utilizan 'active\_symbol\_table'. Deberá reemplazarla por `&symbol_table` si desea trabajar específicamente con la tabla de símbolos 'principal'. También se pueden aplicar las mismas funciones a matrices, como se explica más abajo.

### Ejemplo C-3. Comprobando si \$algo existe en una tabla de símbolos

```
if (hash_exists(active_symbol_table,"algo",sizeof("algo"))) { existe... }
else { no existe }
```

#### Ejemplo C-4. Hallando el tamaño de una variable en una tabla de símbolos

```
hash_find(active_symbol_table, "algo", sizeof("algo"), &valptr);
check(valptr.type);
```

Las matrices en PHP 3.0 se implementan utilizando las mismas tablas hash que para las tablas de símbolos. Ello quiere decir que las dos funciones anteriores se pueden usar también para comprobar variables dentro de matrices.

Si desea definir un nuevo símbolo de matriz en una tabla de símbolos, deberá hacer lo que sigue.

Primero, deberá comprobar si ya existe usando `hash_exists()` o `hash_find()` y abortar la ejecución de forma apropiada.

Luego inicialice la matriz:

#### Ejemplo C-5. Inicializando una nueva matriz

```
pval matriz;

if (array_init(&matriz) == FAILURE) { falló... };
hash_update(active_symbol_table, "algo", sizeof("algo"), &matriz, sizeof(pval), NULL);
```

Este código declara una nueva matriz, llamada `$algo`, en la tabla de símbolos activa. Esta matriz está vacía.

Ahora se muestra cómo añadirle elementos:

#### Ejemplo C-6. Añadir entradas a una nueva matriz

```
pval elemento;

elemento.type = IS_LONG;
elemento.value.lval = 5;

/* define $algo["bar"] = 5 */
hash_update(matriz.value.ht, "bar", sizeof("bar"), &elemento, sizeof(pval), NULL);

/* define $algo[7] = 5 */
hash_index_update(matriz.value.ht, 7, &elemento, sizeof(pval), NULL);

/* define el siguiente puesto libre en $algo[,
 * $algo[8], como 5 (funciona como en php2)
 */
hash_next_index_insert(matriz.value.ht, &elemento, sizeof(pval), NULL);
```

Si desea modificar un valor que ha insertado en una matriz asociativa, deberá primero extraerlo de ella. Para evitar esa sobrecarga, puede pasarle un puntero `pval **` a la función para insertar en una matriz asociativa, y será actualizada con la dirección `pval *` del elemento insertado dentro de la matriz. Si dicho valor es **NULL** (como en todos los ejemplos anteriores), el parámetro se ignora.

`hash_next_index_insert()` usa más o menos la misma lógica que `"$algo[] = bar;"` en el PHP 2.0.

Si está preparando una matriz como valor devuelto por una función, puede inicializar la misma como antes, haciendo:

```
if (array_init(return_value) == FAILURE) { falló...; }
```

... y luego añadiéndole valores con las funciones auxiliares:

```
add_next_index_long(return_value, val_long);  
add_next_index_double(return_value, val_double);  
add_next_index_string(return_value, estrdup(val_cadena));
```

Por supuesto, si la adición no se realiza justo después de inicializar la matriz, probablemente tenga que buscarla antes:

```
pval *matriz;  
  
if (hash_find(active_symbol_table, "algo", sizeof("algo"), (void **)&matriz) == FAILURE) {  
no se hayó... }  
else { usar matriz->value.ht... }
```

Nótese que `hash_find` recibe un puntero a un puntero a `pval`, y no un puntero a `pval`.

Casi cualquier función de matrices asociativas devuelve `SUCCESS` o `FAILURE` (excepto por `hash_exists()`, que devuelve un valor lógico de certeza).

## Devolviendo valores simples

Están disponibles varias macros para facilitar la devolución de valores de una función.

Todas las macros `RETURN_*` fijan el valor y retornan de la función:

- `RETURN`
- `RETURN_FALSE`
- `RETURN_TRUE`
- `RETURN_LONG(l)`
- `RETURN_STRING(s,dup)` Si `dup` es **TRUE**, duplica la cadena
- `RETURN_STRINGL(s,l,dup)` Devuelve la cadena (s) especificando el largo (l).
- `RETURN_DOUBLE(d)`

Las macros `RETVAL_*` fijan el valor, pero no retornan.

- `RETVAL_FALSE`
- `RETVAL_TRUE`
- `RETVAL_LONG(l)`
- `RETVAL_STRING(s,dup)` Si `dup` es **TRUE**, duplica la cadena
- `RETVAL_STRINGL(s,l,dup)` Devuelve la cadena (s) especificando el largo (l).
- `RETVAL_DOUBLE(d)`

Las macros anteriores harán un `estrdup()` del argumento 's', de modo que puede liberar con seguridad el argumento después de llamar a la macro, o, alternativamente, utilizar memoria asignada estáticamente.

Si su función devuelve respuestas lógicas de éxito/error, use siempre RETURN\_TRUE y RETURN\_FALSE respectivamente.

## Devolviendo valores complejos

Su función también puede devolver un tipo de datos complejo, tal como un objeto o una matriz.

Devolviendo un objeto:

1. Llame a `object_init(return_value)`.
2. Rellénela con valores. Las funciones disponibles para ello son listadas más abajo.
3. Posiblemente registre funciones para este objeto. Para obtener valores del objeto, la función deberá de obtener "this" desde la `active_symbol_table`. Su tipo deberá ser `IS_OBJECT`, y básicamente se trata de una matriz asociativa estándar (es decir, que podrá usar funciones de matriz asociativa sobre `.value.ht`). El registro en sí de la función se puede hacer utilizando:

```
add_method( return_value, nombre_func, puntero_func );
```

Las funciones utilizadas para rellenar un objeto son:

- `add_property_long( return_value, nombre_propiedad, l )` - Añade una propiedad llamada 'nombre\_propiedad', de tipo long, y con valor 'l'
- `add_property_double( return_value, nombre_propiedad, d )` - Igual, pero añadiendo un double
- `add_property_string( return_value, nombre_propiedad, cad )` - Igual, pero añadiendo una cadena
- `add_property_stringl( return_value, nombre_propiedad, cad, l )` - Igual, pero añadiendo una cadena de longitud 'l'

Devolviendo una matriz:

1. Llame a `array_init(return_value)`.
2. Rellénela con valores. Las funciones disponibles para ello son listadas más abajo.

Las funciones utilizadas para rellenar una matriz son:

- `add_assoc_long(return_value,clave,l)` - añade un elemento asociativo con clave 'clave' y valor long 'l'
- `add_assoc_double(return_value,clave,d)`
- `add_assoc_string(return_value,clave,cad,duplicar)`
- `add_assoc_stringl(return_value,clave,cad,largo,duplicar)` - especifica el largo de la cadena
- `add_index_long(return_value,indice,l)` - añade un elemento en la posición 'indice' con valor long 'l'
- `add_index_double(return_value,indice,d)`
- `add_index_string(return_value,indice,cad)`
- `add_index_stringl(return_value,indice,cad,largo)` - especifica el largo de la cadena
- `add_next_index_long(return_value,l)` - añade un elemento a la matriz en la próxima posición libre con valor long 'l'
- `add_next_index_double(return_value,d)`
- `add_next_index_string(return_value,cad)`
- `add_next_index_stringl(return_value,cad,largo)` - especifica el largo de la cadena

## Usando la lista de recursos

El PHP 3.0 tiene una forma estandarizada de tratar con distintos tipos de recursos. Esto sustituye a las listas enlazadas locales del PHP 2.0.

Funciones disponibles:

- `php3_list_insert(ptr, tipo)` - devuelve el 'id' del recurso recién insertado
- `php3_list_delete(id)` - borra el recurso con el id especificado
- `php3_list_find(id,*tipo)` - devuelve el puntero al recurso con el id especificado, y actualiza 'tipo' al tipo del mismo

Estas funciones se utilizan típicamente para controladores SQL, pero pueden utilizarse para cualquier otra cosa, como, por ejemplo, para mantener descriptores de archivo.

El código típico de un lista sería como este:

### Ejemplo C-7. Añadiendo un nuevo recurso

```
RESOURCE *recurso;

/* ...asignar memoria para el recurso y adquirirlo... */
/* añadir un recurso a la lista */
return_value->value.lval = php3_list_insert((void *) recurso, LE_RESOURCE_TYPE);
return_value->type = IS_LONG;
```

### Ejemplo C-8. Utilizando un recurso existente

```
pval *id_recurso;
RESOURCE *recurso;
int tipo;

convert_to_long(id_recurso);
recurso = php3_list_find(id_recurso->value.lval, &tipo);
if (tipo != LE_RESOURCE_TYPE) {
    php3_error(E_WARNING, "el recurso número %d tiene el tipo
equivocado", id_recurso->value.lval);
    RETURN_FALSE;
}
/* ...usar recurso... */
```

### Ejemplo C-9. Borrando un recurso

```
pval *id_recurso;
RESOURCE *recurso;
int tipo;

convert_to_long(id_recurso);
php3_list_delete(id_recurso->value.lval);
```

Los tipos de recursos deben registrarse en `php3_list.h`, en la enumeración `list_entry_type`. Además, hay que añadir código de desconexión para cada tipo de recurso definido en la función `list_entry_destructor()` de `list.c` (incluso si no hay nada que hacer para la desconexión, deberá añadir un caso vacío).

## Utilizando la tabla de recursos persistentes

El PHP 3.0 tiene una forma estándar de almacenar recursos persistentes (es decir, recursos que se mantienen entre accesos). El primer módulo que utilizó esta característica fue el MySQL y tras él fue el mSQL, así que uno puede hacerse una buena idea de cómo utilizar un recurso persistente leyendo mysql.c. Las funciones a revisar son:

```
php3_mysql_do_connect  
php3_mysql_connect()  
php3_mysql_pconnect()
```

La idea general de los módulos persistentes es:

1. Codifique todos sus módulos para que funcionen con la lista regular de recursos mencionadas en la sección (9).
2. Codifique funciones extra de conexión que comprueben si el recurso ya está en la lista de recursos persistentes. Si ya está, regístrelo en la lista regular como un puntero a la lista de recursos persistentes (debido a 1., el resto del código deberá funcionar de inmediato). Si no está en la lista, créelo, añádalo a la lista de recursos persistentes Y añada un puntero al mismo desde la lista regular de recursos. Así todo el código funcionará porque está en la lista regular, pero en la siguiente conexión el recurso ya estará en la lista persistente y podrá ser usado sin re-crearlo. Deberá registrar estos recursos con un tipo diferente (por ejemplo, LE\_MYSQL\_LINK para el enlace no persistente y LE\_MYSQL\_PLINK para un enlace persistente).

Si se leyera mysql.c, notaría que, salvo por que hay una función de conexión más compleja, no hay que cambiar nada más del resto del módulo.

Existe exactamente la misma interfaz para la lista de recursos regular y para la lista de recursos persistente, pero cambiando únicamente 'lista' por 'listap':

- php3\_plist\_insert(ptr, tipo) - devuelve el 'id' del recurso recién insertado
- php3\_plist\_delete(id)- borra el recurso con el id especificado
- php3\_plist\_find(id,\*tipo) - devuelve el puntero al recurso con el id especificado, y actualiza 'tipo' al tipo del mismo

Sin embargo, es más que probable que estas funciones se muestren inútiles cuando intente implementar un módulo persistente. Típicamente usted querrá usar el hecho de que la tabla de recursos persistentes es en realidad una matriz asociativa. Por ejemplo, en los módulos MySQL/mSQL, cuando hay una llamada a pconnect() (conexión persistente), la función combina en una cadena el servidor/usuario/clave que se pasaron a la función y codifica el enlace SQL con esta cadena como clave. La siguiente vez que alguien llame a pconnect() con el mismo servidor/usuario/clave, se generará la misma clave, y la función hará el enlace SQL en la lista persistente.

Hasta que se documente mejor, deberá mirar en mysql.c o en msql.c para ver como utilizar las capacidades de matriz asociativa de la listap.

Una cosa importante: a los recursos que van a parar a la lista de recursos persistentes \*NO\* se les debe asignar memoria usando el gestor de memoria del PHP, es decir, que NO deben ser creados utilizando emalloc() o estrdup(), etc. En este caso se debe usar las funciones habituales malloc(), strdup(), etc. La razón para esto es simple: al final de la petición (final del acceso), se borran todos los trozos de memoria asignados con el gestor de memoria del PHP. Como la lista persistente se supone que no se debe borrar al final de una petición, no se debe utilizar el gestor de memoria del PHP para asignar memoria a los recursos de la misma.

Cuando registre un recurso que vaya a estar en la lista persistente, deberá añadir destructores tanto a la lista persistente como a la no persistente. El destructor de la lista no persistente no deberá hacer nada. El de la lista persistente deberá liberar adecuadamente los recursos obtenidos por dicho tipo (por ejemplo, memoria, enlaces SQL, etc.). Tal y como pasa para los recursos no persistentes, DEBERÁ añadir destructores para cada recurso aunque no sean necesarios y estén vacíos. Recuerde que como no se pueden usar emalloc() y similares en conjunción con la lista persistente, tampoco podrá utilizar efree() aquí.

## Añadiendo directivas de configuración en tiempo de ejecución

Muchas de las características del PHP3 pueden ser configuradas en tiempo de ejecución. Estas directivas de configuración pueden aparecer tanto en el fichero php3.ini o, en el caso de la versión de módulo del Apache, en los archivos .conf del propio Apache. La ventaja de tenerlos en los archivos .conf del Apache es que se pueden configurar directorio por directorio. Esto quiere decir que cada uno puede tener un cierto safemodeexecdir, por ejemplo, mientras otro directorio puede tener otro. Esta granularidad en la configuración es especialmente útil cuando un servidor soporta múltiples servidores virtuales.

