

Análisis comparativo de las herramientas de programación Web: PHP, ASP y JSP, bajo los sistemas operativos Linux y Windows

Daladier Jabba Molinares *, Adalgisa Alcocer Olaciregui **,
Carmenza Rojas Morales ***

Resumen

La programación en la Web ha generado como consecuencia la creación de varias herramientas de desarrollo, por lo que es importante identificar cuáles ofrecen un mejor rendimiento y bajo qué Sistema Operativo. Las herramientas de programación Web analizadas fueron: PHP, ASP y JSP, bajo los sistemas operativos Linux y Windows utilizando criterios comunes.

Se llevó a cabo un estudio descriptivo - deductivo; se desarrolló un prototipo en el que se muestra el funcionamiento de las herramientas mencionadas, con bases de datos bajo Windows/Linux. Se diseñó y desarrolló una página Web prototipo, y se la implementó en PHP, ASP y JSP. Para esto se instalaron los servidores Web IIS, PWS, Apache, Tomcat e Instan ASP para realizar las pruebas, teniendo en cuenta las siguientes variables: Tiempo de respuesta, complejidad de la programación, integridad de la base de datos, arquitectura de software y hardware, detección de fallas, confiabilidad y portabilidad.

Palabras claves: PHP, ASP y JSP, Windows 2000, Linux Red Hat, Prototipo, Access, MySQL, Apache, Html.

Abstract

Web Programming has generated in consequence the creation of several development tools, therefore is important to identify which offer good performance and under Operative System. The Web programming tools analyzed were: PHP, ASP and JSP, using the Operative System Linux and Windows with a common criterias.

It was a descriptive trial, developing a prototype for PHP, ASP and JSP, in Windows/Linux. A web page was made with that tools. The web servers Web IIS, PWS, Apache, Tomcat e One ASP were installed to see the next elements: Answer time, complexity,

Fecha de recepción: 5 de marzo de 2004
Fecha de aceptación: 29 de noviembre de 2004

* Ingeniero de Sistemas, Universidad del Norte; Magister en Ciencias Computacionales del convenio ITESM-CUTB. Docente del Departamento de Sistemas, Universidad del Norte, Barranquilla (Colombia). djabba@uninorte.edu.co

** Ingeniero de Sistemas, Universidad del Norte, Barranquilla (Colombia). aadalgisa@hotmail.com

*** Ingeniero de Sistemas, Universidad del Norte, Barranquilla (Colombia). Cari_roj@hotmail.com

data base integrity, software and hardware architecture, fails detection, trustworthy and portability.

Key words: ASP, JSP, PHP, Windows 2000, Linux Red Hat, prototype, Access, MySQL, Apache, Html.

1. FORMULACIÓN DEL PROBLEMA

Largo ha sido el camino evolutivo del manejo de la información en internet para llegar a la utilización de sistemas de Bases de datos. La humanidad ha sido testigo de la forma en que los *websites* pioneros han pasado a ser los maravillosos sitios interactivos de hoy en día.

Este cambio se ha dado como respuesta a la aparición de nuevos problemas. Ha sido necesario estructurar los contenidos informativos, bajo un esquema cliente/servidor cada vez menos centralizado, promoviendo la especialización de búsquedas de acuerdo a las diferentes categorías, todo esto de una forma simplificada. Bajo este contexto, las bases de datos son parte del eje central de los *websites*, y gran parte de su demanda radica en la producción de contenidos, de tal manera que se puedan generar páginas HTML. Para conseguir esto, existen formas variadas, dentro de las que se destaca la programación de propósito general y herramientas de script.

Sobre el último punto existe la necesidad de saber elegir cuál herramienta utilizar, ya que hay varias como Server Pages™ (ASP), JavaServer Pages™ (JSP), Personal Home Pages (PHP), Cold Fusion, etc., y muchas veces esta elección se torna difícil, debido a que cada una tiene sus ventajas y desventajas, así como diversos usos. Por ejemplo, ASP provee el ambiente de desarrollo más fácil, mientras que JSP requiere el conocimiento de Java y Servlets. PHP es una tecnología popular basada en una fuente de desarrollo abierta. ASP se acomoda mejor para sitios pequeños que tengan entre 100-500 consultas por día, mientras que JSP es capaz de manejar entre 100 y un millón de consultas por día.

