

# Apache, PHP and SQL *for web developers*

By Saul Cisneros

Part. 1 of 2

Part 1..... Theory

Part 2.....Implementation

## Preface

There are probably several hundred ways to explain these concepts. I will try not to include unnecessary information, but also not be too vague. The title may look somewhat familiar since many books are available that describe Apache, PHP and MySQL. However, I choose not to go on about the specifics of MySQL since it is not usually relevant when creating a site that needs relational database features. The only reason I'm including Apache is because of the way it handles security and virtual hosts that are relevant when designing a site. PHP of course is what everything revolves around. It is the "core" tool that will allow you to produce dynamic content. This means that you could decide to run a IIS server, and have PHP scripts calling data from an MSSQL database server. That will not be explained here though. Please enjoy.

## Why not use plain HTML?

The sole purpose of HTML is to display content while providing limited formatting capabilities and allowing documents across a network to be "linked" together. This is great for a site that will display a permanent set of information that will be updated infrequently. However, what if you needed to accept user input? You could use an HTML feature to collect the data and email it to a pre-determined address. But what if you needed to be able to immediately have that information available? What if you wanted each visitor to have a unique profile so that each time they visit they would have a set of specific options available to them? HTML does not provide those features.

## Overview of a web browser

A web browser such as Internet Explorer, or Mozilla Firefox, interprets the HTML sent from a web server and assembles it into the content you see in the browser window. That is called a "client side" scripting process. See fig. 1. As you may have guessed that means the processing of the HTML data is done by software based on a user's workstation. Processing done on the server is called "server side" scripting; this is what happens to PHP scripts you have written on your webpage. A "script" is a special command embedded into the web document that will be processed on the web server. A web browser does have some ability to store information. This is comes in the form of a small set of universal environment variables. This means that the variable names are the same

from any web browser whether it is Mozilla or Explorer. These variables provided by the web browser are not expandable. Some hold predetermined types of values, such as the IP address of the workstation, and the workstations current screen resolution. Others can be assigned other variable values. Cookies are variables called arrays that are able to hold almost any type of information. A scripting language can use these variables to determine if it should display a pre-made site tailored to a specific screen resolution or log the IP of the web visitor. A web browser, interpreting only HTML has no way of knowing which page you have visited before the one you are looking at, does not have a robust system to work with variables, and does not have a way to permanently store data.

## Overview of a web server

A web server is just what it says it is, software that serves web pages. Web server software can be run in almost any type of device from a cell phone to a router. The Apache web server is a program that listens for incoming requests for connections, and when it receives a request it carries out options specified in its main configuration file, that instruct Apache if and how it will fulfill the requests. The Apache configuration file allows apache a lot of flexibility. It includes support for ACL's (Access Control List), that you can use to filter requests by IP, and/or hostname to the whole web server or specific directories. You can also specify to not allow certain files to be viewed, or you can specify to only allow password protected access to given directories. There are many more options you can configure but we will only be considering a few of them. Apache is open source, this means that the source code or the main program that makes up Apache can be modified or added to by anyone that sees the need to make a contribution, provided that they submit this addition to Apache for future releases. What if you want to host more than one website on the server? This is the function of the Apache Virtual Host feature. This feature allows you to run separate hosts from within the same Apache server, with each having its own configuration options. These hosts can reside on the same IP which is called name based Virtual Hosting, or reside on different IP's which is called IP based Virtual Hosting. Support for encryption such as SSL, requires a separate installation of an Apache server running on a separate port from the non-encrypted server.

## Connecting the browser to the web server

The routing that directs traffic from the web browser to the web server is directed by registrar domain records and DNS (Domain Name System) servers. Every domain name has an associated IP address. It is possible to have one IP address associated with several domains, as in the case of name based virtual hosting. The registrar is a company that has the authority to sell domain names. When you purchase a domain, a record is created for it called a "whois" record. The main contents of this record are the domain owners contact information, domain expiry information and the DNS routing information. The DNS servers are usually managed by your web host, and contain a "zone record" for each domain that specifies where to route traffic destined for that domain. So let's say for

example you type in an address of a domain you own such as mydomain.com. Once you try to bring up the address in the web browser, the request is sent to the domain registrar to find out which DNS servers contain specific information regarding your domain, the proper DNS servers are contacted and are given the specific IP that your domain is associated with, then you are directed to the correct web server. See fig. 2. If the server is using name based virtual hosting, it uses the domain name information given by the incoming request to determine which virtual host to display.

## Storing information

Now, with the following arrangement we can server any type of static document. But what if we want to be able to store information about a web visitor, to have an online store, or to accept information from an online visitor to be used at a later point? These things require some type of system to store information. This can generally be done in two ways. One, you could keep everything in variables. This would be great but once the users browsing session ends, the data will be lost. If you need a permanent way to store the information you need some type of database. Since a database is just a collection of data a database can be a text file on the server. Although your data will be stored permanently, this is not a robust system. Most hosting providers these days make “database servers” available. A database server is a program that manages a collection of information or as we would call it a “database”. Most modern database servers provide methods to organize and display data, control user access, and maintain the integrity of the database. If you were to use a flat file database you would need to develop ways to provide those features via the scripting language provided by your hosting provider. Also most database servers are ODBC (Open DataBase Connectivity) compliant. This means that you can access a compliant server using a standard set of SQL (Server Query Language) commands. See fig. 3. The SQL language was developed with people who were new to computers in mind. Therefore many of the commands are intuitive and straight forward. This was to make it easy to transition traditional bookkeepers to using this language to manage the database information. Because of this open standard, applications built using SQL developed for a certain database server, can be easily made to work with another database server.