The Center for Research in Technical Communication:
Initial Ideas and Development for a Worldwide Network

http://www.mtu.edu.admin/~dkwalika.tc

Report of Mock Up site

Kirsti Arko
Leroy Steinbacher
John Velat
Dennis Walikainen

Michigan Technological University
December 17,2001
Abstract:
The Center for Research in Technical Communication was a web project to develop a mock up for a site that will eventually work as a network for technical communicators. The site has used some concepts that are used with a portal, so the site has the benefits of a portal without the expense of the software. Some of the software that was used was Clipper Electronic Arts Service, Adobe Photoshop 6.0, Macromedia Dreamweaver 4.0, and Microsoft Excel. Most of the work that was done on the development of the site does not show up on the actual site because it entailed a lot of planning with even more trial and error…

Our contribution to this future professional research site has not led to the final product, but there is now a starting point with some definite ideas for the content and layout. The largest problem that we faced was with the site design. We reversed the steps of web site development; rather than starting with the information for the site, we tried to develop the site without all of the resources. This proved to be a costly mistake but it resulted in a lot of changes and more work.

Table of Contents:
I. Introduction
II. Technical Approach
III. Description of Design
IV. Evaluation of Design
V. Conclusions Recommendations

List of Figures:
I. Website pages
II. XML Data Model for Article Abstract Entries
Introduction

The original intent of this website was not our idea. The work we have done on this website is to carry out a larger project for the MTU Humanities Department. Ultimately, this site is going to be the first step in establishing the technical communication program at MTU. Once the web site is complete, the next step is a technical communication conference for those in the academic and professional field.

The site itself is to fill a void in technical communication research. Presently, the information available covers the pedagogical field, so this site will hopefully fill in the gaps with historical, theoretical and philosophical research in technical communication. Not only will the web site contain articles for research, it will also be a network for those in the field. The available searches will include articles, other informative sites, professionals and academics in the field and universities with a technical communication program.

Our intended audience for this site includes technical communicators in both academics and the professional world. We anticipate undergraduates, graduates, professors and professionals. This audience includes a lot of different backgrounds, but with our diversity, we hope to provide something for all. Anyone with an interest in technical communication will find this web site beneficial for his or her research or mere interest.
At this point in the web site development, we have only just begun the information gathering and design. Our final product is by no means the final product for the site; we have merely begun the long process of developing a professional research web site.

The overall design that we worked with will most likely be the most beneficial for the final product. Being a research site, it is not likely that the site can contain all the information that is available due to resources, so the overall design of the web site is to develop a self-building site. We set up the databases for the searches in all of the fields mentioned, but the goal is to have the users add their own resources to make it as comprehensive as possible. Each field will have the necessary program to allow the users to add themselves, articles, or even other resources.

**Technical Approach**

We decided to use a simple design that would hopefully get the users to the links they wanted to use as quickly as possible.

At first, we were going to emulate or use the portal philosophy as an easy way into the site, but upon further review, we decided to go with straight html look.

Using the Clipper Electronic Arts Service, we took the image of the many mice and imported it into Adobe Photoshop 6.0. There, we first made it one image, saved as a JPG file with separate layers for the different text choices.
Also within Photoshop, we took the individual mice and cropped them, with their text, into separate JPG files for each of the second level pages. Tools we used within Photoshop for these individual images included Unsharpen Mask (30 percent), Bevel and Emboss (75 percent), and Drop Shadow (70 percent). We used these tools for both the background layers and for the text layers.

For the html document, we used Macromedia Dreamweaver 4.0. There, we used the Page Properties tool to choose a blue background color (#66CCFF) complimentary to the main JPG image. For the links within the secondary levels, we used #0066CC, a blue that matches closely the main color; and for the visited links, we used #CC0000, a sharp red, which will contrast sharply from the blue.

For the database pages (abstracts, researchers, institutions), we used a variety of programs, including XML. A description of using XML in the abstract articles follows.

