



Torry Harris Business Solutions

PHP Scripting Language

Author - Karthik T.S.

www.thbs.com

Note: The contents of this Document are considered Torry Harris Confidential. No part of this Document may be reproduced in any form by any means without the prior written authorization of Torry Harris Business Solutions.

This article aims at presenting an overview of PHP technology and its use as a server-side scripting language for developing dynamic web-based sites and applications. The article discusses about various features of PHP, overview of standard functions and extension libraries, serving as a starting point for developers who would like to know about PHP and intend to learn it.

What is PHP?

PHP is a recursive acronym for "PHP: Hypertext Preprocessor" (formerly PHP was called Personal Home Pages). It is a widely used Open Source scripting language that is especially suited for server-side applications and dynamic web content.

To give a brief history, initial version of PHP was developed by Rasmus Lerdorf in 1995. PHP has undergone lot of evolution since then. PHP interpreter was re-written based on a scripting engine called 'Zend Engine', which was developed by a company called Zend Technologies. Since 1998, Zend Technologies maintains PHP, and manages all new feature evolutions. The latest PHP version is 5, based on Zend Engine II.

PHP supports interaction with a large number of databases such as MySQL, Oracle, IBM DB2, Microsoft SQL Server, PostgreSQL, Firebird and SQLite. PHP runs on most major operating systems, including Unix, Linux, Netware, Windows, and Mac OS X, and can integrate with many major web servers. PHP is typically installed in conjunction with other open source software to build Web applications, a platform referred to by the term "LAMP" (which stands for Linux, Apache, MySQL, and PHP).

PHP scripts can be used from a command line interface without any web server or a browser. This type of usage is ideal for scripts regularly executed using cron or simple text processing tasks.

Why PHP?

- PHP's syntax is similar to that of C, making it easy to learn and use for anyone with basic programming skills.
- Easy-to-use scripting language means PHP code is significantly easier to maintain and update compared to Perl or ASP code.
- PHP has basic object-oriented features such as classes, attributes, operations, etc; however, the latest version of PHP PHP5 significantly improved the OO model to support private member variables and methods, static variables, inheritance, abstract classes etc.
- PHP is open source, very well supported by a large developer community. The community gives excellent technical support to users, and bugs are found and fixed quickly.
- PHP is based on Zend engine, which offers PHP performance, reliability and scalability.
- Very good support for XML parsing.
- Widely used as a scripting interface by provisioning products.

Basic Syntax

PHP scripts are always enclosed between opening and closing PHP tags. The three different forms are shown below. In the examples shown below, 'echo' and 'printf' functions can be used interchangeably

PHP's language syntax is similar to C's and Perl's. The variables need not be declared before using them. PHP denotes all of its variables by using the \$ operator followed by any combination of characters. For example, \$greeting = "Hello World", \$name="Torry Harris", \$age=20. PHP is what is known as a "loosely typed" language. What that means is any given variable can be an integer, floating-point number, string, object, or an array.

```
<html>
<body>

<?php
    echo "Hello World";
php?>

<?
    printf("Hello World");
?>

<script language="php">
    echo "Hello World";
</script>

</body>
</Html>
```

The control structures in PHP are very similar to those used by the C language. PHP's control structures have two syntaxes that can be used interchangeably. The first form uses C-style curly braces to enclose statement blocks,

while the second style uses a more verbose syntax that includes explicit ending statements. The first style is preferable when the control structure is completely within a PHP code block. The second style is useful when the construct spans a large section of intermixed code and HTML. Common control structures with dual syntax are if/endif, for/endif, while/endwhile, switch/endswitch.

PHP provides three primitive data types: integers, floating point numbers, and strings. A string is a sequence of characters. A string can be delimited by single quotes or double quotes: 'Hello, World' "Hello, World!"

An array is a compound data type that can contain multiple data values. PHP treats all arrays as hash-tables and the hash, or index, can be a string or a number. These kinds of arrays are called associative arrays:

```
$var[0]="Hello World"; or $var["January"]=1;
$var = array("January" => 1, "February" => 2, "March" => 3);
$strArray = array("en" => "English", "fr" => "French", "sp" => "Spanish");
```

Creating Dynamic Web Pages with PHP

PHP is well suited for developing dynamic web pages, handling HTML forms, using cookies, sessions, etc. PHP supports pre-defined associative arrays such as \$_GET, \$_POST to access form data. There are other pre-defined arrays such as \$_COOKIE, \$_SESSION to represent the list of cookies and HTTP session variables. The following login form example demonstrates the simplicity in accessing form fields.

```
<form action="login.php" method="post">
  <p>User Name: <input type="text"
name="username" /></p>
  <p>Password: <input type="password"
name="password" /></p>
  <p><input type="submit" /></p>
</form>
```

```
<?php
// login.php Sample PHP source for a typical login
// handler using $_POST and $_SESSION built-in arrays

$loginID = $_POST["username"];
$password = $_POST["password"];

// In a real app, lookup the login and password from a database
if( $loginID == "hello" && $password == "world" )
{
    // Login successful, so start a HTTP Session
    session_start();

    // Set two session variables user name and login time
    $_SESSION["loginTime"] = date("d-M-Y");
    $_SESSION["principal"] = $loginID;

    echo "Login Successful";

    // Forward the control to App's home page
    include "homepage.php";
}
else
{
    echo "Incorrect credentials, Login failed";

    // Go back to login or an error page
    include "loginform.php";
}

Php?>
```

In addition to the above example, PHP allows access to standard CGI variables such as `REMOTE_ADDR` (client's IP address), `HTTP_USER_AGENT` (client's browser info), etc. These variables can be accessed using `$_SERVER` global array. PHP also supports `$_REQUEST` associative array consisting of the contents of `$_GET`, `$_POST`, and `$_COOKIE`.

