Study on the Implementation of Macromedia ColdFusion as a Web Server Application

Joanelle J. Baptiste
Kaiem L. Frink
1704 Weeksville Road, Box 672
Elizabeth City, North Carolina 27909

Abstract - The ONR 2004-2005 Multimedia Research Team studied the implementation of Macromedia's ColdFusion Server software as a web application server for small scale online databases. ColdFusion is utilized by the ECSU staff and integrates with the current Macromedia tools employed by the ONR media developers. The team's goal was to utilize the University server's ColdFusion software for the implementation of a small, startup site. This site involved documenting the Submerged Aquatic Vegetation (SAV) Herbarium of Northeast North Carolina waters for the ECSU Geological, Environmental, and Marine Sciences (GEMS) department. The team developed the initial site with database files in place to begin the recording of information. The future phases of will include the insertion, manipulation, and retrieval of actual gathered data. This project sought to research the learning curve, integration, and ease of use of the ColdFusion software in comparison with other web application software.

I. CERSER

A. *Introduction*

The Center of Excellence in Remote Sensing Education and Research (CERSER) was established in 2003 at Elizabeth City State University in Elizabeth City, North Carolina. This project is a partnership for innovative ocean, coastal, and marine research with ECSU, NASA, NOAA, ONR, SeaSpace Corporation, and the Wakefield Office of NOAA's National Weather Service. The CERSER computer lab consists of a SeaSpace TeraScan HRPT system. The TeraScan HRPT system consists of a 1.5m polar orbiting tracking antenna, Global Positioning System (GPS) Antenna/Receiver, Telemetry Receiver, Workstation, Uninterruptible Power Supply (UPS), and TeraScan Software Suite.

B. *Data Collection*

CERSER collects image data from several satellites passing overhead daily. This information is converted into images and stored in a central location along with information such as:

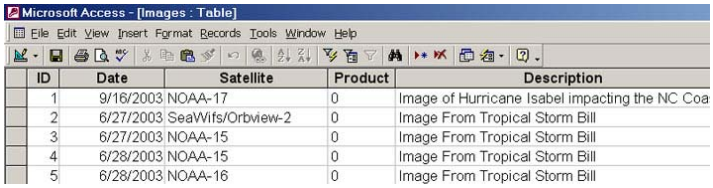
- Date
- Event (hurricane, storm, front, etc…)
- Satellite name (NOAA-17, SeaWifs, NOAA-15, etc…)
- Type of data (Sea Surface Temperature, Water Clarity, Chlorophyll, etc…)
- General Description (such as "image of Hurricane Isabel approaching the US…")

II. THE DATABASE

A. *Types of Data Stored*

The online storage of the satellite information is taken care of by a Microsoft Access Database entitled "terascan.mdb" located on the CERSER server at http://cerser.ecsu.edu.

The data stored is currently text and dates with some text items being URL's to images. The image below illustrates a section of that database.



ID	Date	Satellite	Product	Description
1	9/16/2003	NOAA-17	0	Image of Hurricane Isabel impacting the NC Coa
2	6/27/2003	SeaWifs/Orbview-2	0	Image From Tropical Storm Bill
3	6/27/2003	NOAA-15	0	Image From Tropical Storm Bill
4	6/28/2003	NOAA-15	0	Image From Tropical Storm Bill
5	6/28/2003	NOAA-16	0	Image From Tropical Storm Bill

Figure 01 WEB APPLICATION SERVERS

A. *Definition*

A Web Application Server is a software program that lets the web server perform more tasks, such as interact with a database, serve up customized information based on user preferences or requests, or validate user actions.

B. *Using ColdFusion*

Building a basic site with ColdFusion is accomplished by the following steps:

- Step 1: Build the Database
- Step 2: Place files on the Server
- Step3: A request come in from a client

- Step4: IF .cfm or .cfml are present, the Web Application Server then processes the request
- Step5: IF ColdFusion code is present, it is executed and data is received from the database
- Step 6: Data is returned to the application server
- Step 7: The application server then returns an html page to the web server
- Step 8: The web server returns the requested page to the client

IV. ACTIVE SERVER PAGES

A. Definition

ASP is a specification that enables database-driven Web sites. Web pages that have an .asp extension (instead of an .html or .htm extension) are executed immediately using updated information from the database. This enables pages to be created when they are requested and easier content management. It can also present security problems because it opens "holes" in security to enable information to be accessed and viewed in real time. Ensuring security with active server pages is possible, but it requires specialized Web site development skills to create secure Web sites.