Los pasos necesarios para añadir una nueva directiva:

1. Añada la directiva a la estructura php3\_ini\_structure en mod\_php3.h.
2. En main.c, edite la función php3\_module\_startup y añada la llamada a cfg\_get\_string() o a cfg\_get\_long() según se requiera.
3. Añada la directiva, las restricciones y un comentario a la estructura php3\_commands en mod\_php3.c. Cuidado con la parte de restricciones. Las de tipo RSRC\_CONF sólo puede aparecer en los archivos .conf del Apache. Las directivas de tipo OR\_OPTIONS pueden aparecer en cualquier parte, incluso en los habituales archivos .htaccess.
4. Añada el elemento apropiado para su directiva, bien en php3take1handler(), bien en php3flaghandler().
5. Necesita añadir su nueva directiva a la sección de configuración de la función \_php3\_info() en functions/info.c.
6. Y finalmente, por supuesto, deberá utilizar su nueva directiva en algún sitio. Estará accesible como php3\_ini.directiva.

## Notas

- [1] Tenga cuidado aquí. El valor a usar se debe asignar dinámicamente y de forma manual, pues el código de manejo de memoria intentará liberar este puntero más adelante. Nunca pase memoria asignada de forma estática a SET\_VAR\_STRING.

---

[Anterior](#)

Cookies

[Inicio](#)

[Subir](#)

[Siguiete](#)

Llamando a Funciones del Usuario



# Apéndice D. El debugger de PHP

## Tabla de contenidos

[Usando el Debugger](#)

[Protocolo del debugger](#)

## Usando el Debugger

El debugger interno de PHP es útil para localizar fallos que se resisten. El debugger funciona conectandose a un puerto TCP cada vez que PHP comienza. Todos los mensajes de error de esa petición serán enviados a esta conexión TCP. Esta información se entiende que que es para un "servidor de debugger" que puede ejecutarse en un IDE o en un editor programabler (como Emacs).

Como poner en marcha el debugger:

1. Establezca un puerto TCP para el debugger en el [archivo de configuración](#) ([debugger.port](#)) y actívelo ([debugger.enabled](#)).
2. Ponga en marcha un módulo de escucha de TCP en algun sitio (por ejemplo `socket -l -s 1400` en UNIX).
3. En su código, ejecute "`debugger_on(host)`", donde *host* es la dirección IP o el nombre de el host ejecutando un módulo de escucha de TCP.

Ahora, todos los avisos, notificaciones, etc. se mostrarán en ese módulo de escucha, *incluso si lo ha desactivado con [error\\_reporting\(\)](#)*.

# Apéndice E. PHP reserved words

Here is the list of PHP reserved words, usually [constants](#) and [predefined variables](#). You won't find any functions here, but rather language constructs. You shouldn't try to use such names as variable, function, constant or method names, as it will surely lead to confusion.

<a href="#">and</a>	<a href="#">E_PARSE</a>	<a href="#">old_function</a>
<a href="#">\$argv</a>	<a href="#">E_ERROR</a>	<a href="#">or</a>
<a href="#">as</a>	<a href="#">E_WARNING</a>	parent
<a href="#">\$argc</a>	<a href="#">eval</a>	<a href="#">PHP_OS</a>
<a href="#">break</a>	<a href="#">exit()</a>	<a href="#">\$PHP_SELF</a>
<a href="#">case</a>	<a href="#">extends</a>	<a href="#">PHP_VERSION</a>
<a href="#">cfunction</a>	<b><a href="#">FALSE</a></b>	<a href="#">print()</a>
<a href="#">class</a>	<a href="#">for</a>	<a href="#">require()</a>
<a href="#">continue</a>	<a href="#">foreach</a>	<a href="#">require_once()</a>
declare	<a href="#">function</a>	<a href="#">return</a>
<a href="#">default</a>	<a href="#">\$HTTP_COOKIE_VARS</a>	<a href="#">static</a>
<a href="#">do</a>	<a href="#">\$HTTP_GET_VARS</a>	<a href="#">switch</a>
<a href="#">die()</a>	<a href="#">\$HTTP_POST_VARS</a>	<a href="#">stdClass</a>
<a href="#">echo()</a>	<a href="#">\$HTTP_POST_FILES</a>	<a href="#">\$this</a>
<a href="#">else</a>	<a href="#">\$HTTP_ENV_VARS</a>	<b><a href="#">TRUE</a></b>
<a href="#">elseif</a>	<a href="#">\$HTTP_SERVER_VARS</a>	<a href="#">var</a>
<a href="#">empty()</a>	<a href="#">if</a>	<a href="#">xor</a>
enddeclare	<a href="#">include()</a>	<a href="#">virtual()</a>
<a href="#">endfor</a>	<a href="#">include_once()</a>	<a href="#">while</a>
<a href="#">endforeach</a>	<a href="#">global</a>	<a href="#">__FILE__</a>
<a href="#">endif</a>	<a href="#">list()</a>	<a href="#">__LINE__</a>

[endswitch](#) [new](#)

[\\_\\_sleep](#)

[endwhile](#) [not](#)

[\\_\\_wakeup](#)

[E\\_ALL](#) NULL

[Anterior](#)

[Inicio](#)

[Siguiete](#)

Protocolo del debugger

[Subir](#)

PHP Resource Types

## Apéndice F. PHP Resource Types

The following is a list of functions which create, use or destroy PHP resources. The function `is_resource()` can be used to determine if a variable is a resource and `get_resource_type()` will return the type of resource it is.

Tabla F-1. Resource Types

Resource Type Name	Created By	Used By	Destroyed By	Definition
aspell	<a href="#">aspell_new()</a>	<a href="#">aspell_check()</a> , <a href="#">aspell_check_raw()</a> , <a href="#">aspell_suggest()</a>	None	Aspell dictionary
bzip2	<a href="#">bzopen()</a>	<a href="#">bzerrno()</a> , <a href="#">bzerror()</a> , <a href="#">bzerrstr()</a> , <a href="#">bzflush()</a> , <a href="#">bzread()</a> , <a href="#">bzwrite()</a>	<a href="#">bzclose()</a>	Bzip2 file
COM	<a href="#">com_load()</a>	<a href="#">com_invoke()</a> , <a href="#">com_propget()</a> , <a href="#">com_get()</a> , <a href="#">com_propput()</a> , <a href="#">com_set()</a> , <a href="#">com_propput()</a>	None	COM object reference
VARIANT				
cpdf	<a href="#">cpdf_open()</a>	<a href="#">cpdf_page_init()</a> , <a href="#">cpdf_finalize_page()</a> , <a href="#">cpdf_finalize()</a> , <a href="#">cpdf_output_buffer()</a> , <a href="#">cpdf_save_to_file()</a> , <a href="#">cpdf_set_current_page()</a> , <a href="#">cpdf_begin_text()</a> , <a href="#">cpdf_end_text()</a> , <a href="#">cpdf_show()</a> , <a href="#">cpdf_show_xy()</a> , <a href="#">cpdf_text()</a> , <a href="#">cpdf_set_font()</a> , <a href="#">cpdf_set_leading()</a> , <a href="#">cpdf_set_text_rendering()</a> , <a href="#">cpdf_set_horiz_scaling()</a> , <a href="#">cpdf_set_text_rise()</a> , <a href="#">cpdf_set_text_matrix()</a> , <a href="#">cpdf_set_text_pos()</a> , <a href="#">cpdf_set_text_pos()</a> , <a href="#">cpdf_set_word_spacing()</a> , <a href="#">cpdf_continue_text()</a> , <a href="#">cpdf_stringwidth()</a> , <a href="#">cpdf_save()</a> , <a href="#">cpdf_translate()</a> , <a href="#">cpdf_restore()</a> , <a href="#">cpdf_scale()</a> , <a href="#">cpdf_rotate()</a> , <a href="#">cpdf_setflat()</a> , <a href="#">cpdf_setlinejoin()</a> , <a href="#">cpdf_setlinecap()</a> , <a href="#">cpdf_setmiterlimit()</a> , <a href="#">cpdf_setlinewidth()</a> , <a href="#">cpdf_setdash()</a> , <a href="#">cpdf_moveto()</a> , <a href="#">cpdf_rmoveto()</a> , <a href="#">cpdf_curveto()</a> , <a href="#">cpdf_lineto()</a> , <a href="#">cpdf_rlineto()</a> , <a href="#">cpdf_circle()</a> , <a href="#">cpdf_arc()</a> , <a href="#">cpdf_rect()</a> , <a href="#">cpdf_closepath()</a> , <a href="#">cpdf_stroke()</a> , <a href="#">cpdf_closepath_fill_stroke()</a> , <a href="#">cpdf_fill_stroke()</a> , <a href="#">cpdf_clip()</a> , <a href="#">cpdf_fill()</a> , <a href="#">cpdf_setgray_fill()</a> , <a href="#">cpdf_setgray_stroke()</a> , <a href="#">cpdf_setgray()</a> , <a href="#">cpdf_setrgbcolor_fill()</a> , <a href="#">cpdf_setrgbcolor_stroke()</a> , <a href="#">cpdf_setrgbcolor()</a> , <a href="#">cpdf_add_outline()</a> ,	<a href="#">cpdf_close()</a>	PDF document with CPDF lib

		<a href="#">cpdf_set_page_animation()</a> , <a href="#">cpdf_import_jpeg()</a> , <a href="#">cpdf_place_inline_image()</a> , <a href="#">cpdf_add_annotation()</a>		
cpdf outline				
curl	<a href="#">curl_init()</a>	<a href="#">curl_init()</a> , <a href="#">curl_exec()</a>	<a href="#">curl_close()</a>	Curl session
dbm	<a href="#">dbmopen()</a>	<a href="#">dbmexists()</a> , <a href="#">dbmfetch()</a> , <a href="#">dbminsert()</a> , <a href="#">dbmreplace()</a> , <a href="#">dbmdelete()</a> , <a href="#">dbmfirstkey()</a> , <a href="#">dbmnextkey()</a>	<a href="#">dbmclose()</a>	Link to DBM database
dba	<a href="#">dba_popen()</a>	<a href="#">dba_delete()</a> , <a href="#">dba_exists()</a> , <a href="#">dba_fetch()</a> , <a href="#">dba_firstkey()</a> , <a href="#">dba_insert()</a> , <a href="#">dba_nextkey()</a> , <a href="#">dba_optimize()</a> , <a href="#">dba_replace()</a> , <a href="#">dba_sync()</a>	<a href="#">dba_close()</a>	Link to DBA base
dba persistent	<a href="#">dba_open()</a>	<a href="#">dba_delete()</a> , <a href="#">dba_exists()</a> , <a href="#">dba_fetch()</a> , <a href="#">dba_firstkey()</a> , <a href="#">dba_insert()</a> , <a href="#">dba_nextkey()</a> , <a href="#">dba_optimize()</a> , <a href="#">dba_replace()</a> , <a href="#">dba_sync()</a>	None	Persistent link to DBA base
dbase	<a href="#">dbase_open()</a>	<a href="#">dbase_pack()</a> , <a href="#">dbase_add_record()</a> , <a href="#">dbase_replace_record()</a> , <a href="#">dbase_delete_record()</a> , <a href="#">dbase_get_record()</a> , <a href="#">dbase_get_record_with_names()</a> , <a href="#">dbase_numfields()</a> , <a href="#">dbase_numrecords()</a>	<a href="#">dbase_close()</a>	Link to Dbase base
dbx_link_object	<a href="#">dbx_connect()</a>	<a href="#">dbx_query()</a>	<a href="#">dbx_close()</a>	dbx connection
dbx_result_object	<a href="#">dbx_query()</a>	()	None	dbx result
domxml attribute				
domxml document				
domxml node				
xpath context				
xpath object				
fbsql database	<a href="#">fbsql_select_db()</a>	()	None	fbsql database
fbsql link	<a href="#">fbsql_change_user()</a> , <a href="#">fbsql_connect()</a>	<a href="#">fbsql_autocommit()</a> , <a href="#">fbsql_change_user()</a> , <a href="#">fbsql_create_db()</a> , <a href="#">fbsql_data_seek()</a> , <a href="#">fbsql_db_query()</a> , <a href="#">fbsql_drop_db()</a> , (), <a href="#">fbsql_select_db()</a> , <a href="#">fbsql_errno()</a> , <a href="#">fbsql_error()</a> , <a href="#">fbsql_insert_id()</a> , <a href="#">fbsql_list_dbs()</a>	<a href="#">fbsql_close()</a>	Link to fbsql database
fbsql plink	<a href="#">fbsql_change_user()</a> , <a href="#">fbsql_pconnect()</a>	<a href="#">fbsql_autocommit()</a> , <a href="#">fbsql_change_user()</a> , <a href="#">fbsql_create_db()</a> , <a href="#">fbsql_data_seek()</a> , <a href="#">fbsql_db_query()</a> , <a href="#">fbsql_drop_db()</a> , (), <a href="#">fbsql_select_db()</a> , <a href="#">fbsql_errno()</a> , <a href="#">fbsql_error()</a> , <a href="#">fbsql_insert_id()</a> , <a href="#">fbsql_list_dbs()</a>	None	Persistent link to fbsql database

