

Server-Side Scripting With Databases using ASP, PHP, Cold Fusion and Java Servlets

EDUARDO CARRILLO ZAMBRANO

Facultad de Ingeniería de Sistemas – Laboratorio de Cómputo Especializado

Universidad Autónoma de Bucaramanga

Calle 48 No. 39 - 234, Bucaramanga

COLOMBIA

Abstract: - The development of Dynamic Web Applications constitutes a mandatory study topic for professionals and students involved with the software development, keeping in mind that organizations request its presence in the Web with the purpose of developing electronic commerce, as well as the implementation of corporate Intranets or Web Applications in general. In this work a description of the elements involved in the development of Web applications is made that include connectivity to databases, the available solutions for different server platforms, different development technologies to be used, its characteristics, functionality, advantages and disadvantages under different perspectives, including examples of connectivity with databases.

Keywords: - Web Applications, Dynamic Applications, ASP, CGI, ISAPI, PHP, JSP, Cold Fusion

1. Introduction

This paper presents a review of different solutions to develop server side scripting. First, the main components of the web applications are presented. In the next section the first kind of server side solutions- CGI (Common Gateway Interface) is presented, and then, different characteristics of the ASP (Active Server Pages), PHP (Professional Home Pages), Cold Fusion and Servlets-JSP (Java Server Pages) are exposed, including scripts that present interaction with databases.

2. Web Applications

A Web application is a software stored in a Web server Web and that can be executed through a Web browser in an Intranet or in Internet under a Client/Server architecture .

The client makes use of a Web browser, by means of which invokes to a Web page located in a server, which executes programs that receives requirements and returns responses, acting like a connection layer between the client and the server.

In de development process, once the specific problem that will be to solve with the

application Web is understood, the necessities of information should be analyzed, and decisions should be made that will depend mainly on the following factors:

- Hardware, operating system and Web server.
- Database Management System (DBMS).
- Middleware and development architecture.
- Development tools for user's interface through Web pages. Web documents can be developed: coding in HTML (Hyper Text Marking Language), XML (Extensible Markup Language), or WML (Wireless Markup Language)
- External programs (scripts CGI, PHP or ASP) or Dynamic Linking Libraries (DLL) under ISAPI architecture .
- Mechanisms of security that will be implemented.
- Plugins (Some solutions can require the previous installation of plugins, software that should be downloaded by the client to be able to execute other programs)
- Browsers that will be used in client side.

3. Models for Connectivity with Databases

3.1 CGI Model

It was the architecture most commonly used in the first developments that involved access to data stored in Web servers. This solution is based on the existence of external programs in the server, which are invoked by the Web pages, receiving its data (commonly through the standard input) and generating results in the required format.

The architecture CGI manages a standard group of variables to pass information in double via between the server and the browser, allowing the develop of applications with different programming languages like: Perl (Practical Extraction and Report Language), C++, Visual Basic or Delphi, among other, being constituted in one of the first solutions for access to data stored in the server[5].

Disadvantages

- Any change in the database or in the interface will require changes in the programming.
- The main obstacle of this solution resides in the excessive load of the server, because the initialization of the CGI and its load in memory it is carried out so as many times as requests to the server. In addition to, the libraries that have to be initialized before the CGI program can be executed, which degrades the performance and time of answer.

Recently the Apache server allows the libraries to remain in memory along the execution of the CGI, making the following calls not to require access to disk, improving the time of answer and maintaining good performance. Apache allows complete interpreters to be loaded in the Web server when starting up, this way those essential components are always in memory, improving the time of answer again. At this time the existent Apache modules load packages interpreter/compiler PERL and PHP at the starting up[4].

3.2 Server Side Scripting Model

This model is characterized by the inclusion of labels or specific marks inside the documents (HTML, XML or WML) located the server, inside those, orders required for the server to execute are included, after that an HTML (or XML/WML) answer is generated which is understandable for the client's browser (See. Fig. 1).

