

Comparing Functional Programming and Object Oriented Programming Languages on the Server Side Programming

Özgür Örnek
International Burch University
Sarajevo, Bosnia and Herzegovina
oornek@ibu.edu.ba

Abstract : The use of web applications has been rapidly expanded into all sectors of society such as government, business, education, and industry. Web applications have to be thorough tested to ensure their correctness and meet the software requirements. Today, there are many different methods available to develop dynamic web applications but all these methods have some advantages and some disadvantages. It is very difficult to decide which solution should be chosen for a particular problem in a specific environment. Some dynamic script languages based on object oriented programming have been introduced such as JSP servlets and ASP.NET. Some of them based on functional programming languages such as ASP and PHP. In this paper, we adapt to specify advantages and disadvantages of developing web application with object oriented programming and functional programming languages. To achieve this aim, we picked up JSP and servlet as object oriented programming language and PHP as a functional programming language. We implemented different stock programs in JSP and PHP programming languages in order to compare their performances.

Keywords: JSP, Servlet, PHP, Server Side Programming

1. Introduction

In the early years of the Internet, the web application was static. The web content that sent to the browser was entirely same. It was not possible to implement web-based business applications that require database connection, user authentication, session management, and dynamic HTML generation.

To eliminate web-based business applications demand, server side programming languages were developed as a dynamic web application. Over the years, numerous alternatives and enhancements, such as FastCGI, mod_perl from Apache, NSAPI from Netscape, ISAPI, ASP, ASP.NET from Microsoft, PHP, Coldfusion, and Java servlets, JSP from Sun Microsystems, have been created.

Today, there are many different methods available to develop dynamic web applications but all these methods have some advantages and disadvantages. It is very difficult to decide which solution should be chosen for a particular problem in a specific environment. Some dynamic script languages based on object oriented programming have been introduced such as JSP servlets and ASP.NET. Some of them based on functional programming languages such as ASP and PHP.

However, as much attention has been paid to the development of Web applications, there exist very few reports on Web application testing. In this paper, we endeavor to specify advantages and disadvantages of developing web application with object oriented programming and functional programming languages. For this purpose, we picked up JSP as object oriented programming language and PHP as a functional programming language. We implemented same stock implementation in JSP and PHP programming languages in order to compare their performances.

Similar studies have been handled by particular researchers. Henderson-Sellers (Henderson-Seller, 2000) concluded that in object oriented programming languages are more successful in complex application development. Roberts (Roberts, Gary, 2005) analyzed server side programming languages such as PHP, ASP, ASP.NET, and ColdFusion with advantages and disadvantages. Based on his study, preferring suitable server side technology depends on operating system, economy, ease of using, and sustainability. Yang et al. (Yang et al. 2002) expanded traditional software testing architecture to support Web Application setting. A set of tool was developed to help analyze documents, develop test cases, execute tests, monitor failures, and support test measurement.

Ricca and Tonella (Ricca and Tonella, 2002) described an analysis model that captures web pages, forms, frames, and their relationship of Web applications.

Masaud et al. (Masaud et al. 2006) compared JSP and ASP.NET framework in MVC implementation. Their findings showed that implementing the ASP.Net framework was easier than JSP framework to achieve the

MVC major concept.

Suzumura et al. (Suzumura et al. 2008) conducted performance comparison of web service engines in PHP, Java, and C.

2. Web Application Architecture

A typical web application consists of a web server, an application server and a database server. Figure 2.1 illustrates web application architecture.

