



PHP Tech Talk

Presented By: Jignasha Vithalani

Overview:

- Server side scripting language
- Dynamic web page generation
- Open source (PHPLicense incompatible with GPL)
- Free
- Cross platform
- recursive acronym of Hypertext preprocessor which was previously Personal Home Page whose syntax borrowed mostly from Perl

PHP is mainly focused on server-side scripting, so you can do anything any other CGI program can do, such as collect form data, generate dynamic page content, or send and receive cookies. But PHP can do much more

As HTML can be mixed with PHP scripts which will be executed on server side , It will be less overhead to output HTML like in C or Perl

There are three main areas where PHP scripts are used

1) Server-side scripting

You need three things to make this work. The PHP parser (CGI or server module), a web server and a web browser. You need to run the web server, with a connected PHP installation. You can access the PHP program output with a web browser, viewing the PHP page through the server

2) Command line scripting

You can make a PHP script to run it without any server or browser. You only need the PHP parser to use it this way. This type of usage is ideal for scripts regularly executed using cron (on *nix or Linux) or Task Scheduler (on Windows). These scripts can also be used for simple text processing tasks

3) Writing desktop applications

PHP is probably not the very best language to create a desktop application with a graphical user interface, but if you know PHP very well, and would like to use some advanced PHP features in your client-side applications you can also use PHP-GTK to write such programs. You also have the ability to write cross-platform applications this way

- 1) PHP can be used on all major operating systems, including Linux, many Unix variants (including HP-UX, Solaris and OpenBSD), Microsoft Windows, Mac OS X, RISC OS, and probably others
- 2) PHP has also support for most of the web servers today. This includes Apache, IIS, lighttpd and many others , And this includes any web server that can utilize the FastCGI PHP binary, like lighttpd and nginx.
- 3) You have the choice of using procedural programming or object oriented programming (OOP), or a mixture of them both
- 4) With PHP you are not limited to output HTML. PHP's abilities includes outputting images, PDF files and even Flash movies (using libswf and Ming) generated on the fly. You can also output easily any text, such as XHTML and any other XML file.
- 5) Most significant features in PHP is its support for a wide range of database. Writing a database-enabled web page is incredibly simple using one of the database specific extensions (e.g., for mysql), or using an abstraction layer like PDO, or connect to any database supporting the Open Database Connection standard via the ODBC extension

- 1) PHP also has support for talking to other services using protocols such as LDAP, IMAP, SNMP, NNTP, POP3, HTTP, COM (on Windows) and countless others. You can also open raw network sockets and interact using any other protocol. PHP has support for the WDDX complex data exchange between virtually all Web programming languages. Talking about interconnection, PHP has support for instantiation of Java objects and using them transparently as PHP objects.
- 2) PHP has useful text processing features, which includes the Perl compatible regular expressions (PCRE), and many extensions and tools to parse and access XML documents. PHP standardizes all of the XML extensions on the solid base of libxml2, and extends the feature set adding SimpleXML, XMLReader and XMLWriter support.
- 3) Availability of various extensions and modules for different features and reusable codes
- 4) As source code is available It can be modified according to program need and PHP can then be recompiled for generating PHP binary
- 5) New Extension can easily be added which can either be dynamically linked or linked with Zend engine

- ❖ 6 out of 10 most popular websites of world has used PHP

Website	Popularity(unique visitors)
Google.com	1,000,000,000
Facebook.com	880,000,000
Youtube.com	800,000,000
Yahoo	590,000,000
Wikipedia.org	410,000,000
Wordpress.com	130,000,000

- ❖ **PHP is used by 77.8%** of all the websites whose server-side programming language known
- ❖ 20 million web servers over world has installed PHP support
- ❖ Free of cost LAMP stack instead of Microsoft stack costing lots of dollars for WISC

- ❖ Rapid web application development for small to medium sized websites due to lots of open source frameworks available

CMS

Joomla , Drupal , Moodle etc

Blog

Wordpress

Forum

PHPBB , vBulletin etc

CRM

SugarCRM

Ecommerce

Magento , Oscommerce , Zen cart etc

- ❖ *Around 20 Advanced Object Oriented Frameworks available which uses complex design patterns and structured design which can be used for large size robust software development*