Debido a lo anterior, se hace imprescindible comparar de una forma dinámica, a nivel cualitativo y cuantitativo, las herramientas de programación ASP, JSP y PHP, ya que los documentos encontrados que comparan dichas herramientas se limitan a describir elementos (como el costo, la escalabilidad, servidor empleado, etc.) que si bien son importantes, pueden ser pocos para que el potencial usuario haga su elección de acuerdo a sus necesidades.

2. METODOLOGÍA

La investigación se desarrolló de la siguiente manera:

- Instalación de los sistemas operativos: Windows 2000 y Linux Red Hat 7.
- Instalación de los servidores Web Apache, Apache Tomcat y Internet Information Server en Windows 2000; en Linux: los anteriores, pero a diferencia de IIS, se trabajará One Active Server Page.
- Instalación de JDK y PHP.
- Diseño de la base de datos.
 - Creación del Modelo Entidad Relación
 - Creación del Modelo Relacional
 - Creación de tablas en la base de datos Access y My SQL
 - Desarrollo de prototipo en ASP, JSP y PHP.
- Desarrollo de las pruebas, ejecutando los servidores, haciendo llamados a través de Internet Browser, de las páginas de cada prototipo.

Las variables tipo cuya posible respuesta es un si o un no, se midieron por observación directa. Para evaluar la portabilidad, se colocaron los códigos fuentes en cada servidor y sistema operativo sin ningún cambio, así: migrar de Linux [6] a Windows sin alterar el código y viceversa, y migrar de un servidor Web hacia otro sin hacer modificaciones en el código. El tiempo de respuesta se midió desde el inicio del código de cada herramienta hasta el fin, tomando el tiempo al inicio y al final de cada ejecución, y de esta forma se obtuvo la diferencia.

En cuanto a la complejidad, se tomaron como líneas de código todas aquellas que representaban una instrucción, excepto las declaraciones y las llaves de los ciclos repetitivos.

Por otro lado, para la Integridad de la base de datos se realizaron pruebas de concurrencia, en las cuales se intentó modificar, eliminar o agregar un registro simultáneamente a la base de datos, observándola antes y después de esta operación para determinar si se mantuvo la integridad de los datos.

3. PRESENTACIÓN Y ANÁLISIS DE LOS RESULTADOS

3.1. PORTABILIDAD EN LOS SISTEMAS OPERATIVOS

Las tres herramientas son portables de Windows a Linux y viceversa; esto quiere decir que se puede migrar de un sistema operativo a otro sin realizar cambios en el código. Sin embargo, es importante destacar que ASP [3] no fue diseñado para trabajar en ambientes Linux, ya que es un producto exclusivo de Microsoft. Debido a esto, la empresa SUN desarrolló una herramienta llamada One Active Server Page, la cual interpreta el código ASP y permite el funcionamiento de éste bajo Linux. Esto tiene algunas limitantes para los usuarios Linux, ya que la herramienta no es de tipo Freeware, lo cual reduce el uso de ASP sobre Linux y en caso de que el programador la utilice, si desea migrar de un sistema operativo hacia otro, debe asegurarse de utilizar componentes ADO para la conexión con las bases de datos.

Tabla 1
Portabilidad en los servidores, según los sistemas operativos

Herramienta	Apache		IIS		Tomcat		OneASP(Modulo)	
	Win 2000	Linux	Win 2000	Linux	Win 2000	Linux	Win 2000	Linux
PHP								
ASP								
JSP								

Fuente: Pruebas realizadas con PHP, ASP y JSP en Windows y Linux por el grupo investigador.

Como se observa en la tabla anterior, en el servidor Apache, tanto PHP como JSP (a partir de la versión 4) son portables de Windows 2000 hacia Linux Redhat y viceversa. Se puede apreciar que ASP sólo puede ejecutarse en IIS en Windows y en Linux sólo si se agrega un *software* intérprete como modulo de Apache, en nuestro caso One Active Server Page. Por otro lado, JSP es portable en ambos sistemas operativos bajo el servidor Tomcat. Es de destacar que PHP puede ejecutarse en IIS.