B. CERSER ASP Pages

CERSER utilizes ASP pages to store, document, and retrieve images and their data from the server. This coding was performed by a graduate student familiar with the language. Due to the fact that this student has moved on and there is little support for ASP on the ECSU campus, it is difficult to maintain or improve the CERSER system. Several recent hacks and system failures have prompted the staff to begin considering the long-term maintenance and upgrades of this system. Due to a lack of commenting (explaining the code in “Plain English”), the code is challenging to troubleshoot and troublesome to upgrade.

V. COLDFUSION VS. ASP

A. Why ColdFusion

The ColdFusion Web Application Server software is integrated with the current Macromedia tools (Dreamweaver, Flash, Fireworks) that we and the University currently use. The support and instruction available on the campus is not transitory and will provide a continuity of knowledge that we currently do not have with ASP.

B. Comparisons to be Made

One of the comparisons that can be made between ASP and ColdFusion is the price. Macromedia ColdFusion sells for between $849 for the educational version and $4,999 for the full enterprise edition. ASP, on the other hand, is free. This makes the ASP software attractive to the majority of developers.

The second comparison that can be made is the speed of development. This is the time that it takes to prepare a finished product from beginning to end. ColdFusion Markup Language is both easier for beginners to read. ColdFusion applications can be developed very quickly and are easy to troubleshoot and fix. The error messages sent by ColdFusion when the code doesn't work tend to be specific, which helps reduce development time. ASP is good for experienced programmers who need the flexibility of new technologies.

The third comparison is the amount of coding needed. ColdFusion takes less coding (lines of code) than ASP for the same jobs.

The last comparison is the learning curve. ColdFusion has the shortest learning curve of any language and ASP has a moderate learning curve, focus is shifting to new technologies (asp.net)

VI. OTHER WEB APPLICATION SERVERS

A. Microsoft ASP.NET

There are several other web application servers that are in use today. One of these is a new Microsoft language titled ASP.NET. It is an emerging technology that closely resembles ColdFusion in capability and use. It is utilized on the Microsoft IIS 5 with .NET Framework servers.

Sun JavaServer Pages (JSP) is another popular application that runs on the following servers: Macromedia JRun, IBM WebSphere, Apache Tomcat, and BEA WebLogic. While not as powerful, it does have a greater cross platform

capability.The language PHP Hypertext Preprocessor (PHP) is a free language that is used by those familiar with programming. This language is versatile, but requires a higher degree of competency. It is utilized only with the PHP server.

VII. ColdFusion code

A. Tags and Functions

The last component that makes up the complete structure of Coldfusion is the Coldfusion Markup Language (CFML). The Coldfusion Markup Language uses HTML-like tags embedded in the Web pages. CFML can be highly compared to HTML in that their tags are similar-looking. CFML includes start and end tags in which each tag is enclosed in angle brackets (< >). All ending tags are followed with a forward slash (/) and all tag names are preceded with cf (i.e. <cfstarttagname> </cfendtagname>)

VIII. The roadblock

A. Development Roadblock

During development, it quickly became evident that a major roadblock was going to be access to a ColdFusion enabled server. While most developers agreed that ColdFusion was faster and easier to develop in, they balked at implementing it due to the return-on-investment (ROI). Seeing this, the team decided to turn to the ECSU University Webmaster in this phase of the research. The Webmaster stated that he would allow access to a directory on the university site that was under development for another project (SAV Herbarium) for a connection to the ColdFusion drivers. This aided the project by allowing a smaller project to be implemented permitting a closer look at the development requirements.

B. Gems SAV Herbarium

The SAV Herbarium website was built using Macromedia Dreamweaver software. The website consists of seven pages:

- Home
- Procedures
    - Archiving Specimens
    - Collecting Specimens
    - Preserving Plant Samples for Biochemical Analysis
- Database
- Links

IX. The SAV herbarium

A. Purpose

The purpose of the SAV Herbarium is the collection, preservation, curation, and study of species of submerged aquatic vegetation found within the waters of northeastern North Carolina, including Currituck Sound, Albemarle Sound, and the rivers that flow into these sounds. Collected samples are dried, mounted, archived, and sampled for various biochemical analyses, particularly genetic profiles of naturally occurring SAV. Depending upon their availability, samples of specimens will be available to scientists for research studies upon request. Herbarium specimens will also be available for physical analysis or through the web site.

X. The database

A. Definition

A database is an organized collection of information or data into individual records. Data can be described as a single piece of information, and can include, but is not limited to, text, images, numbers, and media clips. Databases provide easy storage and access of data. Within a database, one can search, sort, and analyze data swiftly with the use of a database program. A database program allows one to make changes to a database by deleting or adding data.