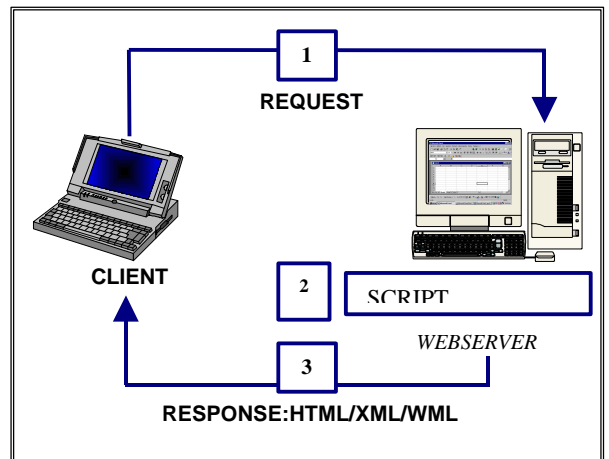


Fig. 1. Server Side Scripting.

The general process that is followed is:

- The browser specifies the URL from the document to be executed.
- A request is sent to Internet or the host of the Web server.
- The Web server executes the script (included inside the labels in the document or in an independent file).
- The script communicates with an external application and it recovers data.
- The script returns the output to the Web server.
- The Web server sends the output from the script to the client's browser.

The main advantages of this type of solutions include the independence in connection with the client's browser; the use of SQL language to get the interaction with databases, and the easiness and short time of test.

Disadvantages

Its main disadvantages reside in the sacrifice in performance because the scripts are interpreted (they require that the server carries out syntactic revision in each execution) and the necessity to modify the code of the scripts when there are changes carried out in the interface.

This model is characteristic of solutions as ASP, Cold Fusion and PHP, and other solutions.

3.2.1 ASP

This is the solution of Microsoft which can be implemented in servers with Windows NT with Internet Information Server of Microsoft, and inclusive for Windows 95/98 with Personal Web Server installed. .

The main characteristics are: [2]

- The access to databases is carried out by means of ODBC and objects ADO (Active Data Objects) that are placed in the scripts.
- Simple of to create and to update, then it does not need too much experience in programming.
- It allows to manage sessions and therefore the storage of variables can remain during different connections.
- It allows the use of different script languages (inclusive combined inside the same file) like: VBScript (subset of Visual Basic), JavaScript, PerlScript.

Disadvantages

- They have been broadly proven in servers IIS of Microsoft, and only until recently they have been begun to develop products for other platforms (as iASP for Linux).
- When being interpreted, performance is sacrificed, because syntactic revision is made in each execution (This restrictive one can be overcome using Web classes, which allow to create compiled objects, but it will be required the acquisition of a programming language that supports this technology additionally)

Next a ASP script is presented, it visualizes data of a database Access, that contains a chart

friends with two fields: telephone and name, for which it has been created a DSN (Data Source Name) previously in the control panel denominated aliasamigos.

```
<HTML>
<Title>Example of Consultation on a database
with ASP </TITLE>
<%
'To validate if an object created connection
exists
IF not IsObject(Session (with)) Then
' To create the connection
Set Session (with) = Server.CreateObject ("
ADODB.Connection ")
'To assign the DSN created with ODBC to the
connection
Session (with) .open " aliasamigos ", "", ""
END IF
'To define the query
query1 = SELECT * FROM friends"
'To execute the query, results are stored in a
recordset
SEP RS = Session (with) .Execute(query1)
'To verify if there were records that satisfy the
query
IF RS.EOF THEN
Response.write (there are not available
records")
ELSE
'To present titles of the columns
Response.write (TELEF. NAMES")
Response.write (" <p>")
'To travel the group of registrations (recordset)
WHILE NOT RS.EOF
'To visualize the columns
Response.write (RS (" TELEPHONE "))
Response.write (" " + RS (IT NAMES))
Response.write (" <p>")
'To go to the following record
RS.MoveNext
WEND
END IF
'To close the recordset
RS.Close
'To liberate the resources
Sep RS =nothing
%>
</HTML>
```

3.2.2 PHP

This is a solution of available public domain in Internet, which can be implemented inside the server IIS modifying the Windows registry, and

it can also be used in Unix/Linux with Apache Web Server installed.