3.2. CONFIABILIDAD

Al ejecutar los prototipos desarrollados con las tres herramientas, se pudo observar que al seleccionar la opción *Ver código fuente*, disponible en el navegador de internet, ninguna de las herramientas muestra el código del

lado del cliente, lo cual es algo favorable desde el punto de vista de la seguridad. A pesar de lo anterior, ASP está sujeto a la decisión, o al desconocimiento, de parte del programador, para que el usuario vea o no el código, debido a que esto depende de las etiquetas que se utilicen. Por ejemplo, la etiqueta `<script></script>` permite que sea visible, en tanto que `<%%>` no.

3.3. ARQUITECTURA DE SOFTWARE Y HARDWARE

Tabla 2
Arquitectura de *software* y *hardware*

Características necesarias para un funcionamiento adecuado	Herramientas		
	PHP	ASP	JSP
Sistema Operativo	Linux	Windows	Linux
Servidor	Apache	IIS	Tomcat
Memoria	128 o más	128 o más	256 o más

Fuente: Pruebas realizadas con PHP, ASP y JSP en Windows y Linux por el grupo investigador.

A pesar que PHP y JSP ejecutan correctamente bajo Windows, su funcionamiento adecuado se da en Linux, debido a que inicialmente fueron diseñados para este último sistema operativo, y como era de esperarse, ASP tiene un mejor desempeño en Windows, debido a que es una herramienta exclusiva Microsoft. De igual forma sucede con el manejo de los servidores sobre los cuales se almacenan las páginas desarrolladas en los mismos. Por otro lado, hay que destacar que JSP [4] requiere una memoria más grande, debido a que está basado en JAVA, y consume gran cantidad de recursos de memoria.

3.4. DETECCIÓN DE FALLAS

Se ejecutaron varios procesos paralelos en cada una de las herramientas de programación bajo la Web, y se encontraron los siguientes resultados:

Tabla 3
Grado de detección de fallas

Detección de fallas	Herramientas		
	PHP	ASP	JSP
Óptimo			
No óptimo			

Fuente: Pruebas realizadas con PHP, ASP y JSP en Windows y Linux por el grupo investigador.

En la tabla anterior se puede observar que PHP tiene la celda parcialmente sombreada; esto quiere indicar que dicha herramienta no detecta por completo qué tipo es el que se produce.

Tabla 4
Calidad de fallas detectadas

Herramienta	Hay error+ Ubicación	Tipo de error
PHP	135 (90%)	23 (15%)
ASP	83 (55%)	69(46%)
JSP	143 (95%)	140 (93%)

Fuente: Pruebas realizadas con PHP, ASP y JSP en Windows y Linux por el grupo investigador.

Para los datos desplegados en la tabla anterior se tuvo en cuenta la ejecución de procesos que generaban 150 errores en promedio. JSP fue el que tuvo la mejor calidad de detección, ubicación de errores y del tipo de error. En cuanto a PHP [2], posee una buena detección y ubicación de errores, sin embargo tiene problemas en referenciar el tipo de error. ASP detectó el 55% (Hay error + ubicación del error) y el 46% (tipo de error), mientras que JSP el 95% y 93% respectivamente.

Aunque JSP muestre el mejor resultado en cuanto a la detección de error, se debe anotar que la correcta identificación de los mismos sólo esta garantizada cuando estos son interpretados por un usuario avanzado.

Claramente podemos ver que para el caso de detección de fallas, la mejor herramienta que se puede utilizar es JSP, en segundo lugar PHP y por último ASP.

Tabla 5
Integridad de la base de datos

Herramientas	Integridad	
	Windows	Linux
PHP	132 (88%)	141 (94%)
ASP	71 (47%)	68 (45%)
JSP	67 (46%)	74 (49%)

Fuente: Pruebas realizadas con PHP, ASP y JSP en Windows y Linux por el grupo investigador.