B. Main Parts

The database is divided into three main parts, which are data, field, and record. Any information that one gathers can be considered data. Whatever is entered into a database is classified as data. The second component, fields, is where the data is placed when it is entered into a database. Take an employee directory for example. Field names such as *social security number (SS#)* or *Name* indicates the type of information designated for that specific field. The third component is the record. Just as fields are the columns of a database, records are the

rows in a database. However, the first row may not be called a record if it contains field names. Records will begin with the second row in this case. Being knowledgeable about databases is essential for the implementation of an online database for the ECSU SAV Herbarium website.

C. *SAV Herbarium Fields*

The SAV Herbarium website needs an online database to manage and make available the resources of the herbarium's specimen collection to other scientists and the general public for research purposes. Even more useful to the viewers are the categories of each specimen. These categories include but are not limited to:

a) Field ID
b) Accession number
c) Genus
d) Species
e) Common Name 1
f) Common Name 2
g) Latitude
h) Longitude
i) Date
j) Collected By
k) Mounted By
l) Water Depth, meters
m) Secchi Depth, cm
n) Turbidity, NTU
o) Ph
p) DO
q) ORD
r) Salinity
s) Temperature
t) Sediment Percent Gravel
u) Sediment Percent Sand
v) Sediment Percent Fines
w) Sediment Percent Organic Carbon
x) Field Comments
y) Image

XI. BUILDING WITH COLDFUSION

A. *Build the Database*

The first step in implementing ColdFusion is to build the database. Dr. Fischer and her students provided the fields as mentioned before and will provide future formats for requesting data from them. The next step is to upload the database to the University Server located at www.ecsu.edu. The University Webmaster must then connect the database using the Object Database Connectivity (ODBC) allowing ColdFusion to access and modify the contents.

B. *Data Input Page*

Building the input page is needed for entry of data to enable the database to grow as needed. Modifications and additions will also be needed over time due to changing requirements. To build the input page, the basic Form tag is used to allow for text blanks, buttons, drop-down fields, and other inputs that will hold values for the database. The developer must provide space for each field needed by the customer, which in this case is GEMS. If entering only specific data, an all inclusive entry page may not be needed. The SAV Herbarium has 25 fields making an entry page very long if all were included.

The results page will perform the actual input of the data to the database. It should include feedback on whether the data entry succeeded or not. The code in figure 2 illustrates a simple entry of "Genus" to the database.

```
<form name="inputResultsOne.cfm" method="post">
  <label></label>
  <table>
    <tr>
      <td colspan="4"><strong>Data Entry</strong></td>
    </tr>
    <label>Genus
    <select name="select">
      <option>AA</option>
      <option>BB</option>
    </select>
    </label>
    </td>
   </tr>
  </table>
    <label>
    <input type="submit" name="Submit" value="Submit">
    </label>
</form>
```

Figure 2

C. *Results Page*

The results page actually performs the entry into the database through the ColdFusion code shown in figure 3 Once the data is entered, feedback should be given to ensure the data was entered. In this case, the actual data from the table is returned to the client by the code in Figure 3.

```
<CFQUERY NAME="insert_data" DATASOURCE="herbarium">
    INSERT INTO tableOne(GENUS)
    VALUES
    ('#Form.GENUS#)
</CFQUERY>
```

Figure 2

```
<CFQUERY NAME="select_data" DATASOURCE="herbarium">
    SELECT * FROM tableOne
</CFQUERY>
```

Figure 3

A. Future Work

The project that was initially envisioned took many turns due to money, time, resources, and communication complexities between different University departments. Future projects should look at taking a smaller piece of this project and completing it in a step fashion. The following recommendations can be made for future media research.

Establish a dedicated test server for CERSER translation to ColdFusion allowing the continued use of the current site during development.

Automate CERSER image reconfiguration so that original image resizing can be included in the ColdFusion/ASP scripting. This can be performed using current Macromedia software utilizing JavaScript's which may be modified and executed by the web application software.

Secure the SAV Herbarium input page from unwanted inputs through secure code and databases. The same implementation should be performed for the CERSER system as well.

Future work would also include establishing appropriate queries for the SAV Herbarium. This should be done as a team effort with the GEMS department in order to maximize the utilization of the system.

REFERENCES

[1] Macromedia, Retrieved November 28, 2004 from the World Wide Web: http://www.macromedia.com/

[2] Macromedia ColdFusion Support, Retrieved November 2, 2004 from the World Wide Web: http://www.macromedia.com/examples/cfgettingstarted/experience/index_content.cfm

[3] Macromedia Dreamweaver Support, Retrieved November 2, 2004 from the World Wide Web: http://www.macromedia.com/software/dreamweaver/

[4] HouseofFusion, Retrieved December 12, 2004 from the World Wide Web: http://www.houseoffusion.com/

[5] ColdFusion Search, Retrieved November 3, 2004 from the World Wide Web: http://www.cfsearch.com/