fbsql result	<a href="#">fbsql_db_query()</a> , <a href="#">fbsql_list_dbs()</a> , <a href="#">fbsql_query()</a> , <a href="#">fbsql_list_fields()</a> , <a href="#">fbsql_list_tables()</a> , <a href="#">fbsql_tablename()</a>	<a href="#">fbsql_affected_rows()</a> , <a href="#">fbsql_fetch_array()</a> , <a href="#">fbsql_fetch_assoc()</a> , <a href="#">fbsql_fetch_field()</a> , <a href="#">fbsql_fetch_lengths()</a> , <a href="#">fbsql_fetch_object()</a> , <a href="#">fbsql_fetch_row()</a> , <a href="#">fbsql_field_flags()</a> , <a href="#">fbsql_field_name()</a> , <a href="#">fbsql_field_len()</a> , <a href="#">fbsql_field_seek()</a> , <a href="#">fbsql_field_table()</a> , <a href="#">fbsql_field_type()</a> , <a href="#">fbsql_next_result()</a> , <a href="#">fbsql_num_fields()</a> , <a href="#">fbsql_num_rows()</a> , <a href="#">fbsql_result()</a> , <a href="#">fbsql_num_rows()</a>	<a href="#">fbsql_free_result()</a>	fbsql result
fdf	<a href="#">fdf_open()</a>	<a href="#">fdf_create()</a> , <a href="#">fdf_save()</a> , <a href="#">fdf_get_value()</a> , <a href="#">fdf_set_value()</a> , <a href="#">fdf_next_field_name()</a> , <a href="#">fdf_set_ap()</a> , <a href="#">fdf_set_status()</a> , <a href="#">fdf_get_status()</a> , <a href="#">fdf_set_file()</a> , <a href="#">fdf_get_file()</a> , <a href="#">fdf_set_flags()</a> , <a href="#">fdf_set_opt()</a> , <a href="#">fdf_set_submit_form_action()</a> , <a href="#">fdf_set_javascript_action()</a>	<a href="#">fdf_close()</a>	FDF File
ftp	<a href="#">ftp_connect()</a>	<a href="#">ftp_login()</a> , <a href="#">ftp_pwd()</a> , <a href="#">ftp_cdup()</a> , <a href="#">ftp_chdir()</a> , <a href="#">ftp_mkdir()</a> , <a href="#">ftp_rmdir()</a> , <a href="#">ftp_nlist()</a> , <a href="#">ftp_rawlist()</a> , <a href="#">ftp_systype()</a> , <a href="#">ftp_pasv()</a> , <a href="#">ftp_get()</a> , <a href="#">ftp_fget()</a> , <a href="#">ftp_put()</a> , <a href="#">ftp_fput()</a> , <a href="#">ftp_size()</a> , <a href="#">ftp_mdtm()</a> , <a href="#">ftp_rename()</a> , <a href="#">ftp_delete()</a> , <a href="#">ftp_site()</a>	<a href="#">ftp_quit()</a>	FTP stream
gd	<a href="#">imagecreate()</a> , <a href="#">imagecreatefromgif()</a> , <a href="#">imagecreatefromjpeg()</a> , <a href="#">imagecreatefrompng()</a> , <a href="#">imagecreatefromwbmp()</a> , <a href="#">imagecreatefromstring()</a> , <a href="#">imagecreatetruecolor()</a>	<a href="#">imagearc()</a> , <a href="#">imagechar()</a> , <a href="#">imagecharup()</a> , <a href="#">imagecolorallocate()</a> , <a href="#">imagecolorat()</a> , <a href="#">imagecolorclosest()</a> , <a href="#">imagecolorexact()</a> , <a href="#">imagecolorresolve()</a> , <a href="#">imagegammaconvert()</a> , <a href="#">imagegammaconvert()</a> , <a href="#">imagecolorset()</a> , <a href="#">imagecolorsforindex()</a> , <a href="#">imagecolorstotal()</a> , <a href="#">imagecolortransparent()</a> , <a href="#">imagecopy()</a> , <a href="#">imagecopyresized()</a> , <a href="#">imagedashedline()</a> , <a href="#">imagefill()</a> , <a href="#">imagefilledpolygon()</a> , <a href="#">imagefilledrectangle()</a> , <a href="#">imagefilltoborder()</a> , <a href="#">imagegif()</a> , <a href="#">imagepng()</a> , <a href="#">imagejpeg()</a> , <a href="#">imagewbmp()</a> , <a href="#">imageinterlace()</a> , <a href="#">imageline()</a> , <a href="#">imagepolygon()</a> , <a href="#">imagepstext()</a> , <a href="#">imagerectangle()</a> , <a href="#">imagesetpixel()</a> , <a href="#">imagestring()</a> , <a href="#">imagestringup()</a> , <a href="#">imagesx()</a> , <a href="#">imagesy()</a> , <a href="#">imagettftext()</a> , <a href="#">imagefilledarc()</a> , <a href="#">imageellipse()</a> , <a href="#">imagefilledellipse()</a> , <a href="#">imagecolorclosestalpha()</a> , <a href="#">imagecolorexactalpha()</a> , <a href="#">imagecolorresolvealpha()</a> , <a href="#">imagecopymerge()</a> , <a href="#">imagecopymergegray()</a> , <a href="#">imagecopyresampled()</a> , <a href="#">imagetruecolorpalette()</a> , <a href="#">imagesetbrush()</a> , <a href="#">imagesettile()</a> , <a href="#">imagesetthickness()</a>	<a href="#">imagedestroy()</a>	GD Image

gd font	<a href="#">imagedloadfont()</a>	<a href="#">imagechar()</a> , <a href="#">imagecharup()</a> , <a href="#">imagefontheight()</a>	None	Font for GD
gd PS encoding				
gd PS font	<a href="#">imagepsloadfont()</a>	<a href="#">imagepstext()</a> , <a href="#">imagepslantfont()</a> , <a href="#">imagepsextendfont()</a> , <a href="#">imagepsencodefont()</a> , <a href="#">imagepsbbox()</a>	<a href="#">imagepsfreefont()</a>	PS font for GD
GMP integer	<a href="#">gmp_init()</a>	<a href="#">gmp_intval()</a> , <a href="#">gmp_strval()</a> , <a href="#">gmp_add()</a> , <a href="#">gmp_sub()</a> , <a href="#">gmp_mul()</a> , <a href="#">gmp_div_q()</a> , <a href="#">gmp_div_r()</a> , <a href="#">gmp_div_qr()</a> , <a href="#">gmp_div()</a> , <a href="#">gmp_mod()</a> , <a href="#">gmp_divexact()</a> , <a href="#">gmp_cmp()</a> , <a href="#">gmp_neg()</a> , <a href="#">gmp_abs()</a> , <a href="#">gmp_sign()</a> , <a href="#">gmp_fact()</a> , <a href="#">gmp_sqrt()</a> , <a href="#">gmp_sqrtrm()</a> , <a href="#">gmp_perfect_square()</a> , <a href="#">gmp_pow()</a> , <a href="#">gmp_powm()</a> , <a href="#">gmp_prob_prime()</a> , <a href="#">gmp_gcd()</a> , <a href="#">gmp_gcdext()</a> , <a href="#">gmp_invert()</a> , <a href="#">gmp_legendre()</a> , <a href="#">gmp_jacobi()</a> , <a href="#">gmp_random()</a> , <a href="#">gmp_and()</a> , <a href="#">gmp_or()</a> , <a href="#">gmp_xor()</a> , <a href="#">gmp_setbit()</a> , <a href="#">gmp_clrbit()</a> , <a href="#">gmp_scan0()</a> , <a href="#">gmp_scan1()</a> , <a href="#">gmp_popcount()</a> , <a href="#">gmp_hamdist()</a>	None	GMP Number
hyperwave document	<a href="#">hw_cp()</a> , <a href="#">hw_docbyanchor()</a> , <a href="#">hw_getremote()</a> , <a href="#">hw_getremotechildren()</a>	<a href="#">hw_children()</a> , <a href="#">hw_childrenobj()</a> , <a href="#">hw_getparents()</a> , <a href="#">hw_getparentsobj()</a> , <a href="#">hw_getchildcoll()</a> , <a href="#">hw_getchildcollobj()</a> , <a href="#">hw_getremote()</a> , <a href="#">hw_getsrcbydestobj()</a> , <a href="#">hw_getandlock()</a> , <a href="#">hw_gettext()</a> , <a href="#">hw_getobjectbyquerycoll()</a> , <a href="#">hw_getobjectbyquerycollobj()</a> , <a href="#">hw_getchilddoccoll()</a> , <a href="#">hw_getchilddoccollobj()</a> , <a href="#">hw_getanchors()</a> , <a href="#">hw_getanchorsobj()</a> , <a href="#">hw_inscoll()</a> , <a href="#">hw_pipedocument()</a> , <a href="#">hw_unlock()</a>	<a href="#">hw_deleteobject()</a>	Hyperwave object
hyperwave link	<a href="#">hw_connect()</a>	<a href="#">hw_children()</a> , <a href="#">hw_childrenobj()</a> , <a href="#">hw_cp()</a> , <a href="#">hw_deleteobject()</a> , <a href="#">hw_docbyanchor()</a> , <a href="#">hw_docbyanchorobj()</a> , <a href="#">hw_errormsg()</a> , <a href="#">hw_edittext()</a> , <a href="#">hw_error()</a> , <a href="#">hw_getparents()</a> , <a href="#">hw_getparentsobj()</a> , <a href="#">hw_getchildcoll()</a> , <a href="#">hw_getchildcollobj()</a> , <a href="#">hw_getremote()</a> , <a href="#">hw_getremotechildren()</a> , <a href="#">hw_getsrcbydestobj()</a> , <a href="#">hw_getobject()</a> , <a href="#">hw_getandlock()</a> , <a href="#">hw_gettext()</a> , <a href="#">hw_getobjectbyquery()</a> , <a href="#">hw_getobjectbyqueryobj()</a> , <a href="#">hw_getobjectbyquerycoll()</a> , <a href="#">hw_getobjectbyquerycollobj()</a> , <a href="#">hw_getchilddoccoll()</a> , <a href="#">hw_getchilddoccollobj()</a> , <a href="#">hw_getanchors()</a> , <a href="#">hw_getanchorsobj()</a> , <a href="#">hw_mv()</a> , <a href="#">hw_incollections()</a> , <a href="#">hw_info()</a> , <a href="#">hw_inscoll()</a> , <a href="#">hw_insdock()</a> , <a href="#">hw_insertdocument()</a> , <a href="#">hw_insertobject()</a> ,	<a href="#">hw_close()</a> , <a href="#">hw_free_document()</a>	Link to Hyperwave server

		<a href="#">hw_mapid()</a> , <a href="#">hw_modifyobject()</a> , <a href="#">hw_pipedocument()</a> , <a href="#">hw_unlock()</a> , <a href="#">hw_who()</a> , <a href="#">hw_getusername()</a>		
hyperwave link persistent	<a href="#">hw_pconnect()</a>	<a href="#">hw_children()</a> , <a href="#">hw_childrenobj()</a> , <a href="#">hw_cp()</a> , <a href="#">hw_deleteobject()</a> , <a href="#">hw_docbyanchor()</a> , <a href="#">hw_docbyanchorobj()</a> , <a href="#">hw_errormsg()</a> , <a href="#">hw_edittext()</a> , <a href="#">hw_error()</a> , <a href="#">hw_getparents()</a> , <a href="#">hw_getparentsobj()</a> , <a href="#">hw_getchildcoll()</a> , <a href="#">hw_getchildcollobj()</a> , <a href="#">hw_getremote()</a> , <a href="#">hw_getremotechildren()</a> , <a href="#">hw_getsrcbydestobj()</a> , <a href="#">hw_getobject()</a> , <a href="#">hw_getandlock()</a> , <a href="#">hw_gettext()</a> , <a href="#">hw_getobjectbyquery()</a> , <a href="#">hw_getobjectbyqueryobj()</a> , <a href="#">hw_getobjectbyquerycoll()</a> , <a href="#">hw_getobjectbyquerycollobj()</a> , <a href="#">hw_getchilddoccoll()</a> , <a href="#">hw_getchilddoccollobj()</a> , <a href="#">hw_getanchors()</a> , <a href="#">hw_getanchorsobj()</a> , <a href="#">hw_mv()</a> , <a href="#">hw_incollections()</a> , <a href="#">hw_info()</a> , <a href="#">hw_inscoll()</a> , <a href="#">hw_insdock()</a> , <a href="#">hw_insertdocument()</a> , <a href="#">hw_insertobject()</a> , <a href="#">hw_mapid()</a> , <a href="#">hw_modifyobject()</a> , <a href="#">hw_pipedocument()</a> , <a href="#">hw_unlock()</a> , <a href="#">hw_who()</a> , <a href="#">hw_getusername()</a>	None	Persistent link to Hyperwave server
icap	<a href="#">icap_open()</a>	<a href="#">icap_fetch_event()</a> , <a href="#">icap_list_events()</a> , <a href="#">icap_store_event()</a> , <a href="#">icap_snooze()</a> , <a href="#">icap_list_alarms()</a> , <a href="#">icap_delete_event()</a>	<a href="#">icap_close()</a>	Link to icap server
imap	<a href="#">imap_open()</a>	<a href="#">imap_append()</a> , <a href="#">imap_body()</a> , <a href="#">imap_check()</a> , <a href="#">imap_createmailbox()</a> , <a href="#">imap_delete()</a> , <a href="#">imap_deletemailbox()</a> , <a href="#">imap_expunge()</a> , <a href="#">imap_fetchbody()</a> , <a href="#">imap_fetchstructure()</a> , <a href="#">imap_headerinfo()</a> , <a href="#">imap_header()</a> , <a href="#">imap_headers()</a> , <a href="#">imap_listmailbox()</a> , <a href="#">imap_getmailboxes()</a> , <a href="#">imap_get_quota()</a> , <a href="#">imap_status()</a> , <a href="#">imap_listsubscribed()</a> , <a href="#">imap_set_quota()</a> , <a href="#">imap_set_quota()</a> , <a href="#">imap_getsubscribed()</a> , <a href="#">imap_mail_copy()</a> , <a href="#">imap_mail_move()</a> , <a href="#">imap_num_msg()</a> , <a href="#">imap_num_recent()</a> , <a href="#">imap_ping()</a> , <a href="#">imap_renamemailbox()</a> , <a href="#">imap_reopen()</a> , <a href="#">imap_subscribe()</a> , <a href="#">imap_undelete()</a> , <a href="#">imap_unsubscribe()</a> , <a href="#">imap_scanmailbox()</a> , <a href="#">imap_mailboxmsginfo()</a> , <a href="#">imap_fetchheader()</a> , <a href="#">imap_uid()</a> , <a href="#">imap_msgno()</a> , <a href="#">imap_search()</a> , <a href="#">imap_fetch_overview()</a>	<a href="#">imap_close()</a>	Link to IMAP, POP3 server
imap chain persistent				
imap persistent				