Como se observa en la tabla anterior, de 150 pruebas, PHP mantuvo la integridad en base de datos en el 88% de las veces en Windows, y el 94% en Linux. El resultado de esto fue mayor en Windows que en Linux para ASP (47% Vs 45%), a diferencia de JSP (46% Vs 49%). Sin embargo, a pesar que PHP mantuvo la integridad en base de datos en un gran porcentaje en ambos sistemas operativos, en contraste con las otras herramientas, que no llegaron a la mitad, la diferencia no fue estadísticamente significativa (p: 0.95505).

Según los datos obtenidos en el manejo de integridad de bases de datos, PHP es la herramienta más robusta, en segundo lugar JSP y por último ASP; aun en su propio ambiente de trabajo que es Windows, ASP no es la mejor solución si se tiene en cuenta la este punto.

3.5. COMPLEJIDAD DE LA PROGRAMACIÓN

En la siguiente tabla se procederá a mostrar cuál es la herramienta más beneficiosa teniendo en cuenta la complejidad en la programación y según la información resultado obtenida de las pruebas desarrolladas por el grupo de investigación.

Tabla 6
Complejidad en la programación

	PHP	ASP	JSP
ACTUALIZACIONES			
Artículo	41	43	66
Cliente	39	39	70
Vendedor	39	39	73
INSERCCIONES			
Artículo	15	18	31
Cliente	12	18	31
Vendedor	12	19	37
CONSULTAS			
Artículo	31	48	71
Cliente	31	40	66
Vendedor	31	40	68
Listado	11	20	28
ELIMINACIONES			
Artículo	29	69	74
Cliente	28	53	71
Vendedor	28	60	70
VENTA	231	291	299

Fuente: Pruebas realizadas con PHP, ASP y JSP en Windows y Linux por el grupo investigador.

Se realizaron pruebas sobre un prototipo de *software* que tenía los módulos de cliente, vendedor y artículos.

En cuanto a las actualizaciones, no se observó diferencia estadísticamente significativa entre PHP y ASP ($P = 0.87843$), entre PHP y JSP ($P = 0.89323$), ni entre ASP y JSP ($P = 0.47717$).

En cuanto a las inserciones, no se observó diferencia estadísticamente significativa entre PHP y ASP ($P = 0.58381$), entre PHP y JSP ($P = 0.38218$), ni entre ASP y JSP ($P = 0.76039$).

En cuanto a las consultas, no se observó diferencia estadísticamente significativa entre PHP y ASP ($P = 0.97148$), entre PHP y JSP ($P = 0.38218$), ni entre ASP y JSP ($P = 0.91691$).

En las eliminaciones no se observó diferencia estadísticamente significativa entre PHP y ASP ($P = 0.73130$), entre PHP y JSP ($P = 0.9487$), ni entre ASP y JSP ($P = 0.71204$).

Los resultados anteriores estaban dentro de lo esperado, debido a que los prototipos fueron diseñados con las mismas especificaciones, por lo cual, aunque en algunos gráficos se observe mayor número de líneas entre una herramienta y otra, no se puede sugerir asociación estadística, por tanto, se puede decir que las aplicaciones tienen la misma complejidad (complejidad en número de instrucciones que se va a ejecutar).

El incremento de líneas entre uno y otro se atribuye a que, por ejemplo, JSP [4] exige la utilización de instrucciones como `try{ } catch{ }`, que pueden generar diferencias entre una herramienta y otra, pero esto no quiere decir que dichas instrucciones no puedan utilizarse en PHP y ASP, sino que su uso no es obligatorio.

La siguiente tabla despliega los tiempos de ejecución de cada una de las herramientas para los comandos de actualización, borrado, consulta e inserciones sobre el mismo prototipo de *software* mencionado con anterioridad.

