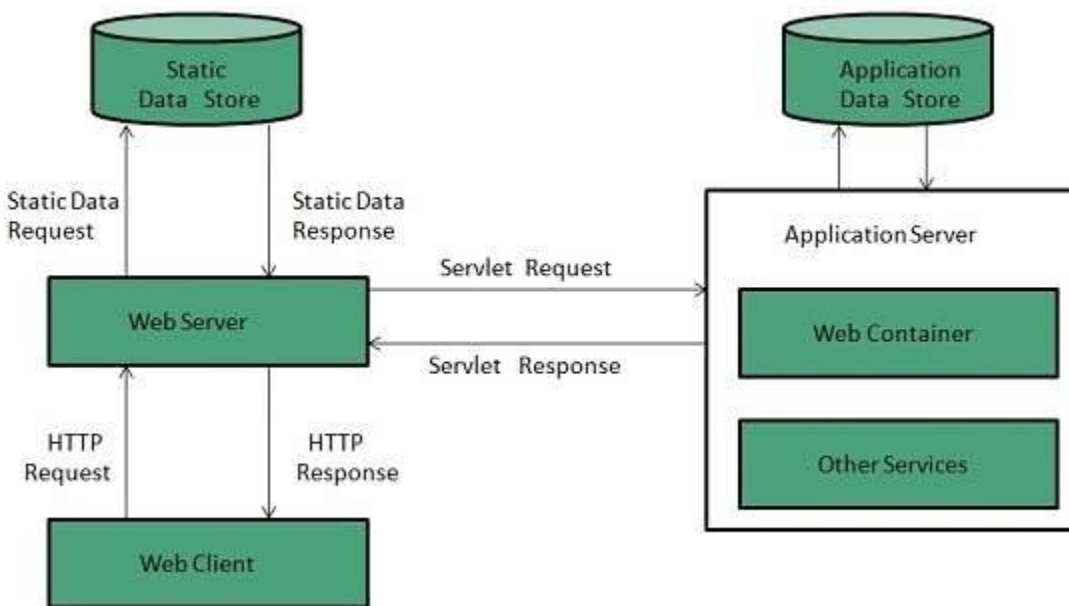## Web Server

Web server is a computer where the web content is stored. Basically web server is used to host the web sites but there exists other web servers also such as gaming, storage, FTP, email etc.

Web site is collection of web pages while web server is software that respond to the request for web resources. Primary components of a web server are it's **software** and **hardware** which works as per the requirement of an **operating system**

Web Server Working

Web server respond to the client request in either of the following two ways:

- Sending the file to the client associated with the requested URL.
- Generating response by invoking a script and communicating with database

Key Points

- When client sends request for a web page, the web server search for the requested page if requested page is found then it will send it to client with an HTTP response.
- If the requested web page is not found, web server will the send an **HTTP response: Error 404 Not found.**
- If client has requested for some other resources then the web server will contact to the application server and data store to construct the HTTP response.

Architecture

Web Server Architecture follows the following two approaches:

1. Concurrent Approach
2. Single-Process-Event-Driven Approach.

Concurrent Approach

Concurrent approach allows the web server to handle multiple client requests at the same time. It can be achieved by following methods:

- Multi-process
- Multi-threaded
- Hybrid method.

Multi-processing

In this a single process (parent process) initiates several single-threaded child processes and distribute incoming requests to these child processes. Each of the child processes are responsible for handling single request.

It is the responsibility of parent process to monitor the load and decide if processes should be killed or forked.

Multi-threaded

Unlike Multi-process, it creates multiple single-threaded process.

Hybrid

It is combination of above two approaches. In this approach multiple process are created and each process initiates multiple threads. Each of the threads handles one connection. Using multiple threads in single process results in less load on system resources.

Examples

Following details describes the most leading web servers available today:

1. Apache HTTP Server

This is the most popular web server in the world developed by the Apache Software Foundation. Apache web server is an open source software and can be installed on almost all operating systems including Linux, UNIX, Windows, FreeBSD, Mac OS X and more. About 60% of the web server machines run the Apache Web Server.