ingres	<a href="#">ingres_connect()</a>	<a href="#">ingres_query()</a> , <a href="#">ingres_num_rows()</a> , <a href="#">ingres_num_fields()</a> , <a href="#">ingres_field_name()</a> , <a href="#">ingres_field_type()</a> , <a href="#">ingres_field_nullable()</a> , <a href="#">ingres_field_length()</a> , <a href="#">ingres_field_precision()</a> , <a href="#">ingres_field_scale()</a> , <a href="#">ingres_fetch_array()</a> , <a href="#">ingres_fetch_row()</a> , <a href="#">ingres_fetch_object()</a> , <a href="#">ingres_rollback()</a> , <a href="#">ingres_commit()</a> , <a href="#">ingres_autocommit()</a>	<a href="#">ingres_close()</a>	Link to ingresII base
ingres persistent	<a href="#">ingres_pconnect()</a>	<a href="#">ingres_query()</a> , <a href="#">ingres_num_rows()</a> , <a href="#">ingres_num_fields()</a> , <a href="#">ingres_field_name()</a> , <a href="#">ingres_field_type()</a> , <a href="#">ingres_field_nullable()</a> , <a href="#">ingres_field_length()</a> , <a href="#">ingres_field_precision()</a> , <a href="#">ingres_field_scale()</a> , <a href="#">ingres_fetch_array()</a> , <a href="#">ingres_fetch_row()</a> , <a href="#">ingres_fetch_object()</a> , <a href="#">ingres_rollback()</a> , <a href="#">ingres_commit()</a> , <a href="#">ingres_autocommit()</a>	None	Persistent link to ingresII base
interbase blob				
interbase link	<a href="#">ibase_connect()</a>	<a href="#">ibase_query()</a> , <a href="#">ibase_prepare()</a> , <a href="#">ibase_trans()</a>	<a href="#">ibase_close()</a>	Link to Interbase database
interbase link persistent	<a href="#">ibase_pconnect()</a>	<a href="#">ibase_query()</a> , <a href="#">ibase_prepare()</a> , <a href="#">ibase_trans()</a>	None	Persistent link to Interbase database
interbase query	<a href="#">ibase_prepare()</a>	<a href="#">ibase_execute()</a>	<a href="#">ibase_free_query()</a>	Interbase query
interbase result	<a href="#">ibase_query()</a>	<a href="#">ibase_fetch_row()</a> , <a href="#">ibase_fetch_object()</a> , <a href="#">ibase_field_info()</a> , <a href="#">ibase_num_fields()</a>	<a href="#">ibase_free_result()</a>	Interbase Result
interbase transaction	<a href="#">ibase_trans()</a>	<a href="#">ibase_commit()</a>	<a href="#">ibase_rollback()</a>	Interbase transaction
java				
ldap link	<a href="#">ldap_connect()</a> , <a href="#">ldap_search()</a>	<a href="#">ldap_count_entries()</a> , <a href="#">ldap_first_attribute()</a> , <a href="#">ldap_first_entry()</a> , <a href="#">ldap_get_attributes()</a> , <a href="#">ldap_get_dn()</a> , <a href="#">ldap_get_entries()</a> , <a href="#">ldap_get_values()</a> , <a href="#">ldap_get_values_len()</a> , <a href="#">ldap_next_attribute()</a> , <a href="#">ldap_next_entry()</a>	<a href="#">ldap_close()</a>	ldap connection

ldap result	<a href="#">ldap_read()</a>	<a href="#">ldap_add()</a> , <a href="#">ldap_compare()</a> , <a href="#">ldap_bind()</a> , <a href="#">ldap_count_entries()</a> , <a href="#">ldap_delete()</a> , <a href="#">ldap_errno()</a> , <a href="#">ldap_error()</a> , <a href="#">ldap_first_attribute()</a> , <a href="#">ldap_first_entry()</a> , <a href="#">ldap_get_attributes()</a> , <a href="#">ldap_get_dn()</a> , <a href="#">ldap_get_entries()</a> , <a href="#">ldap_get_values()</a> , <a href="#">ldap_get_values_len()</a> , <a href="#">ldap_get_option()</a> , <a href="#">ldap_list()</a> , <a href="#">ldap_modify()</a> , <a href="#">ldap_mod_add()</a> , <a href="#">ldap_mod_replace()</a> , <a href="#">ldap_next_attribute()</a> , <a href="#">ldap_next_entry()</a> , <a href="#">ldap_mod_del()</a> , <a href="#">ldap_set_option()</a> , <a href="#">ldap_unbind()</a>	<a href="#">ldap_free_result()</a>	ldap search result
ldap result entry				
mcal	<a href="#">mcal_open()</a> , <a href="#">mcal_popen()</a>	<a href="#">mcal_create_calendar()</a> , <a href="#">mcal_rename_calendar()</a> , <a href="#">mcal_rename_calendar()</a> , <a href="#">mcal_delete_calendar()</a> , <a href="#">mcal_fetch_event()</a> , <a href="#">mcal_list_events()</a> , <a href="#">mcal_append_event()</a> , <a href="#">mcal_store_event()</a> , <a href="#">mcal_delete_event()</a> , <a href="#">mcal_list_alarms()</a> , <a href="#">mcal_event_init()</a> , <a href="#">mcal_event_set_category()</a> , <a href="#">mcal_event_set_title()</a> , <a href="#">mcal_event_set_description()</a> , <a href="#">mcal_event_set_start()</a> , <a href="#">mcal_event_set_end()</a> , <a href="#">mcal_event_set_alarm()</a> , <a href="#">mcal_event_set_class()</a> , <a href="#">mcal_next_recurrence()</a> , <a href="#">mcal_event_set_recur_none()</a> , <a href="#">mcal_event_set_recur_daily()</a> , <a href="#">mcal_event_set_recur_weekly()</a> , <a href="#">mcal_event_set_recur_monthly_mday()</a> , <a href="#">mcal_event_set_recur_monthly_wday()</a> , <a href="#">mcal_event_set_recur_yearly()</a> , <a href="#">mcal_fetch_current_stream_event()</a> , <a href="#">mcal_event_add_attribute()</a> , <a href="#">mcal_expunge()</a>	<a href="#">mcal_close()</a>	Link to calendar server
SWFAction				
SWFBitmap				
SWFButton				
SWFDisplayItem				
SWFFill				
SWFFont				
SWFGradient				
SWFMorph				
SWFMovie				
SWFShape				
SWFSprite				

SWFText				
SWFTextField				
mnogosearch agent				
mnogosearch result				
mysql link	<a href="#">mysql_connect()</a>	<a href="#">mysql()</a> , <a href="#">mysql_create_db()</a> , <a href="#">mysql_createdb()</a> , <a href="#">mysql_drop_db()</a> , <a href="#">mysql_drop_db()</a> , <a href="#">mysql_select_db()</a> , <a href="#">mysql_select_db()</a>	<a href="#">mysql_close()</a>	Link to mSQL database
mysql link persistent	<a href="#">mysql_pconnect()</a>	<a href="#">mysql()</a> , <a href="#">mysql_create_db()</a> , <a href="#">mysql_createdb()</a> , <a href="#">mysql_drop_db()</a> , <a href="#">mysql_drop_db()</a> , <a href="#">mysql_select_db()</a> , <a href="#">mysql_select_db()</a>	None	Persistent link to mSQL
mysql query	<a href="#">mysql_query()</a>	<a href="#">mysql()</a> , <a href="#">mysql_affected_rows()</a> , <a href="#">mysql_data_seek()</a> , <a href="#">mysql_dbname()</a> , <a href="#">mysql_fetch_array()</a> , <a href="#">mysql_fetch_field()</a> , <a href="#">mysql_fetch_object()</a> , <a href="#">mysql_fetch_row()</a> , <a href="#">mysql_fieldname()</a> , <a href="#">mysql_field_seek()</a> , <a href="#">mysql_fieldtable()</a> , <a href="#">mysql_fieldtype()</a> , <a href="#">mysql_fieldflags()</a> , <a href="#">mysql_fieldlen()</a> , <a href="#">mysql_num_fields()</a> , <a href="#">mysql_num_rows()</a> , <a href="#">mysql_numfields()</a> , <a href="#">mysql_numrows()</a> , <a href="#">mysql_result()</a>	<a href="#">mysql_free_result()</a> , <a href="#">mysql_free_result()</a>	mSQL result
mssql link	<a href="#">mssql_connect()</a>	<a href="#">mssql_query()</a> , <a href="#">mssql_select_db()</a>	<a href="#">mssql_close()</a>	Link to Microsoft SQL Server database
mssql link persistent	<a href="#">mssql_pconnect()</a>	<a href="#">mssql_query()</a> , <a href="#">mssql_select_db()</a>	None	Persistent link to Microsoft SQL Server
mssql result	<a href="#">mssql_query()</a>	<a href="#">mssql_data_seek()</a> , <a href="#">mssql_fetch_array()</a> , <a href="#">mssql_fetch_field()</a> , <a href="#">mssql_fetch_object()</a> , <a href="#">mssql_fetch_row()</a> , <a href="#">mssql_field_length()</a> , <a href="#">mssql_field_name()</a> , <a href="#">mssql_field_seek()</a> , <a href="#">mssql_field_type()</a> , <a href="#">mssql_num_fields()</a> , <a href="#">mssql_num_rows()</a> , <a href="#">mssql_result()</a>	<a href="#">mssql_free_result()</a>	Microsoft SQL Server result
mysql link	<a href="#">mysql_connect()</a>	<a href="#">mysql_affected_rows()</a> , <a href="#">mysql_change_user()</a> , <a href="#">mysql_create_db()</a> , <a href="#">mysql_data_seek()</a> , <a href="#">mysql_db_name()</a> , <a href="#">mysql_db_query()</a> , <a href="#">mysql_drop_db()</a> , <a href="#">mysql_errno()</a> , <a href="#">mysql_error()</a> , <a href="#">mysql_insert_id()</a> , <a href="#">mysql_list_dbs()</a> , <a href="#">mysql_list_fields()</a> , <a href="#">mysql_list_tables()</a> , <a href="#">mysql_query()</a> , <a href="#">mysql_result()</a> , <a href="#">mysql_select_db()</a> , <a href="#">mysql_tablename()</a> , <a href="#">mysql_get_host_info()</a> , <a href="#">mysql_get_proto_info()</a> , <a href="#">mysql_get_server_info()</a>	<a href="#">mysql_close()</a>	Link to MySQL database

mysql link persistent	<a href="#">mysql_pconnect()</a>	<a href="#">mysql_affected_rows()</a> , <a href="#">mysql_change_user()</a> , <a href="#">mysql_create_db()</a> , <a href="#">mysql_data_seek()</a> , <a href="#">mysql_db_name()</a> , <a href="#">mysql_db_query()</a> , <a href="#">mysql_drop_db()</a> , <a href="#">mysql_errno()</a> , <a href="#">mysql_error()</a> , <a href="#">mysql_insert_id()</a> , <a href="#">mysql_list_dbs()</a> , <a href="#">mysql_list_fields()</a> , <a href="#">mysql_list_tables()</a> , <a href="#">mysql_query()</a> , <a href="#">mysql_result()</a> , <a href="#">mysql_select_db()</a> , <a href="#">mysql_tablename()</a> , <a href="#">mysql_get_host_info()</a> , <a href="#">mysql_get_proto_info()</a> , <a href="#">mysql_get_server_info()</a>	None	Persistent link to MySQL database
mysql result	<a href="#">mysql_db_query()</a> , <a href="#">mysql_list_dbs()</a> , <a href="#">mysql_list_fields()</a> , <a href="#">mysql_list_tables()</a> , <a href="#">mysql_query()</a>	<a href="#">mysql_data_seek()</a> , <a href="#">mysql_db_name()</a> , <a href="#">mysql_fetch_array()</a> , <a href="#">mysql_fetch_assoc()</a> , <a href="#">mysql_fetch_field()</a> , <a href="#">mysql_fetch_lengths()</a> , <a href="#">mysql_fetch_object()</a> , <a href="#">mysql_fetch_row()</a> , <a href="#">mysql_fetch_row()</a> , <a href="#">mysql_field_flags()</a> , <a href="#">mysql_field_name()</a> , <a href="#">mysql_field_len()</a> , <a href="#">mysql_field_seek()</a> , <a href="#">mysql_field_table()</a> , <a href="#">mysql_field_type()</a> , <a href="#">mysql_num_fields()</a> , <a href="#">mysql_num_rows()</a> , <a href="#">mysql_result()</a> , <a href="#">mysql_tablename()</a>	<a href="#">mysql_free_result()</a>	MySQL result
oci8 collection				
oci8 connection	<a href="#">ocilogon()</a> , <a href="#">ociplogon()</a> , <a href="#">ocinlogon()</a>	<a href="#">ocicommit()</a> , <a href="#">ociserverversion()</a> , <a href="#">ocinewcursor()</a> , <a href="#">ociparse()</a> , <a href="#">ocierror()</a>	<a href="#">ocilogoff()</a>	Link to Oracle database
oci8 descriptor				
oci8 server				
oci8 session				
oci8 statement	<a href="#">ocinewdescriptor()</a>	<a href="#">ocirollback()</a> , <a href="#">ocinewdescriptor()</a> , <a href="#">ocirowcount()</a> , <a href="#">ocidefinebyname()</a> , <a href="#">ocibindbyname()</a> , <a href="#">ociexecute()</a> , <a href="#">ocinumcols()</a> , <a href="#">ocirestult()</a> , <a href="#">ocifetch()</a> , <a href="#">ocifetchinto()</a> , <a href="#">ocifetchstatement()</a> , <a href="#">ocicolumnisnull()</a> , <a href="#">ocicolumnname()</a> , <a href="#">ocicolumnsize()</a> , <a href="#">ocicolumntype()</a> , <a href="#">ocistatementtype()</a> , <a href="#">ocierror()</a>	<a href="#">ocifreestatement()</a>	Oracle Cursor
odbc link	<a href="#">odbc_connect()</a>	<a href="#">odbc_autocommit()</a> , <a href="#">odbc_commit()</a> , <a href="#">odbc_error()</a> , <a href="#">odbc_errormsg()</a> , <a href="#">odbc_exec()</a> , <a href="#">odbc_tables()</a> , <a href="#">odbc_tableprivileges()</a> , <a href="#">odbc_do()</a> , <a href="#">odbc_prepare()</a> , <a href="#">odbc_columns()</a> , <a href="#">odbc_columnprivileges()</a> , <a href="#">odbc_procedurecolumns()</a> , <a href="#">odbc_specialcolumns()</a> , <a href="#">odbc_rollback()</a> , <a href="#">odbc_setoption()</a> , <a href="#">odbc_gettypeinfo()</a> , <a href="#">odbc_primarykeys()</a> , <a href="#">odbc_foreignkeys()</a> , <a href="#">odbc_procedures()</a> , <a href="#">odbc_statistics()</a>	<a href="#">odbc_close()</a>	Link to ODBC database