PHP uses a programming language (similar to language C, Java or Perl), which allows to the developers strings handling, wide range of functions for handling of directory, files and images and use of variables with session environment. The same as the previous scripts managers it allows access to databases through ODBC or OLE/DB and it includes specialized support for Oracle and Mysql.

It is possible to make applications using Windows and migrate them to Linux, without changes. But this is only true if the connection to the database is not used with ODBC or OLE/DB (middleware used in Windows). To obtain this possibility of migration the database must be created using MySQL like Database Management System, and making native connections to MySQL with PHP scripts. In this way the application written over Windows can be migrated to Linux without changes.

Their main disadvantages reside in the necessity of learning a new and exclusive programming language for Web developments and the difficulty of finding very specialized developers for this technology.

The next script visualizes data of a database, which the same structure of the previous scripts.

```
<HTML>
<TITLE>Example of Consultation on a
database in PHP </TITLE>
<CENTER>
<?php
/* The connection is created */
$conexion=odbc_connect (" aliasamigos
","", "");
/* the query is defined */
$query=odbc_exec($conexion, SELECT *
FROM friends");
/* The results of the consultation are
Visualized */
odbc_result_all (" $query ");
?>
</HTML>
```

3.2.3. Cold Fusion

This solution of the Allaire company requires the installation of Cold Fusion Server and the inclusion of special labels inside the documents.

The main characteristics are: [8,9]

- It uses the language CFML (Cold Fusion Markup Language), to allow the inclusion of special labels of Cold Fusion that is understood by the server.
- The server of Cold Fusion interprets CFML and passes HTML with appropriate data to the Web server.
- The Web server of Cold Fusion allows remote administration.

Disadvantages

- It requires the learning of a new proprietary programming language.
- The server only runs in platform NT.

The pages developed with Cold Fusion can use the label CFQUERY to define consultations and the label CFOUTPUT to visualize the results. This is a example of a Cold Fusion scrip:

```
<HTML>
<TITLE> Example of Consultation on a
database in Cold Fusion </TITLE>
<CFQUERY NAME = "Consulta"
DATASOURCE = " aliasamigos ">
SELECT * FROM Friends
</CFQUERY>
</CFOUTPUT QUERY = Consulta>
Telefono=#TELEFONO#<BR>
Nombre=#NOMBRE#<BR>
</CFOUTPUT>
</HTML>
```

3.3. Internet Server Application Interface-ISAPI

It is based in the creation of libraries of dynamic connection (DLLs) which can be created in different programming languages. The DLL are loaded in the same space of the server, becoming residents, and executing much quicker that the applications CGI, because they require

less time of use from CPU when not beginning separate processes.

Disadvantages

- It requires a wide experience in programming, especially because it requires programming multithread to manage the different requests, otherwise the server would fall quickly. Other concepts that are required to manage include: concurrence control, synchronization, structured handling of exceptions, sockets programming, protocols, etc

- An error in the programming can cause the fall of the servant and to generate serious consequences [6].

- When changes to the requirements are carried out the DLL must be recompiled, requiring additional dedication to prove that the new routines won't harm the server

- This outline is inappropriate for a company that uses outsourcing services, because much will depend on who develops the project.

3.4. Servlets-JSP: the Java solution

The applets are programs written in Java that once compiled they are executed on the side of the client and they do not have possibility to write in local files, the above-mentioned with the purpose of managing policies of security.

Later on the technology Servlets was developed (applets without graphic interface that are executed on the side of the server) for access to databases using like intermediate layer to JDBC (Java Database Connectivity) [7,11]. JDBC is a similar product to ODBC created by Sun in 1996, like an interface of programming level that allows the Servlets to have access to the databases.

The servlets are independent of the server and Sun's Java Server Development Kit (now Server Development Kit) is required for its development. The servlets must be compiled and they can be directly invoked from the web browser or from a form tag in a HTML document. Also, servlets can be embedded inside HTML pages server-side-include (SSI) functionality, using the SERVLET tag (if the server supports SSI).

Some characteristics of the servlets are:

- They are loaded by the server

- They allow to manage multiple concurrent and synchronized petitions

- They offer independence of the platform (thanks to Java and the API Java Servlet) and they offer a better performance than the CGI.

