Redesign and Upgrade of the CERSER Website and Satellite Image Library

Jonathan Person, Jameson Jones, Justin Deloatch
1704 Weeksville Road, Box 672
Elizabeth City, North Carolina 27909

Abstract-The Center of Excellence in Remote Sensing Education and Research (CERSER) on the campus of Elizabeth City State University is currently tasked with the responsibility of receiving remotely sensed Advanced Very High Resolution Radiometer (AVHRR) and Sea Wide Field-of-View Sensor (SeaWiFS) data for research projects year round. This data is collected, processed, annotated, and transformed into images in the Tagged Image File Format (tiff). These tiff images are then uploaded to the CERSER archive library server located at http://cerser.ecsu.edu. Once uploaded, they are converted into various resolutions and their information is added to a tracking database maintained with Microsoft Access software. This database provides a searchable means for retrieving satellite image data through various parameters.

The current site and database architecture were developed in 2003 and modified in 2005. New capabilities and site requirements necessitated a redesign of the current site to include a more logical navigation scheme and flexibility to expand as needed. The image database required an update of the over four thousand images, many of which were corrupted or missing information. The database also required the implementation of product categories such as AVHRR color images, SeaWiFS chlorophyll images, and SeaWiFS sea surface temperature, as new satellite licenses have been acquired.

I. INTRODUCTION

A. Overview

The Multimedia Team 2007- 2008 (MMT0708) updated the Center of Excellence in Remote Sensing Education and Research (CERSER) website and image database. For the redesign of the website the team focused on the use of Cascading Style Sheets (CSS) throughout the site for navigation and page layout. Links and pages were revised to provide an easier method to navigate the site with grouped internal and external links. The database contained over 4,000 images when this project began in January of 2008. By mid-March the team had eliminated over 1,000 of these images due to lack of data or corruption in the image. The scripting code was also modified to begin accepting image product information from the CERSER TeraScan image processing system when it becomes available.

II. SITE REDESIGN

A. Introduction

The first step in the redesign process was to examine the flaws to be corrected in the current website. The decision was made to stay away from an all blue scheme substituting a more subtle color scheme that did not overpower the information on the page. This also included utilizing more curves to the graphical interface to provide a more eye-catching look to the page. Another point was to keep the background subtle in order to not overpower the information being delivered by the page. More descriptive sub-headers would also be needed for better navigation of the site.

The navigation of the current CERSER site was confusing and made it hard to find information quickly. Links that had been added over the last few years were not in logical places and shifted from page to page. Major changes in link placements were needed to remove the confusion.

B. The Storyboard

The first step in any design or redesign is to compile the pages and links that will be needed. This is mostly done away from the computer using paper or white board to revise and plan the site along with the links. The end product can be as simple as sticky notes or as elaborate as a project design flow diagram. Once completed, the storyboard provides the designers with an outline for the site.

The storyboard for the navigation of the current site is depicted in the figure 1.
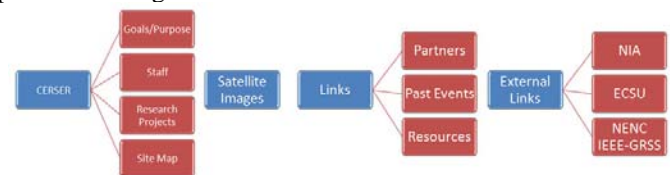

Figure 1

As depicted, not all links were visible at one time. To remedy this, external links and internal links were grouped together. Internal links are those that take the user to a page within the site. External links take the user to another server or address away from the current site. The revised navigation scheme provides the user with a "flat" navigation. The storyboard for the navigation of the new site is depicted in figure 2.
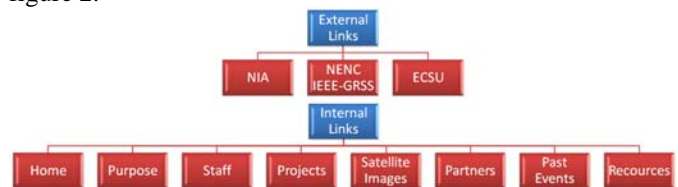

Figure 2

The storyboard also assists with page layout. Web site designers always try to create a layout that is consistent throughout the entire web site and allows the site's visitors to find what they are searching for quickly and easily. Shifting links make navigation tedious and tend to drive users away.

The team's solution to the layout can be viewed in the rough diagram figure 3. Diagrams of this type enabled the team to view multiple variations of navigation and page layout schemes.

Figure 3

III. CASCADING STYLE SHEETS

A. Cascading Style Sheets

Cascading Style Sheets (CSS) were also used on the new CERSER web site. CSS is a simple mechanism for adding a style to a web site easily and efficiently. Styles are used for adding attributes to any tag on an HTML page. Examples of this are font sizes, font families, font weight, background colors, and background images. Once a style is applied throughout a website, the attribute need only be changed in the CSS file to update all pages in the site. CSS also reduces the amount of code in each web page as all attributes are stored in one file.