odbc link persistent	<a href="#">odbc_connect()</a>	<a href="#">odbc_autocommit()</a> , <a href="#">odbc_commit()</a> , <a href="#">odbc_error()</a> , <a href="#">odbc_errormsg()</a> , <a href="#">odbc_exec()</a> , <a href="#">odbc_tables()</a> , <a href="#">odbc_tableprivileges()</a> , <a href="#">odbc_do()</a> , <a href="#">odbc_prepare()</a> , <a href="#">odbc_columns()</a> , <a href="#">odbc_columnprivileges()</a> , <a href="#">odbc_procedurecolumns()</a> , <a href="#">odbc_specialcolumns()</a> , <a href="#">odbc_rollback()</a> , <a href="#">odbc_setoption()</a> , <a href="#">odbc_gettypeinfo()</a> , <a href="#">odbc_primarykeys()</a> , <a href="#">odbc_foreignkeys()</a> , <a href="#">odbc_procedures()</a> , <a href="#">odbc_statistics()</a>	None	Persistent link to ODBC database
odbc result	<a href="#">odbc_prepare()</a>	<a href="#">odbc_binmode()</a> , <a href="#">odbc_cursor()</a> , <a href="#">odbc_execute()</a> , <a href="#">odbc_fetch_into()</a> , <a href="#">odbc_fetch_row()</a> , <a href="#">odbc_field_name()</a> , <a href="#">odbc_field_num()</a> , <a href="#">odbc_field_type()</a> , <a href="#">odbc_field_len()</a> , <a href="#">odbc_field_precision()</a> , <a href="#">odbc_field_scale()</a> , <a href="#">odbc_longreadlen()</a> , <a href="#">odbc_num_fields()</a> , <a href="#">odbc_num_rows()</a> , <a href="#">odbc_result()</a> , <a href="#">odbc_result_all()</a> , <a href="#">odbc_setoption()</a>	<a href="#">odbc_free_result()</a>	ODBC result
velocis link				
velocis result				
OpenSSL key	<a href="#">openssl_get_privatekey()</a> , <a href="#">openssl_get_publickey()</a>	<a href="#">openssl_sign()</a> , <a href="#">openssl_seal()</a> , <a href="#">openssl_open()</a> , <a href="#">openssl_verify()</a>	<a href="#">openssl_free_key()</a>	OpenSSL key
OpenSSL X.509	<a href="#">openssl_x509_read()</a>	<a href="#">openssl_x509_parse()</a> , <a href="#">openssl_x509_checkpurpose()</a>	<a href="#">openssl_x509_free()</a>	Public Key
oracle Cursor	<a href="#">ora_open()</a>	<a href="#">ora_bind()</a> , <a href="#">ora_columnname()</a> , <a href="#">ora_columnsizes()</a> , <a href="#">ora_columntype()</a> , <a href="#">ora_error()</a> , <a href="#">ora_errorcode()</a> , <a href="#">ora_exec()</a> , <a href="#">ora_fetch()</a> , <a href="#">ora_fetch_into()</a> , <a href="#">ora_getcolumn()</a> , <a href="#">ora_numcols()</a> , <a href="#">ora_numrows()</a> , <a href="#">ora_parse()</a>	<a href="#">ora_close()</a>	Oracle cursor
oracle link	<a href="#">ora_logon()</a>	<a href="#">ora_do()</a> , <a href="#">ora_error()</a> , <a href="#">ora_errorcode()</a> , <a href="#">ora_rollback()</a> , <a href="#">ora_commitoff()</a> , <a href="#">ora_commiton()</a> , <a href="#">ora_open()</a> , <a href="#">ora_commit()</a>	<a href="#">ora_logoff()</a>	Link to oracle database
oracle link persistent	<a href="#">ora_plogon()</a>	<a href="#">ora_do()</a> , <a href="#">ora_error()</a> , <a href="#">ora_errorcode()</a> , <a href="#">ora_rollback()</a> , <a href="#">ora_commitoff()</a> , <a href="#">ora_commiton()</a> , <a href="#">ora_open()</a> , <a href="#">ora_commit()</a>	None	Persistent link to oracle database

pdf document	<b>pdf_new()</b>	<a href="#">pdf_add_bookmark()</a> , <a href="#">pdf_add_launchlink()</a> , <a href="#">pdf_add_loccallink()</a> , <a href="#">pdf_add_note()</a> , <a href="#">pdf_add_pdflink()</a> , <a href="#">pdf_add_weblink()</a> , <a href="#">pdf_arc()</a> , <a href="#">pdf_attach_file()</a> , <a href="#">pdf_begin_page()</a> , <a href="#">pdf_circle()</a> , <a href="#">pdf_clip()</a> , <a href="#">pdf_closepath()</a> , <a href="#">pdf_closepath_fill_stroke()</a> , <a href="#">pdf_closepath_stroke()</a> , <a href="#">pdf_concat()</a> , <a href="#">pdf_continue_text()</a> , <a href="#">pdf_curveto()</a> , <a href="#">pdf_end_page()</a> , <a href="#">pdf_endpath()</a> , <a href="#">pdf_fill()</a> , <a href="#">pdf_fill_stroke()</a> , <a href="#">pdf_findfont()</a> , <a href="#">pdf_get_buffer()</a> , <a href="#">pdf_get_image_height()</a> , <a href="#">pdf_get_image_width()</a> , <a href="#">pdf_get_parameter()</a> , <a href="#">pdf_get_value()</a> , <a href="#">pdf_lineto()</a> , <a href="#">pdf_moveto()</a> , <a href="#">pdf_open_ccitt()</a> , <a href="#">pdf_open_file()</a> , <a href="#">pdf_open_image_file()</a> , <a href="#">pdf_place_image()</a> , <a href="#">pdf_rect()</a> , <a href="#">pdf_restore()</a> , <a href="#">pdf_rotate()</a> , <a href="#">pdf_save()</a> , <a href="#">pdf_scale()</a> , <a href="#">pdf_setdash()</a> , <a href="#">pdf_setflat()</a> , <a href="#">pdf_setfont()</a> , <a href="#">pdf_setgray()</a> , <a href="#">pdf_setgray_fill()</a> , <a href="#">pdf_setgray_stroke()</a> , <a href="#">pdf_setlinecap()</a> , <a href="#">pdf_setlinejoin()</a> , <a href="#">pdf_setlinewidth()</a> , <a href="#">pdf_setmiterlimit()</a> , <a href="#">pdf_setpolydash()</a> , <a href="#">pdf_setrgbcolor()</a> , <a href="#">pdf_setrgbcolor_fill()</a> , <a href="#">pdf_setrgbcolor_stroke()</a> , <a href="#">pdf_set_border_color()</a> , <a href="#">pdf_set_border_dash()</a> , <a href="#">pdf_set_border_style()</a> , <a href="#">pdf_set_char_spacing()</a> , <a href="#">pdf_set_duration()</a> , <a href="#">pdf_set_font()</a> , <a href="#">pdf_set_horiz_scaling()</a> , <a href="#">pdf_set_parameter()</a> , <a href="#">pdf_set_text_pos()</a> , <a href="#">pdf_set_text_rendering()</a> , <a href="#">pdf_set_value()</a> , <a href="#">pdf_set_word_spacing()</a> , <a href="#">pdf_show()</a> , <a href="#">pdf_show_boxed()</a> , <a href="#">pdf_show_xy()</a> , <a href="#">pdf_skew()</a> , <a href="#">pdf_stringwidth()</a> , <a href="#">pdf_stroke()</a> , <a href="#">pdf_translate()</a> , <a href="#">pdf_open_memory_image()</a>	<a href="#">pdf_close()</a> , <a href="#">pdf_delete()</a>	PDF document
pdf image	<a href="#">pdf_open_image()</a> , <a href="#">pdf_open_image_file()</a> , <a href="#">pdf_open_memory_image()</a>	<a href="#">pdf_get_image_height()</a> , <a href="#">pdf_get_image_width()</a> , <a href="#">pdf_open_CCITT()</a> , <a href="#">pdf_place_image()</a>	<a href="#">pdf_close_image()</a>	Image in PDF file
pdf object				
pdf outline				
pgsql large object	<a href="#">pg_getlastoid()</a> , <a href="#">pg_loimport()</a> , <a href="#">pg_loimport()</a>	<a href="#">pg_loopen()</a> , <a href="#">pg_getlastoid()</a> , <a href="#">pg_locreate()</a> , <a href="#">pg_loexport()</a> , <a href="#">pg_loread()</a> , <a href="#">pg_loreadall()</a> , <a href="#">pg_lounlink()</a> , <a href="#">pg_lowrite()</a>	<a href="#">pg_loclose()</a>	PostgreSQL Large Object

pgsql link	<a href="#">pg_connect()</a>	<a href="#">pg_cmdtuples()</a> , <a href="#">pg_dbname()</a> , <a href="#">pg_end_copy()</a> , <a href="#">pg_errormessage()</a> , <a href="#">pg_host()</a> , <a href="#">pg_locreate()</a> , <a href="#">pg_loexport()</a> , <a href="#">pg_loimport()</a> , <a href="#">pg_loopen()</a> , <a href="#">pg_lounlink()</a> , <a href="#">pg_options()</a> , <a href="#">pg_port()</a> , <a href="#">pg_put_line()</a> , <a href="#">pg_set_client_encoding()</a> , <a href="#">pg_client_encoding()</a> , <a href="#">pg_trace()</a> , <a href="#">pg_untrace()</a> , <a href="#">pg_tty()</a>	<a href="#">pg_close()</a>	Link to PostgreSQL database
pgsql link persistent	<a href="#">pg_pconnect()</a>	<a href="#">pg_cmdtuples()</a> , <a href="#">pg_dbname()</a> , <a href="#">pg_end_copy()</a> , <a href="#">pg_errormessage()</a> , <a href="#">pg_host()</a> , <a href="#">pg_locreate()</a> , <a href="#">pg_loexport()</a> , <a href="#">pg_loimport()</a> , <a href="#">pg_loopen()</a> , <a href="#">pg_lounlink()</a> , <a href="#">pg_options()</a> , <a href="#">pg_port()</a> , <a href="#">pg_put_line()</a> , <a href="#">pg_set_client_encoding()</a> , <a href="#">pg_client_encoding()</a> , <a href="#">pg_trace()</a> , <a href="#">pg_untrace()</a> , <a href="#">pg_tty()</a>	None	Persistent link to PostgreSQL database
pgsql result	<a href="#">pg_exec()</a>	<a href="#">pg_fetch_array()</a> , <a href="#">pg_fetch_object()</a> , <a href="#">pg_fieldisnull()</a> , <a href="#">pg_fetch_row()</a> , <a href="#">pg_fieldname()</a> , <a href="#">pg_fieldnum()</a> , <a href="#">pg_fieldprtlen()</a> , <a href="#">pg_fieldsize()</a> , <a href="#">pg_fieldtype()</a> , <a href="#">pg_getlastoid()</a> , <a href="#">pg_numfields()</a> , <a href="#">pg_result()</a> , <a href="#">pg_numrows()</a>	<a href="#">pg_freeresult()</a>	PostgreSQL result
pgsql string				
printer				
printer brush				
printer font				
printer pen				
pspell	<a href="#">pspell_new()</a> , <a href="#">pspell_new_config()</a> , <a href="#">pspell_new_personal()</a>	<a href="#">pspell_add_to_personal()</a> , <a href="#">pspell_add_to_session()</a> , <a href="#">pspell_check()</a> , <a href="#">pspell_clear_session()</a> , <a href="#">pspell_config_ignore()</a> , <a href="#">pspell_config_mode()</a> , <a href="#">pspell_config_personal()</a> , <a href="#">pspell_config_repl()</a> , <a href="#">pspell_config_runtogether()</a> , <a href="#">pspell_config_save_repl()</a> , <a href="#">pspell_save_wordlist()</a> , <a href="#">pspell_store_replacement()</a> , <a href="#">pspell_suggest()</a>	None	pspell dictionary
pspell config	<a href="#">pspell_config_create()</a>	<a href="#">pspell_new_config()</a>	None	pspell configuration
Sablotron XSLT	<a href="#">xslt_create()</a>	<a href="#">xslt_closelog()</a> , <a href="#">xslt_openlog()</a> , <a href="#">xslt_run()</a> , <a href="#">xslt_set_sax_handler()</a> , <a href="#">xslt_errno()</a> , <a href="#">xslt_error()</a> , <a href="#">xslt_fetch_result()</a> , <a href="#">xslt_free()</a>	<a href="#">xslt_free()</a>	XSLT parser
shmop	<a href="#">shmop_open()</a>	<a href="#">shmop_read()</a> , <a href="#">shmop_write()</a> , <a href="#">shmop_size()</a> , <a href="#">shmop_delete()</a>	<a href="#">shmop_close()</a>	
sockets file descriptor set	<a href="#">socket()</a>	<a href="#">accept_connect()</a> , <a href="#">bind()</a> , <a href="#">connect()</a> , <a href="#">listen()</a> , <a href="#">read()</a> , <a href="#">write()</a>	<a href="#">close()</a>	Socket
sockets i/o vector				