Disadvantages

- The code HTML is included inside the servlets, this situation force to recompile the servlet when a change in the interface is required.

- They require the use of concepts as exceptions and multitasking programming, for to control the synchronization.

- They require wide experience in Java for their construction.

Another solution of Sun is JSP (Java Server Pages). JSP operates as a server-side includes in a similar way than the previous scripting technologies, for to write servlets it is necessary to enclose the code for the dynamical part in the tags `<%` and `%>` inside the HTML code. The main difference with the servlets is that with JSP is not necessary to compile the code. In the execution process, the JSP page are automatically converted to a servlet the first time that the page is required.

The Java solutions are very portables to different operating systems and web servers, this is a very useful characteristic specially in software development companies that wish to reuse applications for clients with different server platforms.

CONCLUSIONS

In the past the Web programming on the side of the server used very complex bottom technology that was only known by some few expert developers, at the present time this myth has been overcome and great quantity of available solutions exists. Many of the solutions presented mix the regular, static HTML/SML/WML documents with dynamically generated content form scripts.

The webmasters that require handling of databases access will be able to select among models that involve pure edition of code HTML-XML-WML that invokes scripts or libraries of dynamic linking located in the server, tools RAD or a combination of these outlines. In connection with the type Web server to use Apache will be a very attractive alternative, especially keeping in mind that it is of public domain, being able to be combined with PHP or CGI (having a quite marked learning curve). It should be kept in mind that a flaw in a module ISAPI can cause big problems at server level, aspect that is not of common occurrence in the other architectures, being also easier of adapting to changes.

A great obstacle exists in most of the available gateway: in the event of being required a change in the database or in the interface, this change will cause necessity to recompile the programs, situation that is expected to be overcome with XML. The language HTML that has been used traditionally together with the scripts languages to access databases, it tends to be replaced by the standard XML like a great alternative to organize the information that is published in the net.

The decision of the best solution to be used in some Web development depends of many factors and is not possible to generalize what is the most recommendable.

Finally, it is important to quote that it is possible to use the server side scripting technologies presented here in wireless applications to Internet, if the MIME type for WAP is configured at the web server.

References:

[1] AGRANAT, Ian Douglas. Engineering Web Technologies for Embedded Applications. IEE Internet Computing, May-June, 1998. PASG 40-45.

[2] BERTZ, Mark. Active Design Data Objects & ASP. Dr. Dobbs' Journal. Mayo de 1998, Pags 88-91.

[3] BEVERIDGE, Tony, y McGLASHAN, Paul. ISAPI/NSAPI Web Programming. Editorial Coriolis, 1997.

[4] BROWNS, Robert G y HAHN, William. Choices in Server Side Programming A Comparative Programming. ACM, 1999. pags 35-40.

[5] CARRILLO ZAMBRANO, Eduardo. Asp una alternativa a la tecnología CGI. Primeras Jornadas Iberoamericanas de Telemática y Telecomunicaciones, Santa Cruz de la Sierra, Bolivia, 1999.

[6] CHARTE O., Francisco. Aplicaciones Web con Borland Delphi 4. PC World, Septiembre de 1999, Pags 240-248.

[7] CLIP, Paul. Servlets: CGI The Java Way. Byte, May, 1998, Pag 55-56.

[8] GARCIA V., Norman Enrique. Diseño e Implementación del Sitio Web de Egresados de la Facultad de Ingeniería de Sistemas Unab. Tesis de Grado de Ingeniería de Sistemas, 2000. Pag 31-98.

[9] HUTCHINSON Gary, BAUR, Greg. Implementation of a Dynamic Web Database Interface Using Cold Fusion, ACM, 1998, Pags 131-135.

[10] PC MAGAZINE, CREATE A GREAT SITE. Revista PCMagazine, June 8, 1999, Páginas 147-158.

[11] YANG, Andrew, LINN, James, QUADRATO, David. Developing Integrated Web and Database Applications Using Java Applets and JDBC Drivers. ACM, Pag 302-306.

The presentation of this work was financially supported by Fondo Solidario Fortuna and Colciencias (Instituto Colombiano para el Desarrollo de la Ciencia y la Tecnología "Francisco José de Caldas")