B. Redefining HTML Tags

Some of the CSS styles used in the CERSER site were redefinition of the HTML tags "links" and "body". The link tag, "a" was redefined to remove the underline except when the user hovers over the link. The following four lines of code can be found in the CSS sheet for the site:

```
A:link {text-decoration: none}
A:visited {text-decoration: none}
A:active {text-decoration: none}
A:hover {text-decoration: underline}
```

The "body" tag was redefined to display a white background, the margins were set to 0 pixels, a background image was added, and the font family and size were defined. The following code redefines the body tag:

```
body {
        background-color: #FFFFFF;
        margin-bottom: 0px;
        margin-left: 0px;
        margin-right: 0px;
        margin-top: 0px;
        background-image: url(images08/bg_01.jpg);
        font-family: Verdana, Arial, Helvetica, sans-serif;
        font-size: 12px;
}
```

C. CSS Classes

The next CSS style that was utilized is classes. Classes can be applied to any tag in the HTML file. In the CSS file the name of the style begins with a period and the attributes are within braces {}. One of the classes employed was ".txtBOLD". This style applies the text attributes of family (Verdana, Arial, Helvetica, sans-serif), size (12), and weight (bold) to any selected tag in the HTML file. The following CSS code applies the attributes:

```
.txtBOLD {
        font-family: Verdana, Arial, Helvetica, sans-serif;
        font-size: 12px;
        font-weight: bold;
}
```

Another class used was ".global_nav-ext". This class applies attributes to link tags. This class is different in that it must be defined for "links", "visited", and "hover". The attributes defined in the style ".global_nav-ext" were font family and size, font color, text decoration, text alignment, font weight, font variant, letter spacing, and the display area. The following code is used to define the attributes of the external links:

```
a.global_nav-ext:link {
        font-family:Verdana, Arial, Helvetica, sans-serif;
        font-size: 11px;
        color:#FFFFFF;
        text-decoration: none;
        text-align: right;
        font-weight: normal;
        font-variant: small-caps;
        letter-spacing: 1px;
        display: block;
}

a.global_nav-ext:visited {
        font-family:Verdana, Arial, Helvetica, sans-serif;
        font-size: 11px;
        color:#FFFFFF;
        text-decoration: none;
        text-align: right;
        font-weight: normal;
        font-variant: small-caps;
        letter-spacing: 1px;
        display: block;
}
a.global_nav-ext:hover {
        font-family:Verdana, Arial, Helvetica, sans-serif;
        font-size: 11px;
        color:#CCCCCC;
        text-decoration: none;
        text-align: right;
        font-weight: normal;
        font-variant: small-caps;
        letter-spacing: 1px;
        display: block;
}
```

D. Implementing CSS in HTML Files

To implement CSS class styles in an HTML file, the following syntax is used: <tag class= "style">. As an example, to apply the style ".txtBOLD" to a table cell a designer would use <td class= "txtBOLD">. Another example would be the application of the "global_nav-ext" style to a

link. This style would be entered as < a href = "index.html" class = "global_nav-ext">.

A. Introduction

The database section of this project involved evaluating the images in the CERSER image library and determining which images were acceptable and which were corrupted or missing data. The two types of images currently in the database library are the Sea Surface Temperature (SST) and Advanced Very High Resolution Radiometer (AVHRR) images. Once it had been determined which images were corrupt or were missing data they were deleted from the Microsoft Access database and the various image files removed from the server.

There were a large amount of images that provided no data to the users of the image database. Once the evaluation was complete, over 1000 images were identified as corrupt or missing data and therefore deleted. Each image was approximately 2 MB in size. Removing these images freed approximately 2 GB of storage space on the server.

B. Microsoft Access

Microsoft Access is a relational database management system from Microsoft that combines the relational Microsoft Jet Database Engine with a graphical user interface and software development tools. It primarily stores very large databases and can sort them and aid you in searching.

A database is a table that gathers all types of information and places it into different categories. The database is composed of fields that hold the same type of information for each record such as name, date, school, etc.

C. TeraScan

Initially remote sensing was designed to observe the Earth's weather in the form of cloud patterns. However, further research on the sensors clearly demonstrated that they could be used for more than just monitoring weather phenomena. Today satellites are used in many applications such as sea surface temperature, sea ice, geology, vegetation, snow cover, fire monitoring and fuel potential.

TeraScan is used for mapping, gridding and calibration of the satellite data. TeraScan remote sensing ground stations are used worldwide with over 450 delivered stations on all seven continents including Antarctica. The system is used for environmental research, meteorology, oceanography, tactical operations, disaster management, and education. CERSER utilizes TeraScan to process data from a series of satellites daily and uploads the resulting images to the CERSER server.