dir	<a href="#">dir()</a>	<a href="#">readdir()</a> , <a href="#">rewinddir()</a>	<a href="#">closedir()</a>	Dir handle
file	<a href="#">fopen()</a>	<a href="#">feof()</a> , <a href="#">fflush()</a> , <a href="#">fgetc()</a> , <a href="#">fgetcsv()</a> , <a href="#">fgets()</a> , <a href="#">fgetss()</a> , <a href="#">flock()</a> , <a href="#">fpassthru()</a> , <a href="#">fputs()</a> , <a href="#">fwrite()</a> , <a href="#">fread()</a> , <a href="#">fseek()</a> , <a href="#">ftell()</a> , <a href="#">fstat()</a> , <a href="#">ftruncate()</a> , <a href="#">set_file_buffer()</a> , <a href="#">rewind()</a>	<a href="#">fclose()</a>	File handle
pipe	<a href="#">popen()</a>	<a href="#">feof()</a> , <a href="#">fflush()</a> , <a href="#">fgetc()</a> , <a href="#">fgetcsv()</a> , <a href="#">fgets()</a> , <a href="#">fgetss()</a> , <a href="#">fpassthru()</a> , <a href="#">fputs()</a> , <a href="#">fwrite()</a> , <a href="#">fread()</a>	<a href="#">pclose()</a>	Process handle
socket	<a href="#">fsockopen()</a>	<a href="#">fflush()</a> , <a href="#">fgetc()</a> , <a href="#">fgetcsv()</a> , <a href="#">fgets()</a> , <a href="#">fgetss()</a> , <a href="#">fpassthru()</a> , <a href="#">fputs()</a> , <a href="#">fwrite()</a> , <a href="#">fread()</a>	<a href="#">fclose()</a>	Socket handle
stream				
sybase-db link	<a href="#">sybase_connect()</a>	<a href="#">sybase_query()</a> , <a href="#">sybase_select_db()</a>	<a href="#">sybase_close()</a>	Link to Sybase database using DB library
sybase-db link persistent	<a href="#">sybase_pconnect()</a>	<a href="#">sybase_query()</a> , <a href="#">sybase_select_db()</a>	None	Persistent link to Sybase database using DB library
sybase-db result	<a href="#">sybase_query()</a>	<a href="#">sybase_data_seek()</a> , <a href="#">sybase_fetch_array()</a> , <a href="#">sybase_fetch_field()</a> , <a href="#">sybase_fetch_object()</a> , <a href="#">sybase_fetch_row()</a> , <a href="#">sybase_field_seek()</a> , <a href="#">sybase_num_fields()</a> , <a href="#">sybase_num_rows()</a> , <a href="#">sybase_result()</a>	<a href="#">sybase_free_result()</a>	Sybase result using DB library
sybase-ct link	<a href="#">sybase_connect()</a>	<a href="#">sybase_affected_rows()</a> , <a href="#">sybase_query()</a> , <a href="#">sybase_select_db()</a>	<a href="#">sybase_close()</a>	Link to Sybase database using CT library
sybase-ct link persistent	<a href="#">sybase_pconnect()</a>	<a href="#">sybase_affected_rows()</a> , <a href="#">sybase_query()</a> , <a href="#">sybase_select_db()</a>	None	Persistent link to Sybase database using CT library
sybase-ct result	<a href="#">sybase_query()</a>	<a href="#">sybase_data_seek()</a> , <a href="#">sybase_fetch_array()</a> , <a href="#">sybase_fetch_field()</a> , <a href="#">sybase_fetch_object()</a> , <a href="#">sybase_fetch_row()</a> , <a href="#">sybase_field_seek()</a> , <a href="#">sybase_num_fields()</a> , <a href="#">sybase_num_rows()</a> , <a href="#">sybase_result()</a>	<a href="#">sybase_free_result()</a>	Sybase result using CT library
sysvsem	<a href="#">sem_get()</a>	<a href="#">sem_acquire()</a>	<a href="#">sem_release()</a>	System V Semaphore
sysvshm	<a href="#">shm_attach()</a>	<a href="#">shm_remove()</a> , <a href="#">shm_put_var()</a> , <a href="#">shm_get_var()</a> , <a href="#">shm_remove_var()</a>	<a href="#">shm_detach()</a>	System V Shared Memory
wddx	<a href="#">wddx_packet_start()</a>	<a href="#">wddx_add_vars()</a>	<a href="#">wddx_packet_end()</a>	WDDX packet



xml	<a href="#">xml_parser_create()</a>	<a href="#">xml_set_object()</a> , <a href="#">xml_set_element_handler()</a> , <a href="#">xml_set_character_data_handler()</a> , <a href="#">xml_set_processing_instruction_handler()</a> , <a href="#">xml_set_default_handler()</a> , <a href="#">xml_set_unparsed_entity_decl_handler()</a> , <a href="#">xml_set_notation_decl_handler()</a> , <a href="#">xml_set_external_entity_ref_handler()</a> , <a href="#">xml_parse()</a> , <a href="#">xml_get_error_code()</a> , <a href="#">xml_error_string()</a> , <a href="#">xml_get_current_line_number()</a> , <a href="#">xml_get_current_column_number()</a> , <a href="#">xml_get_current_byte_index()</a> , <a href="#">xml_parse_into_struct()</a> , <a href="#">xml_parser_set_option()</a> , <a href="#">xml_parser_get_option()</a>	<a href="#">xml_parser_free()</a>	XML parser
zlib	<a href="#">gzopen()</a>	<a href="#">gzeof()</a> , <a href="#">gzgetc()</a> , <a href="#">gzgets()</a> , <a href="#">gzgetss()</a> , <a href="#">gzpassthru()</a> , <a href="#">gzputs()</a> , <a href="#">gzread()</a> , <a href="#">gzrewind()</a> , <a href="#">gzseek()</a> , <a href="#">gztell()</a> , <a href="#">gzwrite()</a>	<a href="#">gzclose()</a>	gz-compressed file

[Anterior](#)

PHP reserved words

[Inicio](#)

[Subir](#)

[Siguiente](#)

Aliases list

# Apéndice G. Aliases list

Here is the aliases list. All aliases are listed here. It is usually a bad idea to use aliases, as they may be bound to obsolescence or renaming, which will lead to unportable script. This list is provided to help those who want to upgrade their old scripts to newer syntax.

However, some functions simply have two names, and there is no real preference. (For example, [is\\_int\(\)](#) and [is\\_integer\(\)](#) are equally good)

This list is consistent with PHP 4.0.6.

**Tabla G-1. Aliases**

Alias	Master function	Extension used
add	<a href="#">swfmovie_add()</a>	<a href="#">Ming (flash)</a>
add	<a href="#">swfsprite_add()</a>	<a href="#">Ming (flash)</a>
add_root	<a href="#">domxml_add_root()</a>	<a href="#">DOM XML</a>
addaction	<a href="#">swfbutton_addAction()</a>	<a href="#">Ming (flash)</a>
addcolor	<a href="#">swfdisplayitem_addColor()</a>	<a href="#">Ming (flash)</a>
addentry	<a href="#">swfgradient_addEntry()</a>	<a href="#">Ming (flash)</a>
addfill	<a href="#">swfshape_addfill()</a>	<a href="#">Ming (flash)</a>
addshape	<a href="#">swfbutton_addShape()</a>	<a href="#">Ming (flash)</a>
addstring	<a href="#">swftext_addString()</a>	<a href="#">Ming (flash)</a>
addstring	<a href="#">swftextfield_addString()</a>	<a href="#">Ming (flash)</a>
align	<a href="#">swftextfield_align()</a>	<a href="#">Ming (flash)</a>
attributes	<a href="#">domxml_attributes()</a>	<a href="#">DOM XML</a>
children	<a href="#">domxml_children()</a>	<a href="#">DOM XML</a>

chop	<a href="#"><u>rtrim()</u></a>	Base syntax
close	<a href="#"><u>closedir()</u></a>	Base syntax
com_get	<a href="#"><u>com_propget()</u></a>	<a href="#"><u>COM</u></a>
com_propset	<a href="#"><u>com_propput()</u></a>	<a href="#"><u>COM</u></a>
com_set	<a href="#"><u>com_propput()</u></a>	<a href="#"><u>COM</u></a>
cv_add	<a href="#"><u>ccvs_add()</u></a>	<a href="#"><u>CCVS</u></a>
cv_auth	<a href="#"><u>ccvs_auth()</u></a>	<a href="#"><u>CCVS</u></a>
cv_command	<a href="#"><u>ccvs_command()</u></a>	<a href="#"><u>CCVS</u></a>
cv_count	<a href="#"><u>ccvs_count()</u></a>	<a href="#"><u>CCVS</u></a>
cv_delete	<a href="#"><u>ccvs_delete()</u></a>	<a href="#"><u>CCVS</u></a>
cv_done	<a href="#"><u>ccvs_done()</u></a>	<a href="#"><u>CCVS</u></a>
cv_init	<a href="#"><u>ccvs_init()</u></a>	<a href="#"><u>CCVS</u></a>
cv_lookup	<a href="#"><u>ccvs_lookup()</u></a>	<a href="#"><u>CCVS</u></a>
cv_new	<a href="#"><u>ccvs_new()</u></a>	<a href="#"><u>CCVS</u></a>
cv_report	<a href="#"><u>ccvs_report()</u></a>	<a href="#"><u>CCVS</u></a>
cv_return	<a href="#"><u>ccvs_return()</u></a>	<a href="#"><u>CCVS</u></a>
cv_reverse	<a href="#"><u>ccvs_reverse()</u></a>	<a href="#"><u>CCVS</u></a>
cv_sale	<a href="#"><u>ccvs_sale()</u></a>	<a href="#"><u>CCVS</u></a>
cv_status	<a href="#"><u>ccvs_status()</u></a>	<a href="#"><u>CCVS</u></a>
cv_textvalue	<a href="#"><u>ccvs_textvalue()</u></a>	<a href="#"><u>CCVS</u></a>
cv_void	<a href="#"><u>ccvs_void()</u></a>	<a href="#"><u>CCVS</u></a>
die	<a href="#"><u>exit()</u></a>	<a href="#"><u>Miscellaneous functions</u></a>
dir	<a href="#"><u>getdir()</u></a>	Base syntax
diskfreespace	<a href="#"><u>disk_free_space()</u></a>	<a href="#"><u>Filesystem</u></a>
domxml_getattr	<a href="#"><u>domxml_get_attribute()</u></a>	<a href="#"><u>DOM XML</u></a>
domxml_setattr	<a href="#"><u>domxml_set_attribute()</u></a>	<a href="#"><u>DOM XML</u></a>
doubleval	<a href="#"><u>floatval()</u></a>	Base syntax
drawarc	<a href="#"><u>swfshape_drawarc()</u></a>	<a href="#"><u>Ming (flash)</u></a>

drawcircle	swfshape_drawcircle()	<a href="#">Ming (flash)</a>
drawcubic	swfshape_drawcubic()	<a href="#">Ming (flash)</a>
drawcubicto	swfshape_drawcubicto()	<a href="#">Ming (flash)</a>
drawcurve	swfshape_drawcurve()	<a href="#">Ming (flash)</a>
drawcurveto	swfshape_drawcurveto()	<a href="#">Ming (flash)</a>
drawglyph	swfshape_drawglyph()	<a href="#">Ming (flash)</a>
drawline	swfshape_drawline()	<a href="#">Ming (flash)</a>
drawlineto	swfshape_drawlineto()	<a href="#">Ming (flash)</a>
dtd	domxml_intdtd()	<a href="#">DOM XML</a>
dumpmem	domxml_dumpmem()	<a href="#">DOM XML</a>
fbsql	<a href="#">fbsql_db_query()</a>	<a href="#">FrontBase</a>
fputs	<a href="#">fwrite()</a>	Base syntax
get_attribute	domxml_get_attribute()	<a href="#">DOM XML</a>
getascent	swffont_getAscent()	<a href="#">Ming (flash)</a>
getascent	swftext_getAscent()	<a href="#">Ming (flash)</a>
getattr	domxml_get_attribute()	<a href="#">DOM XML</a>
getdescent	swffont_getDescent()	<a href="#">Ming (flash)</a>
getdescent	swftext_getDescent()	<a href="#">Ming (flash)</a>
getheight	swfbitmap_getHeight()	<a href="#">Ming (flash)</a>
getleading	swffont_getLeading()	<a href="#">Ming (flash)</a>
getleading	swftext_getLeading()	<a href="#">Ming (flash)</a>
getshape1	swfmorph_getShape1()	<a href="#">Ming (flash)</a>
getshape2	swfmorph_getShape2()	<a href="#">Ming (flash)</a>
getwidth	swfbitmap_getWidth()	<a href="#">Ming (flash)</a>
getwidth	swffont_getWidth()	<a href="#">Ming (flash)</a>
getwidth	swftext_getWidth()	<a href="#">Ming (flash)</a>
gzputs	<a href="#">gzwrite()</a>	<a href="#">Zlib</a>