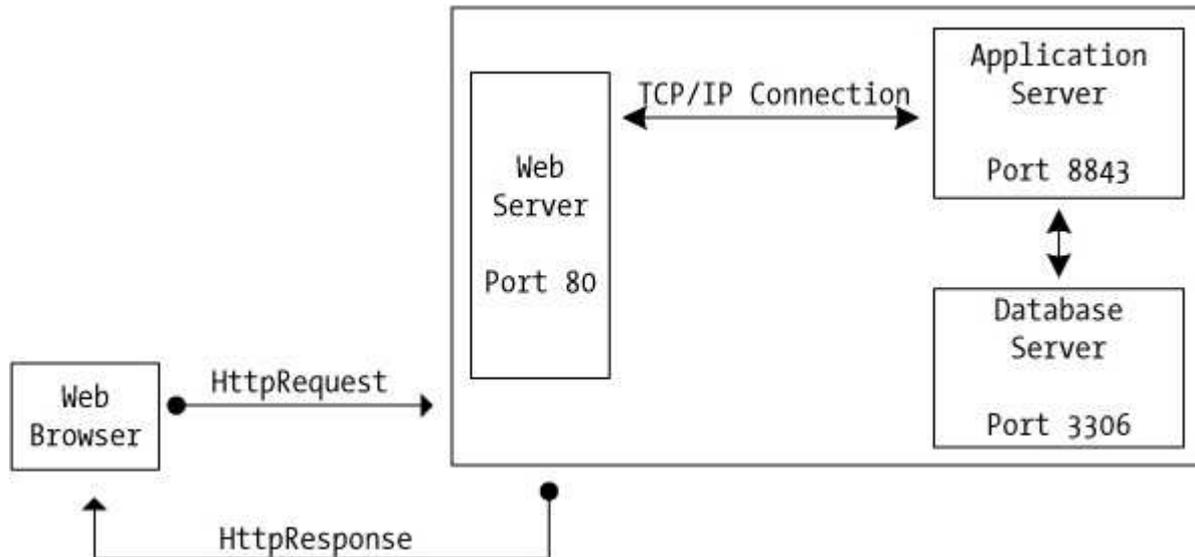


Figure 2.1: Web application architecture

The typical flow of handling a request starts with the browser issuing a request for a specific resource. The web server picks up this request off of port 80 and determines that the application server should handle the request. The application server receives the request and executes some code to handle the request. Within the code, the database may be called by making a connection to whatever port to which the database server is listening. Once a response has been assembled, it's sent back to the browser from where it originated. (Patzner, 2002)

3. Developig Java Web Applications

Server-side Java programming model is based on two recent Java server-side technologies: Servlets, Java Server Pages (JSPs). Servlets (and JSP) offers the following benefits that are not necessarily available in other technologies:

Performance: The performance of servlets is superior to CGI because there is no process creation for each client request. Instead, each request is handled by the servlet container process. After a servlet is finished processing a request, it stays resident in memory, waiting for another request.

Portability: Similar to other Java technologies, servlet applications are portable. You can move them to other operating systems without serious hassles.

Rapid development cycle: As a Java technology, servlets have access to the rich Java library, which helps speed up the development process.

Robustness: Servlets are managed by the Java Virtual Machine. As such, you don't need to worry about memory leak or garbage collection, which helps you write robust applications.

Widespread acceptance: Java is a widely accepted technology. This means that numerous vendors work on

Java-based technologies. One of the advantages of this widespread acceptance is that you can easily find and purchase components that suit your needs, which saves precious development time.

3.1. Java Servlets

Servlets are nothing more than Java classes that provide special server side service. This service defines a set of lifecycle methods that can be overridden to provide dynamic responses to HTTP requests.

The web container can load and manage multiple servlets within the same Java Virtual Machine (JVM). Once a servlet is loaded, it handles each incoming request by spawning a new lightweight thread to perform its task. This enables the application server to service a great number of requests without significant degradation of performance. It is illustrated at Figure 3.1. (Chunlin. ?)

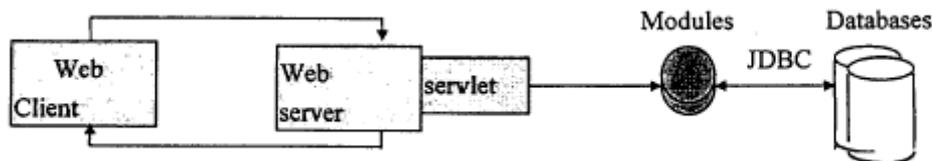


Figure 3.1: Servlet working architecture

3.2. Java Server Pages (JSP)

Java Server Pages (JSP) technology enables to mix regular, static HTML with dynamically generated content from servlets. With JSP, web designers and developers can quickly incorporate dynamic elements into web pages using embedded Java and a few simple markup tags. These tags provide the HTML designer with a way to access data and business logic stored inside Java objects without having to master the complexities of Java application development.