2. **Internet Information Services (IIS)**

The Internet Information Server (IIS) is a high performance Web Server from Microsoft. This web server runs on Windows NT/2000 and 2003 platforms (and may be on upcoming new Windows version also). IIS comes bundled with Windows NT/2000 and 2003; Because IIS is tightly integrated with the operating system so it is relatively easy to administer it.

3. **Lighttpd**

The lighttpd, pronounced lighty is also a free web server that is distributed with the FreeBSD operating system. This open source web server is fast, secure and consumes much less CPU power. Lighttpd can also run on Windows, Mac OS X, Linux and Solaris operating systems.

4. **Sun Java System Web Server**

This web server from Sun Microsystems is suited for medium and large web sites. Though the server is free it is not open source. It however, runs on Windows, Linux and UNIX platforms. The Sun Java System web server supports various languages, scripts and technologies required for Web 2.0 such as JSP, Java Servlets, PHP, Perl, Python, and Ruby on Rails, ASP and Coldfusion etc.

5. **Jigsaw Server**

Jigsaw (W3C's Server) comes from the World Wide Web Consortium. It is open source and free and can run on various platforms like Linux, UNIX, Windows, and Mac OS X Free BSD etc. Jigsaw has been written in Java and can run CGI scripts and PHP programs.

XAMPP Server

XAMPP is a free software. It simulates a real server and is used to test programs written in PHP or Perl languages. XAMPP is a set of connected software programs such as Apache, MySQL, FileZilla, Mercury, and Tomcat. The XAMPP set includes phpMyAdmin program, which can be used to manage permissions and to administer MySQL databases through the browser, or used as a substitution of MySQL, probably with restricted possibilities.

How to Install and Configure XAMPP Server
- Search your browser for the XAMPP downloads sites. Choose one and click Download and Run.
- Select the programs you want to install, in the menu that appears. Do not forget to select the Apache application. Then click Next.
- Look for the next window; choose the appropriate folder and click Next.
- Click Next.
- Click Next again.
- The XAMPP set software should start installing.
- Click Finish when the installation is complete.

- Understand that, when you start XAMPP, you may see an error message. Using the Config button, change the ports for this application and try to start Apache. You may see the message that some applications prevent Apache from starting. To fix this, go to the Computer window and click Manage.
- Click besides Service and Application in the pop-up window. In a drop-down menu, click SQL-Server Configuration and then double click SQL Server Service.
- Stop all running applications using the indicated buttons.
- Close Computer Management and start Apache again. If all is done properly, you will see the running program. You can choose appropriate ports for the fixing process.

WAMP Server

WAMP Server refers to a software stack for the Microsoft Windows operating system, created by **Romain Bourdon** and consisting of the Apache web server, OpenSSL for SSL support, MySQL database and PHP programming language.

How to Install and Configure WAMP Server

- Before you do anything you need to download WAMP from http://www.wampserver.com/en/download.php. You will have the option to choose wither a 32 or 64-bit environment depending upon your computer's configuration. In addition to Apache, PHP and MySQL this download includes phpMyAdmin and SQLBuddy to better help you manage your databases through web based GUI rather than having to use a terminal interface.
- Once you have downloaded and uncompressed your WAMP executable you will need to go ahead and launch it to start the installation process.
- When you click next you will be asked to accept the license agreement. Since it is a GPL license you are free to do just about anything with it so you can go ahead and accept.
- The next step requires you to select the folder where you would like to install your WAMP server. The default will be **c:\wamp** however you can change this to install the server into any directory or partition you choose.
- After you choose your directory you will have the option to setup icons. Once you decide on this click *Next* and then confirm the installation settings again by clicking *Install*.
- Once the installation runs its course you will be asked to choose your default browser. Internet Explorer is the default choice but you can navigate your way to any other browser of your choosing. *if your Windows firewall pops up at this point make sure to grant Apache access.*
- The next decision you will have to make is to set the PHP mail parameters. Many people leave this set to the defaults when setting up a testing server on their local computer. If you wish to configure it to connect to your SMTP server you may do so here but unless you plan on testing email capabilities the default entries can be left and all you need to do is click **Next**.
- Setting the SMTP server (optional)
- Congratulations, WAMP is now installed on your local computer.