i18n_convert	<a href="#"><u>mb_convert_encoding()</u></a>	<a href="#"><u>Multi-bytes Strings</u></a>
i18n_discover_encoding	<a href="#"><u>mb_detect_encoding()</u></a>	<a href="#"><u>Multi-bytes Strings</u></a>
i18n_http_input	<a href="#"><u>mb_http_input()</u></a>	<a href="#"><u>Multi-bytes Strings</u></a>
i18n_http_output	<a href="#"><u>mb_http_output()</u></a>	<a href="#"><u>Multi-bytes Strings</u></a>
i18n_internal_encoding	<a href="#"><u>mb_internal_encoding()</u></a>	<a href="#"><u>Multi-bytes Strings</u></a>
i18n_ja_jp_hantozen	<a href="#"><u>mb_convert_kana()</u></a>	<a href="#"><u>Multi-bytes Strings</u></a>
i18n_mime_header_decode	<a href="#"><u>mb_decode_mimeheader()</u></a>	<a href="#"><u>Multi-bytes Strings</u></a>
i18n_mime_header_encode	<a href="#"><u>mb_encode_mimeheader()</u></a>	<a href="#"><u>Multi-bytes Strings</u></a>
imap_create	<a href="#"><u>imap_createmailbox()</u></a>	<a href="#"><u>IMAP</u></a>
imap_fetchtext	<a href="#"><u>imap_body()</u></a>	<a href="#"><u>IMAP</u></a>
imap_getmailboxes	<a href="#"><u>imap_list_full()</u></a>	<a href="#"><u>IMAP</u></a>
imap_getsubscribed	<a href="#"><u>imap_lsub_full()</u></a>	<a href="#"><u>IMAP</u></a>
imap_header	<a href="#"><u>imap_headerinfo()</u></a>	<a href="#"><u>IMAP</u></a>
imap_listmailbox	<a href="#"><u>imap_list()</u></a>	<a href="#"><u>IMAP</u></a>
imap_listsubscribed	<a href="#"><u>imap_lsub()</u></a>	<a href="#"><u>IMAP</u></a>
imap_rename	<a href="#"><u>imap_renamemailbox()</u></a>	<a href="#"><u>IMAP</u></a>
imap_scan	<a href="#"><u>imap_listscan()</u></a>	<a href="#"><u>IMAP</u></a>
imap_scanmailbox	<a href="#"><u>imap_listscan()</u></a>	<a href="#"><u>IMAP</u></a>
ini_alter	<a href="#"><u>ini_set()</u></a>	<a href="#"><u>Base syntax</u></a>
is_double	<a href="#"><u>is_float()</u></a>	<a href="#"><u>Base syntax</u></a>
is_integer	<a href="#"><u>is_int()</u></a>	<a href="#"><u>Base syntax</u></a>
is_long	<a href="#"><u>is_int()</u></a>	<a href="#"><u>Base syntax</u></a>
is_real	<a href="#"><u>is_float()</u></a>	<a href="#"><u>Base syntax</u></a>
is_writeable	<a href="#"><u>is_writable()</u></a>	<a href="#"><u>Base syntax</u></a>
join	<a href="#"><u>implode()</u></a>	<a href="#"><u>Base syntax</u></a>
labelframe	<a href="#"><u>swfmovie_labelFrame()</u></a>	<a href="#"><u>Ming (flash)</u></a>
labelframe	<a href="#"><u>swfsprite_labelFrame()</u></a>	<a href="#"><u>Ming (flash)</u></a>
last_child	<a href="#"><u>domxml_last_child()</u></a>	<a href="#"><u>DOM XML</u></a>

lastchild	<a href="#"><b>domxml_last_child()</b></a>	<a href="#">DOM XML</a>
ldap_close	<a href="#"><b>ldap_unbind()</b></a>	<a href="#">LDAP</a>
magic_quotes_runtime	<a href="#"><b>set_magic_quotes_runtime()</b></a>	Base syntax
mbstrcut	<a href="#"><b>mb_strcut()</b></a>	<a href="#">Multi-bytes Strings</a>
mbstrlen	<a href="#"><b>mb_strlen()</b></a>	<a href="#">Multi-bytes Strings</a>
mbstrpos	<a href="#"><b>mb_strpos()</b></a>	<a href="#">Multi-bytes Strings</a>
mbstrrpos	<a href="#"><b>mb_strrpos()</b></a>	<a href="#">Multi-bytes Strings</a>
mbsubstr	<a href="#"><b>mb_substr()</b></a>	<a href="#">Multi-bytes Strings</a>
ming_setcubicthreshold	<a href="#"><b>ming setCubicThreshold()</b></a>	<a href="#">Ming (flash)</a>
ming_setscale	<a href="#"><b>ming setScale()</b></a>	<a href="#">Ming (flash)</a>
move	<a href="#"><b>swfdisplayitem_move()</b></a>	<a href="#">Ming (flash)</a>
movepen	<a href="#"><b>swfshape_movepen()</b></a>	<a href="#">Ming (flash)</a>
movepento	<a href="#"><b>swfshape_movepento()</b></a>	<a href="#">Ming (flash)</a>
moveto	<a href="#"><b>swfdisplayitem_moveTo()</b></a>	<a href="#">Ming (flash)</a>
moveto	<a href="#"><b>swfill_moveTo()</b></a>	<a href="#">Ming (flash)</a>
moveto	<a href="#"><b>swftext_moveTo()</b></a>	<a href="#">Ming (flash)</a>
mysql	<a href="#"><b>mysql_db_query()</b></a>	<a href="#">mSQL</a>
mysql_affected_rows	<a href="#"><b>mysql_affected_rows()</b></a>	<a href="#">mSQL</a>
mysql_createdb	<a href="#"><b>mysql_create_db()</b></a>	<a href="#">mSQL</a>
mysql_dbname	<a href="#"><b>mysql_result()</b></a>	<a href="#">mSQL</a>
mysql_dropdb	<a href="#"><b>mysql_drop_db()</b></a>	<a href="#">mSQL</a>
mysql_fieldflags	<a href="#"><b>mysql_field_flags()</b></a>	<a href="#">mSQL</a>
mysql_fieldlen	<a href="#"><b>mysql_field_len()</b></a>	<a href="#">mSQL</a>
mysql_fieldname	<a href="#"><b>mysql_field_name()</b></a>	<a href="#">mSQL</a>
mysql_fieldtable	<a href="#"><b>mysql_field_table()</b></a>	<a href="#">mSQL</a>
mysql_fieldtype	<a href="#"><b>mysql_field_type()</b></a>	<a href="#">mSQL</a>
mysql_freeresult	<a href="#"><b>mysql_free_result()</b></a>	<a href="#">mSQL</a>

mysql_listdbs	<a href="#"><u>mysql_list_dbs()</u></a>	<a href="#"><u>mSQL</u></a>
mysql_listfields	<a href="#"><u>mysql_list_fields()</u></a>	<a href="#"><u>mSQL</u></a>
mysql_listtables	<a href="#"><u>mysql_list_tables()</u></a>	<a href="#"><u>mSQL</u></a>
mysql_numfields	<a href="#"><u>mysql_num_fields()</u></a>	<a href="#"><u>mSQL</u></a>
mysql_numrows	<a href="#"><u>mysql_num_rows()</u></a>	<a href="#"><u>mSQL</u></a>
mysql_regcase	<a href="#"><u>sql_regcase()</u></a>	<a href="#"><u>mSQL</u></a>
mysql_selectdb	<a href="#"><u>mysql_select_db()</u></a>	<a href="#"><u>mSQL</u></a>
mysql_tablename	<a href="#"><u>mysql_result()</u></a>	<a href="#"><u>mSQL</u></a>
mssql_affected_rows	<a href="#"><u>sybase_affected_rows()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_affected_rows	<a href="#"><u>sybase_affected_rows()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_close	<a href="#"><u>sybase_close()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_close	<a href="#"><u>sybase_close()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_connect	<a href="#"><u>sybase_connect()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_connect	<a href="#"><u>sybase_connect()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_data_seek	<a href="#"><u>sybase_data_seek()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_data_seek	<a href="#"><u>sybase_data_seek()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_fetch_array	<a href="#"><u>sybase_fetch_array()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_fetch_array	<a href="#"><u>sybase_fetch_array()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_fetch_field	<a href="#"><u>sybase_fetch_field()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_fetch_field	<a href="#"><u>sybase_fetch_field()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_fetch_object	<a href="#"><u>sybase_fetch_object()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_fetch_object	<a href="#"><u>sybase_fetch_object()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_fetch_row	<a href="#"><u>sybase_fetch_row()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_fetch_row	<a href="#"><u>sybase_fetch_row()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_field_seek	<a href="#"><u>sybase_field_seek()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_field_seek	<a href="#"><u>sybase_field_seek()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_free_result	<a href="#"><u>sybase_free_result()</u></a>	<a href="#"><u>Sybase</u></a>

mssql_free_result	<a href="#"><u>sybase_free_result()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_get_last_message	<a href="#"><u>sybase_get_last_message()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_get_last_message	<a href="#"><u>sybase_get_last_message()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_min_client_severity	<a href="#"><u>sybase_min_client_severity()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_min_error_severity	<a href="#"><u>sybase_min_error_severity()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_min_message_severity	<a href="#"><u>sybase_min_message_severity()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_min_server_severity	<a href="#"><u>sybase_min_server_severity()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_num_fields	<a href="#"><u>sybase_num_fields()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_num_fields	<a href="#"><u>sybase_num_fields()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_num_rows	<a href="#"><u>sybase_num_rows()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_num_rows	<a href="#"><u>sybase_num_rows()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_pconnect	<a href="#"><u>sybase_pconnect()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_pconnect	<a href="#"><u>sybase_pconnect()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_query	<a href="#"><u>sybase_query()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_query	<a href="#"><u>sybase_query()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_result	<a href="#"><u>sybase_result()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_result	<a href="#"><u>sybase_result()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_select_db	<a href="#"><u>sybase_select_db()</u></a>	<a href="#"><u>Sybase</u></a>
mssql_select_db	<a href="#"><u>sybase_select_db()</u></a>	<a href="#"><u>Sybase</u></a>
multicolor	<a href="#"><u>swfdisplayitem_multColor()</u></a>	<a href="#"><u>Ming (flash)</u></a>
mysql	<a href="#"><u>mysql_db_query()</u></a>	<a href="#"><u>MySQL</u></a>
mysql_createdb	<a href="#"><u>mysql_create_db()</u></a>	<a href="#"><u>MySQL</u></a>
mysql_db_name	<a href="#"><u>mysql_result()</u></a>	<a href="#"><u>MySQL</u></a>
mysql_dbname	<a href="#"><u>mysql_result()</u></a>	<a href="#"><u>MySQL</u></a>
mysql_dropdb	<a href="#"><u>mysql_drop_db()</u></a>	<a href="#"><u>MySQL</u></a>
mysql_fieldflags	<a href="#"><u>mysql_field_flags()</u></a>	<a href="#"><u>MySQL</u></a>
mysql_fieldlen	<a href="#"><u>mysql_field_len()</u></a>	<a href="#"><u>MySQL</u></a>



mysql_fieldname	<a href="#"><u>mysql_field_name()</u></a>	<a href="#"><u>MySQL</u></a>
mysql_fieldtable	<a href="#"><u>mysql_field_table()</u></a>	<a href="#"><u>MySQL</u></a>
mysql_fieldtype	<a href="#"><u>mysql_field_type()</u></a>	<a href="#"><u>MySQL</u></a>
mysql_freeresult	<a href="#"><u>mysql_free_result()</u></a>	<a href="#"><u>MySQL</u></a>
mysql_listdbs	<a href="#"><u>mysql_list_dbs()</u></a>	<a href="#"><u>MySQL</u></a>
mysql_listfields	<a href="#"><u>mysql_list_fields()</u></a>	<a href="#"><u>MySQL</u></a>
mysql_listtables	<a href="#"><u>mysql_list_tables()</u></a>	<a href="#"><u>MySQL</u></a>
mysql_numfields	<a href="#"><u>mysql_num_fields()</u></a>	<a href="#"><u>MySQL</u></a>
mysql_numrows	<a href="#"><u>mysql_num_rows()</u></a>	<a href="#"><u>MySQL</u></a>
mysql_selectdb	<a href="#"><u>mysql_select_db()</u></a>	<a href="#"><u>MySQL</u></a>
mysql_tablename	<a href="#"><u>mysql_result()</u></a>	<a href="#"><u>MySQL</u></a>
name	<b>domxml_attrname()</b>	<a href="#"><u>DOM XML</u></a>
new_child	<b>domxml_new_child()</b>	<a href="#"><u>DOM XML</u></a>
new_xmldoc	<b>domxml_new_xmldoc()</b>	<a href="#"><u>DOM XML</u></a>
nextframe	<b>swfmovie_nextFrame()</b>	<a href="#"><u>Ming (flash)</u></a>
nextframe	<b>swfsprite_nextFrame()</b>	<a href="#"><u>Ming (flash)</u></a>
node	<b>domxml_node()</b>	<a href="#"><u>DOM XML</u></a>
oci8append	<b>ocicollappend()</b>	<a href="#"><u>OCI8</u></a>
oci8assign	<b>ocicollassign()</b>	<a href="#"><u>OCI8</u></a>
oci8assignelem	<b>ocicollassignelem()</b>	<a href="#"><u>OCI8</u></a>
oci8close	<b>ocicloselob()</b>	<a href="#"><u>OCI8</u></a>
oci8free	<b>ocifreecoll()</b>	<a href="#"><u>OCI8</u></a>
oci8free	<b>ocifreedesc()</b>	<a href="#"><u>OCI8</u></a>
oci8getelem	<b>ocicollgetelem()</b>	<a href="#"><u>OCI8</u></a>
oci8load	<b>ociloadlob()</b>	<a href="#"><u>OCI8</u></a>
oci8max	<b>ocicollmax()</b>	<a href="#"><u>OCI8</u></a>
oci8ocifreecursor	<a href="#"><u>ocifreestatement()</u></a>	<a href="#"><u>OCI8</u></a>