JSP files contain traditional HTML along with embedded code that allows the page designer to access data from Java code running on the server. When the page is requested by a user and processed by the HyperText Transport Protocol (HTTP) server, the HTML portion of the page is passed straight through. The code portions of the page, however, are executed at the time the request is received, and the dynamic content generated by this code is spliced into the page before it is sent to the user. This provides for a separation of the HTML presentation aspects of the page from the programming logic contained in the code. It is illustrated at Figure 3.2 (Visualbuilder.com Team 2001)

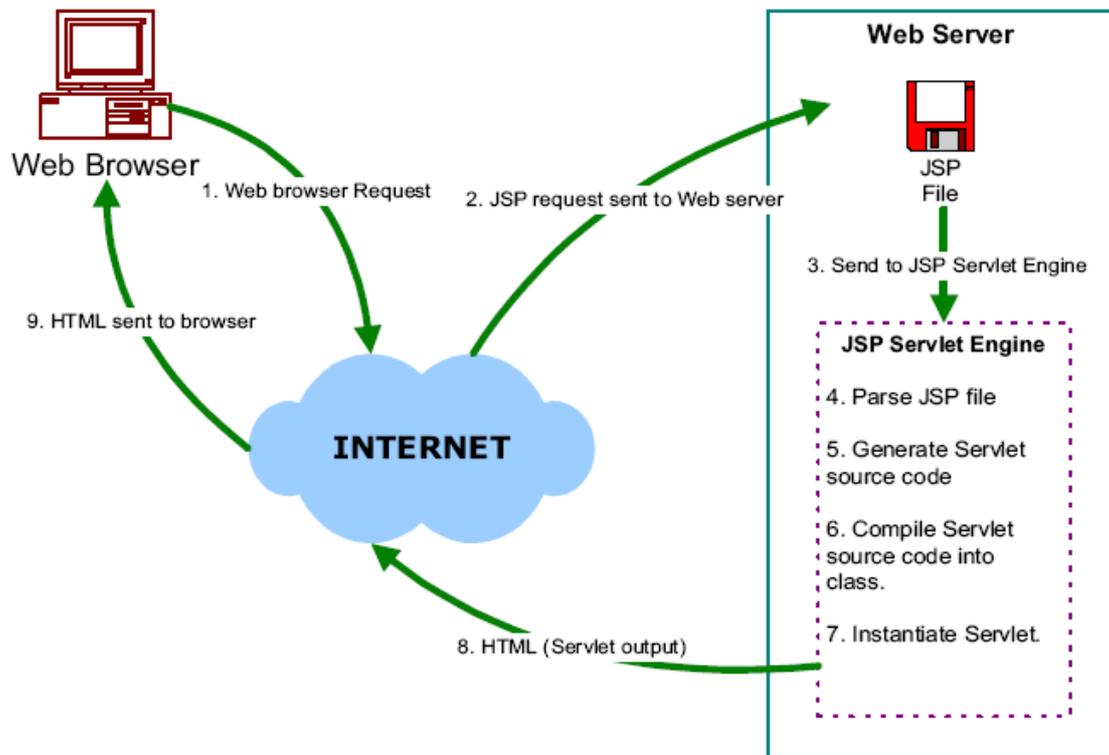


Figure 3.2: JSP Architecture

4. Developing PHP Web Applications

The PHP language is widely used Open Source general-purpose scripting language that is especially suited for Web development and can be embedded into HTML. A great number of open source software and company's web sites use PHP since it can enable high software productivity. PHP usage statistics is illustrated in Figure 4.1 (<http://www.php.net/usage.php>)