Two ways to install PHP

1. PHP as apache module

When PHP runs as an Apache module, it is compiled into the Apache code itself. This means, when an Apache process starts, PHP starts along with it. They are intrinsically linked, and PHP depends on Apache to operate. The benefit of this is that Apache tends to run very efficiently, and PHP is part of each Apache process. Furthermore, Apache configuration, particularly when using .htaccess files, can also be used to control PHP functions

2. PHP as a CGI script

PHP as a CGI script means that PHP operates as an independent binary with its own processes. It is separate from Apache and can, therefore, run as another user, rather than Apache's generic user. This increases security and adds a bit of stability, at the expense of speed

- We never start any PHP daemon or anything by our self , When we start Apache it starts PHP interpreter along itself
- PHP is linked to apache (Using SAPI) by mode_php.so or mode_php.dll
- PHP as a whole consists of 3 modules
(Core PHP , Zend Engine and Extension layer)
- Core PHP is the module which handles the requests , file streams , error handling etc
- Zend engine is one which converts human readable code into machine understandable opcodes , Then It executes this general code into virtual machine
- Extensions are bunch of functions , classes , streams made available to PHP scripts , For example we need mysql extension to connect to mysql database using PHP
- While zend engine executes the generated code , the script might require access to few extensions , Then ZE passes control to extension module/layer which transfers back the control to ZE after completion of tasks.
- The PHP language is interpreted. The binary that lets you interpret PHP is compiled. Means PHP is compiled down to an intermediate bytecode that is then interpreted by the runtime engine

1. Readymade packages is available for windows or Mac Os named WAMP , MAMP
2. XAMPP is available for all platforms which is package of PHP , MySQL , apache , Tomcat , PhpMyadmin , Sqlite
3. LAMP requires separate configuration for all components
4. There are lots of PHP configurations which can be changed from PHP.ini file
5. PHP.ini settings can be changed in three ways
 - Directly in php.ini file
 - htaccess file
 - In PHP scripts using ini_set function
6. Which parameter valid to be changed depends upon changeable flag of parameter
7. PHP_INI_ALL type can be changed from anywhere
8. PHP_INI_PERDIR can be changed from .htaccess or php.ini
9. *php.ini* only can be changed only in php.ini file
10. If you change parameter which is not changeable from there then It's value won't affect. Example - short_open_tag can't affect it's change if done by ini_set function

11. httpd.conf is configuration file of Apache server
12. You can change any settings related to requests to servers , configurations of Virtual hosting , Directory specific permissions , URL rewrite , Error logs settings , Adding new apache modules , Authorization etc
13. httpd.conf settings can also be changed from per directory basis by .htaccess file
14. .htaccess file is apache's directory basis configuration file which can do following
 - Authorization and Authentication
 - Rewriting URLs
 - Blocking
 - Error responses
 - Cache control
15. Loosely coupled language
16. Free from declaring type of variables and don't require to define function definition before declaring it (Developer is free to pass any type of variable to function at run time)
17. Support for default function arguments

18. Provides complex data types in form of Arrays , Objects , Resources
19. Output buffering can easily be set and can clean buffer by sending contents to browser in chunk of processes or either at end of script
20. Modify HTTP response headers any parameter like sending redirect , content type , setting cookies etc
21. `$_GET` , `$_POST` , `$_SESSION` , `$_COOKIE` , `$_REQUEST` , `$_SERVER` super global variables
22. `session.save_handler` php.ini configuration can set where sessions will be stored on server , default is file but can be saved in Database as well
23. `session.save_path` used to set path on file system for storing session data
24. `session_id` is unique session seed and it's value is stored on cookie on client side , If cookie will be disabled on client side then it will pass it to URLs

- ❖ Latest PHP release is 5.4.4
- ❖ PHP5 for the most part is backwards compatible with PHP4, but there are a couple key changes that might break your PHP4 script in a PHP5
- ❖ Advanced OOP concepts has been added from version 5 , version 4 was having limited OOP support
- ❖ Exceptions introduced in version 5 in PHP
- ❖ E_STRICT error level added in version 5 (Deprecated code usage give warning)
- ❖ Useful extensions like SimpleXML , DOM , XSL , PDO , Hash , Curl and lot may more has been added in version 5 onwards
- ❖ Xdebug powerful extension for debugging PHP and PHPUnit for automated test cases unit testing extension added from version 5 onwards