oci8save	<b>ocisavelob()</b>	<a href="#">OCI8</a>
oci8savefile	<b>ocisavelobfile()</b>	<a href="#">OCI8</a>
oci8size	<b>ocicollsize()</b>	<a href="#">OCI8</a>
oci8trim	<b>ocicolltrim()</b>	<a href="#">OCI8</a>
oci8writetemporary	<b>ociwritetemporarylob()</b>	<a href="#">OCI8</a>
oci8writetofile	<b>ociwritelobtofile()</b>	<a href="#">OCI8</a>
odbc_do	<a href="#">odbc_exec()</a>	<a href="#">OCI8</a>
odbc_field_precision	<a href="#">odbc_field_len()</a>	<a href="#">OCI8</a>
orbit_caught_exception	<b>satellite_caught_exception()</b>	<a href="#">Satellite</a>
orbit_exception_id	<b>satellite_exception_id()</b>	<a href="#">Satellite</a>
orbit_exception_value	<b>satellite_exception_value()</b>	<a href="#">Satellite</a>
orbit_get_repository_id	<b>satellite_get_repository_id()</b>	<a href="#">Satellite</a>
orbit_load_idl	<b>satellite_load_idl()</b>	<a href="#">Satellite</a>
output	<b>swfmovie_output()</b>	<a href="#">Ming (flash)</a>
parent	<b>domxml_parent()</b>	<a href="#">DOM XML</a>
pdf_add_outline	<b>pdf_add_bookmark()</b>	<a href="#">PDF</a>
pg_clientencoding	<b>pg_client_encoding()</b>	<a href="#">PostgreSQL</a>
pg_setclientencoding	<b>pg_set_client_encoding()</b>	<a href="#">PostgreSQL</a>
pos	<a href="#">current()</a>	Base syntax
recode	<a href="#">recode_string()</a>	<a href="#">Recode</a>
remove	<b>swfmovie_remove()</b>	<a href="#">Ming (flash)</a>
remove	<b>swfsprite_remove()</b>	<a href="#">Ming (flash)</a>
rewind	<a href="#">rewinddir()</a>	Base syntax
root	<b>domxml_root()</b>	<a href="#">DOM XML</a>
rotate	<b>swfdisplayitem_rotate()</b>	<a href="#">Ming (flash)</a>
rotateto	<b>swfdisplayitem_rotateTo()</b>	<a href="#">Ming (flash)</a>
rotateto	<b>swffill_rotateTo()</b>	<a href="#">Ming (flash)</a>

save	<b>swfmovie_save()</b>	<a href="#">Ming (flash)</a>
savetofile	<b>swfmovie_saveToFile()</b>	<a href="#">Ming (flash)</a>
scale	<b>swfdisplayitem_scale()</b>	<a href="#">Ming (flash)</a>
scalet	<b>swfdisplayitem_scaleTo()</b>	<a href="#">Ming (flash)</a>
scalet	<b>swffill_scaleTo()</b>	<a href="#">Ming (flash)</a>
set_attribute	<b>domxml_set_attribute()</b>	<a href="#">DOM XML</a>
set_content	<b>domxml_set_content()</b>	<a href="#">DOM XML</a>
setaction	<b>swfbutton_setAction()</b>	<a href="#">Ming (flash)</a>
setattr	<b>domxml_set_attribute()</b>	<a href="#">DOM XML</a>
setbackground	<b>swfmovie_setBackground()</b>	<a href="#">Ming (flash)</a>
setbounds	<b>swftextfield_setBounds()</b>	<a href="#">Ming (flash)</a>
setcolor	<b>swftext_setColor()</b>	<a href="#">Ming (flash)</a>
setcolor	<b>swftextfield_setColor()</b>	<a href="#">Ming (flash)</a>
setdepth	<b>swfdisplayitem_setDepth()</b>	<a href="#">Ming (flash)</a>
setdimension	<b>swfmovie_setDimension()</b>	<a href="#">Ming (flash)</a>
setdown	<b>swfbutton_setDown()</b>	<a href="#">Ming (flash)</a>
setfont	<b>swftext_setFont()</b>	<a href="#">Ming (flash)</a>
setfont	<b>swftextfield_setFont()</b>	<a href="#">Ming (flash)</a>
setframes	<b>swfmovie_setFrames()</b>	<a href="#">Ming (flash)</a>
setframes	<b>swfsprite_setFrames()</b>	<a href="#">Ming (flash)</a>
setheight	<b>swftext_setHeight()</b>	<a href="#">Ming (flash)</a>
setheight	<b>swftextfield_setHeight()</b>	<a href="#">Ming (flash)</a>
sethit	<b>swfbutton_setHit()</b>	<a href="#">Ming (flash)</a>
setindentation	<b>swftextfield_setIndentation()</b>	<a href="#">Ming (flash)</a>
setleftfill	<b>swfshape_setleftfill()</b>	<a href="#">Ming (flash)</a>
setleftmargin	<b>swftextfield_setLeftMargin()</b>	<a href="#">Ming (flash)</a>
setline	<b>swfshape_setline()</b>	<a href="#">Ming (flash)</a>

setlinespacing	swftextfield_setLineSpacing()	<a href="#">Ming (flash)</a>
setmargins	swftextfield_setMargins()	<a href="#">Ming (flash)</a>
setmatrix	swfdisplayitem_setMatrix()	<a href="#">Ming (flash)</a>
setname	swfdisplayitem_setName()	<a href="#">Ming (flash)</a>
setname	swftextfield_setName()	<a href="#">Ming (flash)</a>
setover	swfbutton_setOver()	<a href="#">Ming (flash)</a>
setrate	swfmovie_setRate()	<a href="#">Ming (flash)</a>
setratio	swfdisplayitem_setRatio()	<a href="#">Ming (flash)</a>
setrightfill	swfshape_setrightfill()	<a href="#">Ming (flash)</a>
setrightmargin	swftextfield_setRightMargin()	<a href="#">Ming (flash)</a>
setspacing	swftext_setSpacing()	<a href="#">Ming (flash)</a>
setup	swfbutton_setUp()	<a href="#">Ming (flash)</a>
show_source	<b>highlight_file ()</b>	Base syntax
sizeof	<a href="#"><u>count()</u></a>	Base syntax
skewx	swfdisplayitem_skewX()	<a href="#">Ming (flash)</a>
skewxt0	swfdisplayitem_skewXTo()	<a href="#">Ming (flash)</a>
skewxt0	swffill_skewXTo()	<a href="#">Ming (flash)</a>
skewy	swfdisplayitem_skewY()	<a href="#">Ming (flash)</a>
skewyt0	swfdisplayitem_skewYTo()	<a href="#">Ming (flash)</a>
skewyt0	swffill_skewYTo()	<a href="#">Ming (flash)</a>
snmpwalkoid	snmprealwalk()	<a href="#">SNMP</a>
strchr	<a href="#"><u>strstr()</u></a>	Base syntax
streammp3	swfmovie_streamMp3()	<a href="#">Ming (flash)</a>
swfaction	swfaction_init()	<a href="#">Ming (flash)</a>
swfbitmap	swfbitmap_init()	<a href="#">Ming (flash)</a>
swfbutton	swfbutton_init()	<a href="#">Ming (flash)</a>
swfbutton_keypress	swfbutton_keypress()	<a href="#">Ming (flash)</a>

swffill	swffill_init()	<a href="#">Ming (flash)</a>
swffont	swffont_init()	<a href="#">Ming (flash)</a>
swfgradient	swfgradient_init()	<a href="#">Ming (flash)</a>
swfmorph	swfmorph_init()	<a href="#">Ming (flash)</a>
swfmovie	swfmovie_init()	<a href="#">Ming (flash)</a>
swfshape	swfshape_init()	<a href="#">Ming (flash)</a>
swfsprite	swfsprite_init()	<a href="#">Ming (flash)</a>
swftext	swftext_init()	<a href="#">Ming (flash)</a>
swftextfield	swftextfield_init()	<a href="#">Ming (flash)</a>
unlink	domxml_unlink_node()	<a href="#">DOM XML</a>
xpath_eval	xpath_eval()	<a href="#">DOM XML</a>
xpath_eval_expression	xpath_eval_expression()	<a href="#">DOM XML</a>
xpath_init	xpath_init()	<a href="#">DOM XML</a>
xpath_new_context	xpath_new_context()	<a href="#">DOM XML</a>
xptr_new_context	xpath_new_context()	<a href="#">DOM XML</a>

---

[Anterior](#)

PHP Resource Types

[Inicio](#)

[Subir](#)

[Siguiete](#)

Using PHP from the command  
line

# Apéndice H. Using PHP from the command line

The command line options of the PHP executable are useful if you would like to debug or test your PHP setup, but they can also be handy, if you would like to use PHP for a different purpose than web scripting.

Note, that you can always direct the output of the PHP executable to an external file with the > character, so `php -q test.php > test.html` will print out the output of `test.php` without HTTP headers to the `test.html` file in the same directory.

You can only use these command line options if you have the PHP executable. If you built the server module version, and you have no CGI version available on your machine, than you have no chance to use these options. For Windows users both the PHP executable and the server modules are in the binary package, the executable is named `php.exe`.

This list of command line options is consistent with PHP 4.0.6. You can get the actual list and some one line descriptions with the `-h` option. The output of `php -h` should be something like this: Usage: `php [-q] [-h] [-s [-v] [-i] [-f <file>] | {<file> [args...]} -q Quiet-mode. Suppress HTTP Header output. -s Display colour syntax highlighted source. -f <file> Parse <file>. Implies '-q' -v Version number -C Do not chdir to the script's directory -c <path> Look for php.ini file in this directory -d foo[=bar] Define INI entry foo with value 'bar' -e Generate extended information for debugger/profiler -z <file> Load Zend extension <file>. -l Syntax check only (lint) -m Show compiled in modules -i PHP information -h This help`

Here we list some of the most important command line options with detailed explanations.

**Tabla H-1. Command line options**

Option	Description
-q	Suppress HTTP headers output. Normally PHP prints out HTTP headers for the calling program (ie. webserver) to hand on to the browser. When writing command line applications these headers are useless.
-s	Display the color highlighted source of the file given with its name. This is the same as if you were printing out the source using the <code>highlight_file()</code> function in a PHP script.
-v	By calling PHP with this option, you can ask it to print out its version number, ie: 4.0.6.
-C	Normally PHP changes the working directory to the running scripts direcrory. This makes it possible for example, to open files in the same directory, with only specifying the name of the file. If you would like to disable this directory change, use this option.
-c	Using this option, you can specify an alternative <code>php.ini</code> path, so PHP will search your configurations file in this path instead of the default one.

-d	With this option, you can set individual <code>php.ini</code> settings in the time of running a script.
-m	Using this option, PHP prints out the built in (and loaded) PHP and Zend modules, the PHP and Zend version numbers, and a short Zend copyright notice.
-i	This command line option calls <a href="#">phpinfo()</a> , and prints out the results. If PHP is not working well, it is advisable to make a <code>php -i</code> and see if any error messages are printed out before or in place of the information tables.
-h	With this option, you can get information about the actual list of command line options and some one line descriptions about what they do.

The PHP executable can be used to run PHP scripts absolutely independent from the web server. If you are on a Unix system, you should add a special first line to your PHP script, and make it executable, so the system will know, what program should run the script. On a Windows platform you can associate `php.exe -q` with the double click option of the `.php` files, or you can make a batch file to run the script through PHP. The first line added to the script to work on Unix won't hurt on Windows, so you can write cross platform programs this way. A simple example of writing a command line PHP program can be found below.

### Ejemplo H-1. Script intended to be run from command line (script.php)

```
#!/usr/bin/php -q
<?php

if ($argc != 2 || in_array($argv[1], array('--help', '-help', '-h', '-?'))) {
?>

This is a command line PHP script with one option.

Usage:
<?php echo $argv[0]; ?> <option>

<option> can be some word you would like
to print out. With the --help, -help, -h,
or -? options, you can get this help.

<?php
} else {
    echo $argv[1];
}
?>
```

In the script above, we used the special first line to indicate, that this file should be run by PHP and should not print out HTTP headers. There are two variables you can use while writing command line applications with PHP: `$argc` and `$argv`. The first is the number of arguments plus one (the name of the script running). The second is an array containing the arguments, starting with the script name as number zero (`$argv[0]`).

In the program above we checked if there are less or more than one arguments. Also if the argument was `--help`, `-help`, `-h` or `-?`, we printed out the help message, printing the script name dynamically. If we received some other

argument we echoed that out.

If you would like to run the above script on Unix, you need to make it executable, and simply call it as `script.php echothis` or `script.php -h`. On Windows, you can make a batch file for this task:

### **Ejemplo H-2. Batch file to run a command line PHP script (script.bat)**

```
@c:\php\php.exe -q script.php %1 %2 %3 %4
```

Assuming, you named the above program as `script.php`, and you have your `php.exe` in `c:\php\php.exe` this batch file will run it for you with your added options: `script.bat echothis` or `script.bat -h`.

---

[Anterior](#)

Aliases list

[Inicio](#)

[Subir](#)