Tabla 7
Promedios de Tiempos de respuesta

	PHP		ASP		JSP	
ACTUALIZACIONES						
	Linux	Win2000	Linux	Win2000	Linux	Win2000
Artículo	0,0079	0,3949	0,3124	0,2121	0,0029	0,1772
Cliente/Vendedor	0,0081	0,4270	0,3902	0,4996	0,0036	0,1807
INSERCIONES						
Artículo	0,0070	0,2262	0,2456	0,1057	0,0028	0,0677
Cliente/Vendedor	0,0109	0,2212	0,1074	0,1030	0,0052	0,0458
CONSULTAS						
Artículo	0,0056	0,1430	0,1244	0,1007	0,0024	0,0927
Cliente/Vendedor	0,0061	0,1975	0,2596	0,3096	0,0034	0,0949
ELIMINACIONES						
Artículo	0,0314	0,6694	0,1123	0,1612	0,0104	0,2162
Cliente/Vendedor	0,0314	0,3378	0,5352	0,5184	0,0122	0,1744
VENTA	0,0398	0,3945	0,2860	0,3005	0,0181	0,1100
CONSULTA DE LISTADO	0,2228	6,3581	0,9455	7,7026	0,0324	0,1126

Fuente: Pruebas realizadas con PHP, ASP y JSP en Windows y Linux por el grupo investigador.

Los tiempos de respuesta se promediaron, y encontró que en Windows y en Linux, los menores tiempos de actualización, inserción, consultas y eliminaciones correspondieron a JSP. Se hallaron diferencias estadísticamente significativas ($p < 0.05$) entre esta herramienta y las demás, en la mayoría de subvariables, con excepción de las actualizaciones en Windows ($p: 0.14367$) y Linux ($p: 0.5181106$), inserciones ($p: 0.5297860$) y consultas en Linux ($p: 0.4189811$), y eliminaciones en Linux ($p: 0.3047091$) al compararse con PHP. 0.5181106 .

De la misma manera, se observa que el mayor promedio de tiempo de ventas fue para PHP en Windows y ASP en Linux, y el mayor promedio de consultas de listado fue para ASP en ambos sistemas operativos.

Por otro lado, se comparó el tiempo de respuesta de cada una de las herramientas, en Windows y Linux. Se encontró que no existen diferencias estadísticamente significativas para las subvariables en ASP, ni en las eliminaciones para PHP ($p < 0.05$). En el resto de los casos se observó un mejor tiempo de respuesta en Linux.

CONCLUSIONES

Es necesario mencionar que no se puede afirmar tajantemente que una herramienta sea mejor que la otra; lo que se comprobó es que cada una posee fortalezas y debilidades ante determinadas circunstancias. Por ejemplo, JSP presentó los mejores tiempos de respuesta y tuvo la mejor calidad de detección y ubicación de errores, con el tipo de error, pero PHP mantuvo la integridad en base de datos, con mayor frecuencia, en los dos sistemas operativos, y en ASP se obtuvo el mayor promedio de consultas de listado tanto en Windows como en Linux.

También existen diferencias cualitativas que le brindan al usuario una baraja amplia para decidir qué herramienta utilizar en el caso de la portabilidad, además de los costos, porque PHP y JSP cuestan menos, porque los servidores sobre los cuales funcionan óptimamente son de código abierto (*open source*), ya que tanto PHP como JSP son portables de Windows 2000 hacia Linux Redhat y viceversa, ASP sólo puede ejecutarse en IIS en Windows y en Linux sólo si se agrega un *software* intérprete como módulo de Apache; JSP es portable en ambos sistemas operativos bajo el servidor Tomcat, y PHP puede ejecutarse en IIS. Hay que anotar que el funcionamiento más adecuado para PHP y JSP se da en Linux, porque fueron diseñados con la intención de ejecutarse bajo este sistema operativo, así como ASP tiene un mejor desempeño en Windows, por ser una herramienta exclusiva Microsoft. Por otro lado, hay que destacar que JSP requiere una memoria más grande, debido a que está basado en JAVA, por lo cual consume gran cantidad de recursos de memoria.

Sin embargo, así como se presentan estas diferencias, también es sumamente rescatable que las tres herramientas pueden migrar de Windows a Linux y viceversa, sin realizar cambios en el código, y que ninguna muestra el código del lado del cliente, brindando de esta manera mayor seguridad.

GLOSARIO

- APACHE: Servidor web, diseñado principalmente bajo el sistema operativo Linux, posee compatibilidad con otros sistemas operativos.
- ASP: Active Server Page. Es una tecnología propietaria de Microsoft. Se trata básicamente de un lenguaje de tratamiento de textos (scripts), basado en Basic, y que se denomina VBScript (Visual Basic Script).
- CGI: Common Gateway Interface. CGI es una norma para establecer comunicación entre un servidor web y un programa, de tal modo que este último pueda interactuar con Internet. También se usa la palabra CGI para referirse al pro-

grama mismo, aunque lo correcto debería ser script.