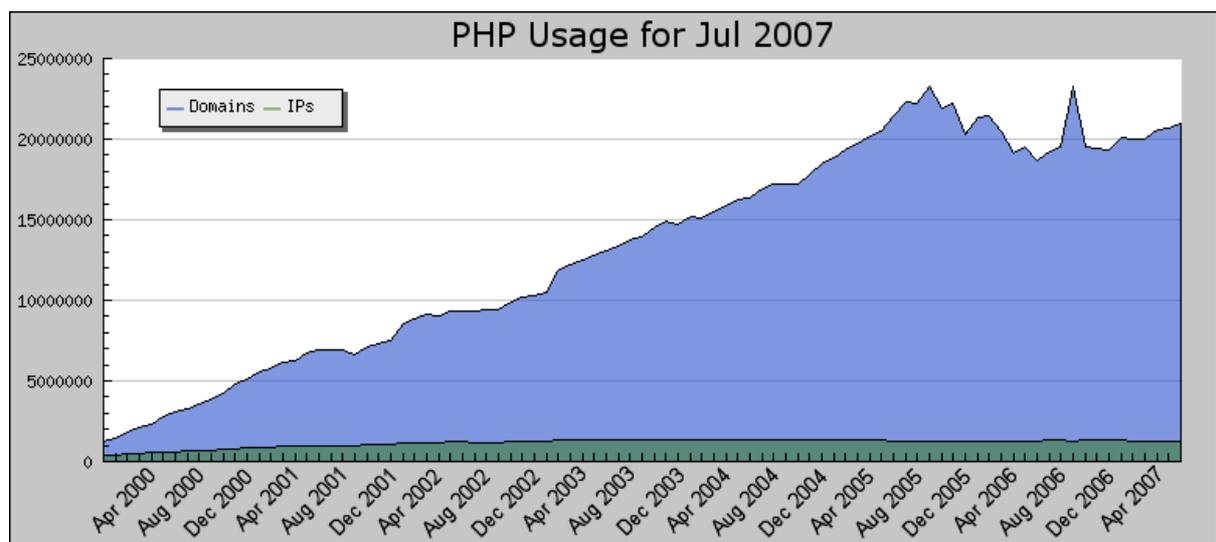


Figure 4.1: PHP usage for Jul 2007

PHP can be used on all major operating systems, including Linux, many Unix variants, Microsoft Windows, Mac OS X, RISC OS, and probably others. PHP has also support for most of the web servers today.

This includes Apache, Microsoft Internet Information Server, Personal Web Server, Netscape and iPlanet servers, and many others. For the majority of the servers PHP has a module, for the others supporting the CGI standard, PHP can work as a CGI processor.

PHP has many strengths, including the following:

Stability, and compatibility : It works well with other software and can be quite fast. PHP uses the machine resources very efficiently.

Currently, PHP runs stable on a range of operating systems including most flavors of UNIX, Windows and Macs and integrates well with most popular servers including IIS and apache. One of the strongest and most significant features in PHP is its support for a wide range of databases. Writing a database-enabled web page is incredibly simple.

Cost : PHP is free. PHP is a popular choice for companies hoping to avoid licensing costs. It's also the most popular choice for small business web hosting providers who almost always provide PHP and MySQL support. MySQL is frequently used by those small businesses, for example, to maintain a database containing a catalog of their merchandise. PHP allows those vendors to calculate prices based on complex selection criteria (like product color, size, and style). PHP (free) is frequently used in conjunction with Linux (a free operating system), MySQL (a free database), and Apache (a free web server) - all free and collectively referred to as "LAMP".

Availability of Support : PHP is Open Source. Because PHP's source code is freely available, a community of developers is always working to improve, add to, and find bugs in the language. Open Source means you never need to rely on the manufacturer to release the next version if something doesn't work or pay for expensive upgrades.

Simplicity with Built in Libraries : The syntax of PHP based on other programming languages, primarily C and Perl. It has many built-in functions for performing many useful web tasks. PHP can generate GIF images on the fly, connect to web services and other network services, parse XML, send email, and generate PDF documents, all with just a few lines of code. (Welling, L. & Thomson L. 2005)

5. Experimental Results

The goal of this benchmark is to demonstrate performance analyze of JSP and PHP technologies. Thus, we developed online two stock programs with two technologies separately. We used MySQL as database server, and Apache as web server.

In order to analyze web applications, virtual test tools are used. To measure Web page performance and to test stability, we used Webserver Stress Tool. With this test tool, we can test with specifying number of virtual users and each user's page request in limited duration.

Test tool was configured to use 100 and 500 concurrent connections in particular duration. For the tests, we used PHP system, with a single 1.4 MHz Pentium M CPU and 256 MB of RAM.