The following is a high end view of the TeraScan processing of data from the satellite:

1. The data is received by the TeraScan system from the satellites
2. TeraScan then annotates and converts the data into an image in the Tagged Image File Format (tiff)
3. The tiff files are then uploaded to the CERSER server

4. The JavaScript code located in the file auto_img_2006.asp then converts the image to JPG format and adds it to the database

The TeraScan antenna at ECSU can be viewed in Figure 4.

Figure 4

D. Determination of Corrupted or Missing Data

The following images are examples of files that were discarded and files that were kept. The determination which files were acceptable was purely a subjective decision not based upon any stated criteria.

E. No Image for this File (Figure 5)

This image was originally a placeholder for image files that were deleted. These images are no longer needed as the database structure has been revised. The JavaScript code no longer requires that the images in the database be named consecutively.

Figure 5

F. Corrupted SST Image (Figure 6)

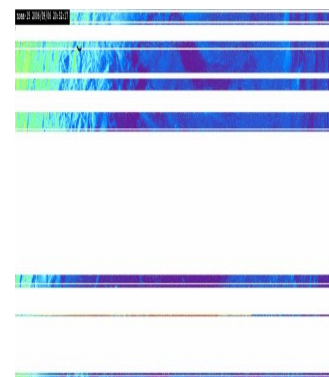This image file is corrupted because the missing data is located in straight patterns.

Figure 6

G. Corrupted AVHRR Image (Figure 7)

The image is corrupted as there is no visible data for a majority of the image and the overlay map of the coast is also missing. This image can not be used because a user has to be able to view a majority of the data to study the Earth's surface and weather patterns.

Figure 7

H. Missing Data (Figure 8)

This image has no data making it unnecessary. The overlay is in view, but without the data the need for the image is irrelevant.

Figure 8

I. Good Image (Figure 9)

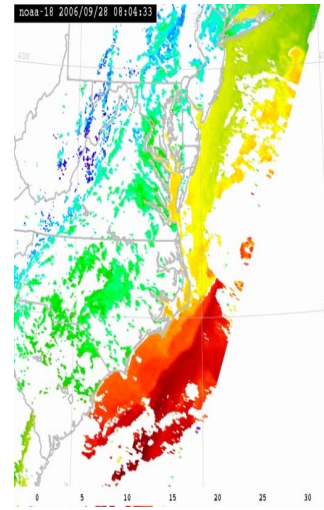This is a good image with a fair amount of data. This image shows large areas of SST data.

Figure 9

J. Great Image (Figure 10)

This image has a great amount of data. This is the type of data that is needed for studies of the ocean's surface temperatures.
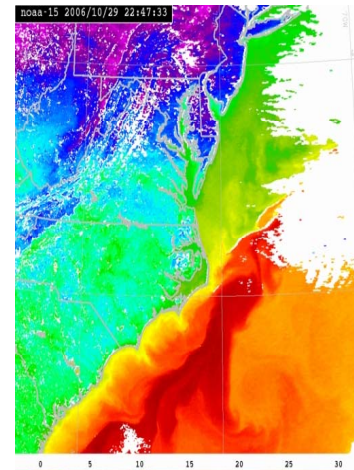

Figure 10

K. Good AVHRR Image (Figure 11)

This is a good AVHRR image with visible data that could be used for studying the weather patterns. This is an image of Hurricane Isabel (2003).
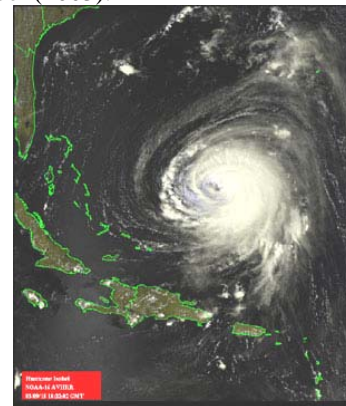

Figure 11

V. Products

A. Defining the Products

During 2007, the TeraScan manager asked that the Active Server Page (ASP) JavaScript be modified to accept "product" types of the various images. "Product" is the term used to define what type of information the image is displaying and what type of sensor is used. The list of product categories requested were:

- NOAA AVHRR Visible Image
- NOAA AVHRR Color Image
- NOAA AVHRR Sea Surface Temperature
- Orbview-2 SeaWiFS Visible Image
- Orbview-2 SeaWiFS Color Image
- Orbview-2 SeaWiFS Chlorophyll Image
- NOAA AVHRR Events
- Orbview-2 SeaWiFS Events

A three letter identifier for each product was identified to be used in the title given by the TeraScan system and read by the JavaScript code in the auto_img_2006.asp file.