- Now that everything has been installed let's test everything out.Using one of the icons you created, or *Start –> All Programs –> WampServer –> start WampServer*, you can launch the management console. Once opened, it will appear in the lower right hand corner of your screen.

<u>LAMP Server</u>

LAMP is an archetypal model of web service stacks, named as an acronym of the names of its original four open-source components: the GNU/Linux operating system, the Apache HTTP Server, the MySQL relational database management system (RDBMS), and the PHP programming language. The LAMP components are largely interchangeable and not limited to the original selection. As a solution stack, LAMP is suitable for building dynamic web sites and web applications.

<u>How to Install and Configure LAMP Server</u>

Step 1: Install Apache and Allow in Firewall

We can install Apache easily using Ubuntu's package manager, apt. A package manager allows us to install most software pain-free from a repository maintained by Ubuntu.

For our purposes, we can get started by typing these commands:

- sudo apt-get update
- sudo apt-get install apache2

Step 2: Install MySQL

Now that we have our web server up and running, it is time to install MySQL. MySQL is a database management system. Basically, it will organize and provide access to databases where our site can store information.

Again, we can use apt to acquire and install our software. This time, we'll also install some other "helper" packages that will assist us in getting our components to communicate with each other:

- sudo apt-get install mysql-server

Step 3: Install PHP

PHP is the component of our setup that will process code to display dynamic content. It can run scripts, connect to our MySQL databases to get information, and hand the processed content over to our web server to display.

We can once again leverage the apt system to install our components. We're going to include some helper packages as well, so that PHP code can run under the Apache server and talk to our MySQL database:

- sudo apt-get install php libapache2-mod-php php-mcrypt php-mysql

We may restart the apache server using this command:

- sudo systemctl restart apache2

Step 4: Test PHP Processing on your Web Server

In order to test that our system is configured properly for PHP, we can create a very basic PHP script.We will call this script info.php. In order for Apache to find the file and serve it correctly, it must be saved to a very specific directory, which is called the "web root".In Ubuntu 16.04, this directory is located at /var/www/html/. We can create the file at that location by typing:

- sudo nano /var/www/html/info.php

This will open a blank file. We want to put the following text, which is valid PHP code, inside the file:

info.php
<?php
phpinfo();
?>

When you are finished, save and close the file.

Now we can test whether our web server can correctly display content generated by a PHP script. To try this out, we just have to visit this page in our web browser. You'll need your server's public IP address again.

The address you want to visit will be:

http://your_server_IP_address/info.php

Using these steps we may install and configure LAMP Server effectively.

Client-side scripting and it's role in web development

The client-side environment used to run scripts is usually a browser. The processing takes place on the end users computer. The source code is transferred from the web server to the users computer over the internet and run directly in the browser. The scripting language needs to be **enabled** on the client computer. Sometimes if a user is conscious of **security risks** they may switch the scripting facility off. When this is the case a message usually pops up to alert the user when script is attempting to run. Client-side scripting is changing interface behaviors within a specific web page in response to mouse or keyboard actions, or at specified timing events. In this case, the dynamic behavior occurs within the presentation. The client-side content is generated on the user's local computer system. Such web pages use presentation technology called rich

interfaced pages. Client-side scripting languages like **JavaScript** or **ActionScript**, used for **Dynamic HTML (DHTML)** and **Flash technologies** respectively, are frequently used to orchestrate media types (sound, animations, changing text, etc.) of the presentation. Client-side scripting also allows the use of remote scripting, a technique by which the DHTML page requests additional information from a server, using a hidden frame, XMLHttpRequests, or a Web service. The first widespread use of JavaScript was in 1997, when the language was standardized as ECMAScript and implemented in Netscape 3.