PHP built-in functions and extensions

PHP contains several hundreds of built-in utility functions in the following categories: String & array handling, regular expressions, mathematical functions, file access, calendar functions, DOM, etc.

There are several other PHP API functions provided by 3rd party vendors and developers. These functions are called as PHP extensions. These functions “extend” PHP's current function set. Extensions are built as dynamic libraries (.so or .dll) depending on the platform. The most useful extension modules ship with the normal PHP distribution such as Bzip, MySQL, etc. Developers can implement their own extensions using C on both Linux/Unix and Windows platforms.

To manage the packaging, distribution and aggregation of useful extensions developed by 3rd party developers, a public repository called PEAR (PHP Extension and Application Repository) exists. PEAR is comparable to Perl's CPAN network.

Case Studies

For one of our Telecom Customers, we had to develop an application using a provisioning product. The product supports PHP as the standard scripting interface. We used PHP to develop web-based administrative GUI. The product provides PHP extensions to interact with its objects. The following simple example demonstrates the use of extension to connect to the product's environment, create an object and execute an operation on it.

```
<?php
// Connect to the product's environment using
// os_connect extension function (using the specified
// (host and port)

$connection = os_connect("myhost.com", 1234);

// Create an object reference in the product's runtime
$objRef = os_create($connection, "mypkg::MyClass");

// Invoke an operation on the object remotely
os_invoke($connection, $objRef, "myFunction");

Php?>
```

The above shown approach helped to exploit PHP's features to quickly develop administrative GUIs. It was also used to perform automated unit testing of the components from a command line environment using stand-alone PHP interpreter.

Further Reading and Useful Links

PHP is widely used by web developers across the world. Hence, lot of useful utilities, frameworks, IDEs, articles exists on the web. The following links i've you pointers for further reading on PHP.

- An IBM DeveloperWorks Article on Recommended PHP reading list:
<http://www-128.ibm.com/developerworks/opensource/library/os-php-read/>
- IDEs for PHP

- PHPEdit: <http://www.waterproof.fr/products/PHPEdit/>
- Zend Studio: http://www.zend.com/products/zend_studio
- PHP Eclipse Plugin: <http://www.phpeclipse.de/>
- PHP Designer 2005: <http://www.mpssoftware.dk/phpdesigner.php>
- Model View Controller (MVC) Frameworks for PHP:
 - Mojavi: <http://www.phpwact.org/php/mojavi>
 - CakePHP: <http://www.cakephp.org/>
 - PHPOnTrax: <http://phpontrax.com/>
 - Prado: <http://www.xisc.com/>
 - Studs (PHP port of Java Struts): <http://www.mojavelinux.com/projects/studs/>
 - Phrame: (Another PHP port of Java Struts): <http://www.phpwact.org/php/phrame>
 - WACT: <http://www.phpwact.org/php/znf>
 - Ambivalence: (A Java Maverick port) : <http://amb.sourceforge.net/>
- AJAX for PHP: <http://www.ajaxprojects.com/ajax/viewcategory.php?categoryid=3>
- Integrating PHP and Java: http://www.onlamp.com/pub/a/php/2001/06/14/php_jav.html
- PHPNuke, a PHP/MySQL based content management framework: <http://phpnuke.org/index.php>
- PEAR Home page: <http://pear.php.net/>
- PHP-GTK (Develop GUIs using PHPs): <http://gtk.php.net/>
- PHP Certification (Zend Certified Engineer): <http://www.zend.com/education/certification>

Glossary

PHP	Recursive acronym PHP Hypertext Preprocessor. Formerly, Personal Home Pages
LAMP	Linux, Apache, MySQL and PHP
Zend	Zeev Suraski and Andi Gutmans started Zend Technologies. The name Zend is a fusion of their forenames, Zeev and Andi.
PEAR	PHP Extension and Application Repository
CPAN	Comprehensive Perl Archive Network

Article sources:

- PHP Official Website: <http://www.php.net>
- Zend Technologies Official Website: <http://www.zend.com>

Torry Harris Business Solutions (THBS) is a US based IT service provider with development facilities in India and China. The services offered are in the areas of SOA, Testing, Offshore Product Development and IT Enterprise Services. The company, started in 1998, has for several years delivered a large variety of middleware services to enterprise clients around the world. Now, with a large pool of highly skilled technologists and rapidly growing, the company remains focused on the middleware and integration space, implementing large projects across the US, Europe, the Middle East and the Far East.

For more information, contact us at torryharris@thbs.com.
Web: www.thbs.com