- AVI - NOAA AVHRR Visible Image
- ACI - NOAA AVHRR Color Image
- AST - NOAA AVHRR Sea Surface Temperature
- OSV - Orbview-2 SeaWiFS Visible Image
- OSC - Orbview-2 SeaWiFS Color Image
- OSH - Orbview-2 SeaWiFS Chlorophyll Image
- AEV - NOAA AVHRR Events
- OEV - Orbview-2 SeaWiFS Events

B. JavaScript Implementation

The code used to take the product from the file name was written using a series of if-then-else statements. The current file name syntax used for the processed images is: "date.time.satellite.tiff". For example: 080325.2216.n15.tiff would be a TeraScan image processed on March 25, 2008 at 2215 (Greenwich Mean Time) from satellite NOAA-15. The proposed syntax for adding the product would be: "date.time.satellite.product.tiff". An example of this would be "080515.0846.n18.AVI.tiff" denoting this as a NOAA AVHRR Visible Image.

The only other JavaScript command added was the MID() command which locates the desired position in a string of text and then selects a given amount of characters after that point. An example of this is the command "Mid(x.name,17,1)" which was located to find the first character in the product string (x.name) at position 17 of the file name.

The following JavaScript code with line numbers is used to read the product information from the file name:

```
1     'START PRODUCT Conversion
2     if (Mid(x.name,17,1)="a" OR Mid(x.name,17,1)="o")  Then
3       if Mid(x.name, 17,3="avi" Then
4         product="NOAA AVHRR visible image"
5       Else
6        if Mid(x.name, 17,3="aci" Then
7          product="NOAA AVHRR color image"
8        Else
9         if Mid(x.name, 17,3="ast" Then
10          product="NOAA AVHRR sea surface temperature"
11        Else
12         if Mid(x.name, 17,3="aev" Then
13          product="NOAA AVHRR event"
14        Else
15         if Mid(x.name, 17,3="osv" Then
16           product=" Orbview-2 SeaWiFS visible image"
17         Else
18          if Mid(x.name, 17,3="oci" Then
19            product=" Orbview-2 SeaWiFS color image"
20          Else
21           if Mid(x.name, 17,3="och" Then
22             product=" Orbview-2 SeaWiFS chlorophyll image"
23           Else
24            if Mid(x.name, 17,3="oev" Then
25              product="Orbview-2 SeaWiFS chlorophyll image"
26            Else
27              product="0"
28            End if
29           End if
30          End if
31         End if
32        End if
33       End if
34      End if
35     End if
36    else
37            product="0"
38    End if
39    'END PRODUCT Conversion
```

VI. Future work

A. CERSER Upgrades

The CERSER site is currently missing an upcoming events section. This page could be developed easily, but may impede upon the simplicity of the current navigation as there are now eight internal pages. If this page were to be added, the navigation may have to be revised to a drop down menu or a combination of pages.

To make the layout more pleasing to the eye, more images should be added to the body of those pages containing only text. These images should be generic enough that they can be switched out easily in order to keep the site fresh.

B. Image Library

The image evaluation and database maintenance consumed over 40 hours of one team members time. It is proposed that another method of maintenance be developed that would allow deletion of images that are not usable as they are identified. This system would have to be password protected and be available via an HTML or ASP interface on the web.

C. Product Development

Now that the JavaScript code is in place to process file names containing product values, the TeraScan system needs to be modified to output the product codes into the file names. This ideally should be done by the TeraScan manager, but may be within the means of a Multimedia Team project as the naming process is accomplished through UNIX shell scripts.

Another needed maintenance needed is the entering of the product values for past images. This would enhance a user's ability to filter their searches to locate specific data needs. This maintenance will be done by hand versus coding making it tedious and time consuming.

REFERENCES

[1]   "Sea Surface Temperature", Retrieved Feb 13, 2008 from the World
      Wide Web, http://en.wikipedia.org/wiki/Sea_surface_temperature
[2]   "Center of Excellence in Remote Sensing Education and Research",
      Retrieved Jan 12, 2008 from the World Wide Web,
      http://cerser.ecsu.edu/
[3]   "What is a Database?" Retrieved Feb 8, 2008 from the World Wide
      Web,
      http://databases.about.com/od/specificproducts/a/whatisadatabase.htm
[4]   "TeraScan Functions" Retrieved Jan 5, 2008 from the World Wide
      Web, http://marine.rutgers.edu/~kerfoot/terascan/html/
[5]   "NOAA Satellite and Information Service" Retrieved Feb 18, 2008
      from the World Wide Web,
      http://www.class.ngdc.noaa.gov/release/data_available/avhrr/
[6]   "Why use CSS?",
      http://www.adobe.com/devnet/dreamweaver/articles/why_css.html
[7]   "CSS Typography",
      http://www.3point7designs.com/blog/2007/05/09/css-typography-
      taking-back-times-new-roman/