- CHILISOFT: Es una compañía que ofrece la tecnología ASP a través de `chilisoft.asp`, es el mismo funcionamiento que ASP de Microsoft, pero corre en sistemas operativos como UNIX, LINUX, SOLARIS, entre otros.
- HTML: Hypertext Markup Language (Lenguaje de Marcado por Hipertexto). HTML es el lenguaje con el que se definen las páginas Web. Básicamente se trata de un conjunto de etiquetas que sirven para definir la forma en la que presentar el texto y otros elementos de la página.
- HTTP: Es la abreviatura de Hypertext Transfer Protocol (Protocolo de Transferencia de Hipertexto). Es un conjunto de reglas, o protocolo, que gobierna la transferencia de hipertexto entre dos o más computadoras. La World Wide Web agrupa el universo de información que está disponible vía HTTP.
- IIS: Internet Information Server. Servidor Web Microsoft para los sistemas operativos Windows NT, Windows Workstation, Windows 2000 server, Windows XP.
- INSTANT ASP: Es un *software* que actúa como intérprete para el funcionamiento de ASP en plataforma LINUX/UNIX.
- JSP: Java Server Page. Es una extensión de la tecnología Java Servlets. Creado por *Sun Microsystems*
- PHP: Hypertext Processor. Es un lenguaje interpretado de alto nivel embebido en páginas HTML.
- PWS: Personal Web Server. Es un producto de Microsoft que viene junto con algunos productos de la casa como Windows 98 o Frontpage 98. Se trata de un servidor de páginas web personal que puede ser utilizado por un máximo de hasta 10 usuarios de forma concurrente. Puede utilizarse para probar el funcionamiento de nuestra Web sin necesidad de publicarla en Internet o para formar una pequeña intranet corporativa.
- TOMCAT: Servidor Web de Apache Group, para el manejo de páginas JSP.
- SERVIDOR
- WEB: Sitio en el que se alojan las páginas Web.
- SCRIPT: Conjunto de instrucciones que se ejecutan paso a paso, instrucción a instrucción.
- JAVA: Es un lenguaje de programación, de alto nivel y orientado a objetos utilizado principalmente para programar en Internet o intranets.
- LINUX: Es un sistema operativo de libre distribución basado en UNIX.
- VBSRIPT: Es un lenguaje de scripting utilizado para agregar funcionalidad a las páginas Webs, el cual fue adaptado para poder generar archivos ejecutables que faciliten ciertas funciones del sistema operativo, si el usuario lo desea. El mismo está basado en el lenguaje de programación Visual Basic, y es desarrollado por Microsoft.
- CLIENTE/
SERVIDOR: Es el procesamiento cooperativo de la información por medio de un conjunto de procesadores, en el cual múltiples clientes, distribuidos geográficamente, solicitan requerimientos a uno o más servidores centrales.

Bibliografía

- [1] COMPARISON BETWEEN SERVER-SIDE SCRIPTING TECHNOLOGIES JSP, ASP, PHP. Tomado de <http://216.239.53.100/search?q=cache:SMCxSaA3JKgC:www.csc.liv.ac.uk/~u1bbj/CGI.doc++%22php%22++%22asp%22++%22jsp%22+++comparison&hl=en&ie=UTF-8>
- [2] MCCARTY, W. (2001). *PHP 4: A Beginner's Guide*. Osborne. McGraw-Hill, p. 544.
- [3] MERCER, D. (2001). *Fundamentos de Programación en ASP 3.0*. McGraw-Hill, Interamericana, p. 573.
- [4] BOLLINGER, G. & NATARAJAN, B. (2001). *JSP: A Beginner's Guide*. McGraw-Hill.
- [5] SILBERSCHATZ, A., KORCH, H. F. & SUDAR, S. (1998). *Fundamentos de Bases de Datos*, 3ª ed. McGraw-Hill, Abraham, España.
- [6] Linux Red Hat. www.redhat.com