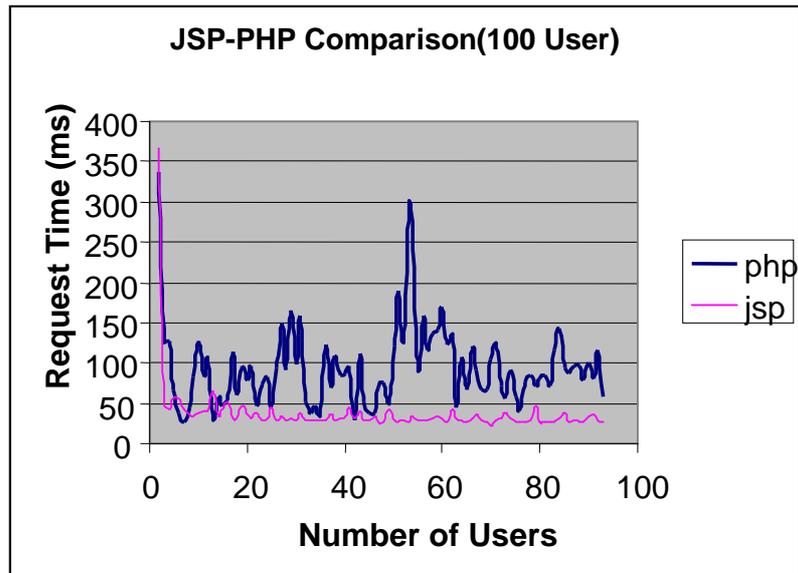


Figure 5.1: 100 users comparison in specified time

First, we tested 100 users with increasing load for a specified time. From Figure 5.1, reasonably high performing compared to PHP. In the beginning, since the page loading such as compiling and interpreting takes much time, request time values are higher. After 50 user, PHP graph change big variation than JSP. As number of users increase, PHP request time increases, whereas JSP has less request time than PHP. We can say that JSP is more stable and faster than PHP.

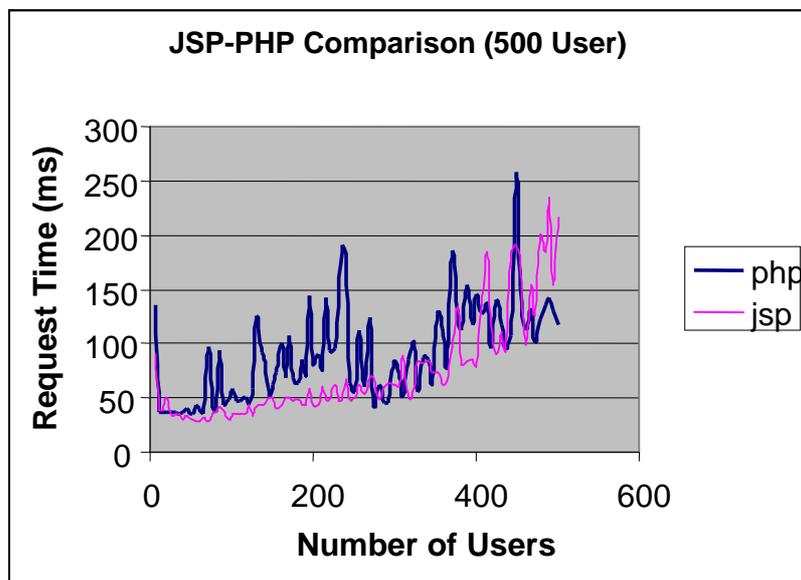


Figure 5.2: 500 users comparison in specified time

Second, we attempted to measure 500 users loading for a specified time. From the results, it is seen that JSP request time smaller than PHP overall users. But, the request time results have little differences between JSP and PHP. JSP is more stable and faster than PHP.

6. Conclusion

According to results of test:

- a. In terms of performance, JSP technology is faster and more stable than PHP.
- b. In terms of simplicity, since PHP syntax is based on C and has built in functions to generate complex issues, it can be said that PHP is more easier than JSP. However, JSP uses Java classes and X M L tags to generate codes.
- c. From the results, we conclude that the security approach gives better performance in JSP.
- d. Considering development time of a project, PHP technology is better solution for small and mid projects. In the complex projects JSP technology will perform better than PHP.

7. References

- Chunlin, L. (1999). *A Java-based Method For Developing Web Application System*. IEEE International Conferenece
- Henderson-Seller, B. & Unhelka. (2000). *Open modeling UML*. Boston: Addison Wesley.
- Masoud, F. A., Halabi, D. H. and Halabi D. H. (2006). *ASP.NET and JSP Frameworks in Model View Controller Implementation*, IEEE,
- Patzer, A. (2002). *JSP Examples and Best Practices*. NY:Apress.
- Roberts, G. (2005). *Learning Server-Side Scripting*, ProQuest Computing, pp. 37
- Suzumura, T., Trent, S., Tatsubori, M., Tozawa, A. and Onodera, T. (2008). *Performance Comparison of Web Service Engines in PHP, Java, and C*, IEEE International Conference on Web Services,
- Tonella, P. and Ricca, F. (2002). *Dynamic Model Extraction and Statiscal Analysis of Web Applications*, in Proc of the 4th International Workshop on Web Site Evolution, pp. 43-52.
- Visualbuilder.com Team (2001), *JSP Tutorial*, <http://www.visualbuilder.com>
- Welling, L., Thomson, L. (2005), *PHP and MySQL Development (pp.4-6)*. Indiana:Sams
- Yang, J.-T, Huang, J.-L, Wang, F.-J, and Chu, W.C. (2002). *Constructing an Object- Oriented Architecture for Web Application Testing*, Journal of Science and Engineering 18(1):59-84