Example
The client-side content is generated on the client's computer. The web browser retrieves a page from the server, then processes the code embedded in the page (typically written in JavaScript) and displays the retrieved page's content to the user. The innerHTML property (or write command) can illustrate the client-side dynamic page generation: two distinct pages, A and B, can be regenerated (by an "event response dynamic") as document.InnerHTML = A and document.innerHTML = B; or "on load dynamic" by document.write (A) and document.write (B).

Server-side scripting and it's role in web development

A program running on a web server (server-side scripting) is used to generate the web content on various web pages, manage user sessions, and control workflow. Server responses may be determined by such conditions as data in a posted HTML form, parameters in the URL, the type of browser being used, the passage of time, or a database or server state. Such web pages are often created with the help of server-side languages such as **ASP, ColdFusion, Perl, PHP, Ruby, Python, WebDNA** and other languages, by a **Support server** that can run on the same hardware on the web server. These server-side languages often use the Common Gateway Interface (CGI) to produce *dynamic web pages*. Two notable exceptions are ASP.NET, and JSP, which reuse CGI concepts in their APIs but actually dispatch all web requests into a shared virtual machine. Dynamic web pages are often cached when there are few or no changes expected and the page is anticipated to receive considerable amount of web traffic that would create slow load times for the server if it had to generate the pages on the fly for each request. The **server-side environment** that runs a scripting language is a web server. A user's request is fulfilled by running a script directly on the web server to generate dynamic HTML pages. This HTML is then sent to the client browser. It is usually used to provide interactive web sites that interface to databases or other data stores on the server. This is different from client-side scripting where scripts are run by the viewing web browser, usually in JavaScript. The primary advantage to server-side scripting is the ability to highly customize the response based on the user's requirements, access rights, or queries into data stores.

Difference of PHP with other server side languages

Difference between PHP and ASP

- PHP is based on C++ language and its structure is similar to C, C++, Java, TCL and Perl while ASP is based on VBScript.

- Operating system strength of PHP is more than ASP. PHP can be easily use on Linux, Unix and Windows operating system while ASP is mostly used on Windows operating system as it is developed by Microsoft.
- Many auxiliary and assembling features like ftp, email sharing and encryption system is already included in PHP, which are totally neglected in ASP.
- Both ASP and PHP are the technologies of '90s. PHP regularly released its latest version within short intervals of time and remain up to date with latest development requirements. While ASP had been dysfunctional since 2000.
- PHP performance and code speed is faster and easier than ASP.
- Microsoft Visual Studio 2010 is used as a platform for ASP developers while Eclipse Helios 2010 is used as a platform for PHP developers.
- MS Sql server is used as a database server for ASP and MySQL database server is used for PHP.
- Although PHP is simple easier than ASP but functions and controls of ASP are more efficient and reliable than PHP.

Difference between PHP and JSP

- JSP (Java Server Pages) is a server-side programming language that allows the formation of platform-independent, dynamic method for constructing applications based on the web. Moreover, JSP uses Java as the base for programming. On the other hand, PHP (Hypertext Preprocessor) is a broadly used as an open-source general-purpose scripting language that is particularly suitable for web development and can be rooted in HTML. Particularly, PHP is a general-purpose programming language and it does not need any other programming language.
- JSP requires more code and is complex. JSP allows the division of stationary contents from active contents. The active or dynamic contents are produced using programming logic and added into the template of static. This really abridges the formation and protection of web contents. Whereas, PHP is a programming language that can perform all kinds of things like making convention web content to serve the browser, assess form data sent from a browser, receive & send cookies and talk to a database. In short, PHP is simple and needs fewer lines of codes.
- You can effectively debug PHP applications. You are required to have the XDebug debugger installed in your PC, the Apache local web server, the PHP engine and configured for PHP development. To be more elaborate, the print statements are used in PHP. On the other hand, it is hard to debug a JSP. JSP be likely to engage a big quantity of server contact, making errors probably but difficult to replicate. In easy words, Products of JSP can be debugged by editors.
- JSP make numerous means accessible to Web developers to secure applications. Resources are protected by recognizing them in the application deployment descriptor and assigning a role to them. There are a number of problems in PHP and sadly not all of them have definite solutions. These are the troubles which, with elegant code, can be eradicated completely. Some of them are SQL Injection, Directory Traversal, Authentication issues and much
- A JSP (Java Server Pages) can be used with an HTML form tag to let users upload files to the server. An uploaded file can be a binary, a text file or an image file or any other

document. Whereas, with PHP, it is easy to upload files to the server! Initially, make sure that PHP is configured to allow file uploads.

- The performance of Garbage Collection of PHP can be difficult to understand and you might it's working & if you can develop its use of your app. If it falls out of range and is not used in any other place of the presently implemented code any longer then the garbage is collected automatically. In JSP, garbage means unreferenced objects. In Java, it is performed automatically. So, java provides better memory management.

Difference between PHP and PYTHON

- Among PHP and Python, the PHP is most widespread and pervasively used.
- PHP and Python, both languages are readable but Python is more maintainable than PHP and comprised of very few keywords.
- PHP allows loosly typed programming practices which result in many security-related bugs, although it can be used securely. On the contrary, Python provides more security features than PHP.
- Python supports functional programming whereas PHP doesn't offer functional paradigms.
- PHP doesn't support exception properly; conversely, in python, there is proper provision for exception handling.
- In python, a "yield" statement is used for the generator function. On the other hand, PHP has no provision for threads (concurrent programming).
- PHP has a syntax that is similar to C. It contains lot of curly braces, dollar signs and "->"-sign. Whitespace is ignored. Python doesn't have curly braces, instead the level of indention of blocks of code are important. It has a very clear, concise, and orthogonal syntax.
- PHP has web development features built directly into the core language, whereas Python's web development capabilities are provided by add-on modules.
- PHP has increment and decrement and assignment operators, whereas assignment is a statement only in Python
- PHP has interfaces. Python has multiple inheritance and Abstract Base Classes.

Difference between PHP and RUBY

- Ruby does not provide different syntax to access the methods, variables, and properties. PHP provides different syntax to access the same.
- In Ruby, statement terminators can be used as Newline terminator, it means a character in next line. In PHP, the statement terminator is a semicolon.
- In Ruby, to concatenate two strings '+ (plus)' is used. In Ruby, to concatenate '. (dot)' operator is used.
- Ruby does not support integer data types. PHP supports 32 and 64-bit integer data types.
- Ruby has strongly typed typing. PHP has weak typing.
- Ruby is expressive and readable. PHP is robust and simple
- Ruby uses 'END' word to finish the code blocks. In PHP, curly brackets {} are used to end the code blocks.

- Ruby use different syntax like to find a length of string it can be written as 'string.length'. Whereas in PHP, it can be written as 'strlen($string)'.
- Ruby applications are expensive and complex to host and very time-consuming process. PHP applications are far cheaper and take less time.
- Ruby has supported records and complex numbers but PHP does not support complex numbers and records.
- Ruby syntax is easier to understand and grasp. PHP syntax is complex as of its standard library.

Difference between PHP and PERL

- In Perl we have access to the symbol table, but in PHP the manipulation of the symbol table is restricted to creating references and the usage of extract function.
- Again with regards to symbol table itself, Perl has separate table entries for scalars, arrays etc but PHP has not.
- In regard with the logical operators Perl gets a return of the arguments while PHP gets a Boolean return.
- The usage of references differs in Perl and PHP. In Perl they are pointers but in PHP they are aliases used in Symbol table.
- Perl has a native regular expressions and regular expression literally support but PHP uses Perl's regular expression functions as an extension.
- Perl supports tail calls explicitly with the help of goto function.
- Variables in Perl are global by default but in PHP variables are local by default.