**Abstract Article Entries – XML Method**

There are several different ways of handing data within a website. XML has become a very popular option and has some advantages over traditional web-enabled database interfaces depending on the type of data you are working with. Here are a few of these advantages:

- Markup language that describes semantics (structure and meaning)
- Control (transformation), Interaction (searching and sorting), and formatting of data all within a web browser
- Facilitates e-business and e-publishing (open information integration)
- Editable in any ASCII text editor.
The main reason for choosing XML to handle the article abstract entry data is that it works better for publishing data. XML gives you more control of the format and style of the output and works well with data entries that are one or more paragraphs long. It can also handle stylistic elements such as bulleted and numbered lists that are a little more difficult to work with in a spreadsheet or database entry.

**Technical Design and Development Process**

The following sections describe the design and development process that was followed to create and manipulate the XML data used for the article abstracts. These sections include limitations, achievements, and next steps for each of the components that were worked on.

**Article Abstract Entry Form**

The main goal behind this form was to implement a method that allows users to enter their own article abstract information. This would provide a means for the website to self-generate content and would also make it easier to maintain. The important thing was to make sure that entry fields in the form would correctly match the elements in the XML file. There are several different ways to create forms to enter data over the Internet. The initial form was created in Macromedia Flash because it is very easy to work with and it gives you more control over the style and appearance of the form components. Flash also has several built in scripting functions that work with XML data.

**Limitations**

There were a few limitations to this approach because unless one person is setting up the whole application on their own, it can be difficult to coordinate the functionality with other developers who use different methods for handling data. It can also be difficult to setup external servers to work properly when you need your application to read and write data.
Achievements
A Flash form was created to take the user entered data and send the variables to an Active Server Page using the POST function. The Active Server Page took these variables and created an XML file that was written to my local C:\ drive.

Next Steps
Currently the Active Server Page only creates an XML file with one article entry. The code will have to be modified so that it opens the XML file and appends the new data to the end of the existing file.

The flash form only allows users to enter data. Given more time we could create one that reads data from the XML file and allows users to view it in a number of different ways or perform searches within each of the different fields of information. Macromedia Flash has some very powerful scripting commands that allow XML data to be processed in a number of different ways. We have really only brushed the surface of the potential functionality that is available.

DTD and Cascading Style Sheet for Article Abstract XML file
An Excel spreadsheet was created for all the article abstracts that were sent to us by email. We used this spreadsheet to create a traditional database. The Excel spreadsheet was also saved as a tabbed-delimited text file so the XML tags could be added to the data.

The first step was to setup a structure that could consistently handle the data. Here is a basic outline of the structure that was developed:

```xml
<tcr>
<article>
<title>Visual Rhetoric: A Reader-Oriented Approach to Graphics and Designs</title>
<name>
<last>Kostelnick</last>
<first>Charles</first>
</name>
<journal>The Technical Writing Teacher</journal>
<year>1989</year>
<volume>XVI(1)</volume>
<page>77-88</page>
<terms>visual communication, visual design, visual rhetoric</terms>
<audience>technical writing instructors</audience>
<focus>a model of visual communication with reader-oriented guidelines for the technical writing classroom</focus>

<abstract>The training for technical writing instructors usually prepares them adequately in verbal communication, but lacks in visual communication instruction. Because visuals are central to cognition, students should be taught how to effectively implement and evaluate the
visual component in a technical writing class. The model for this type
of curriculum is a 12-cell matrix for visual communication. Instructors
can combine the matrix and reader-oriented guidelines to teach visual
rhetoric in a technical writing classroom.</abstract>
</article>

Once the structure was in place the next thing to do was to define the rules of the
structure for the DTD. This part is required for loading and manipulating the XML file in
Macromedia Flash or a different XML editing program. Here are the rules that define this
structure:

```xml
<!DOCTYPE tcr [
<!ELEMENT tcr (article+)>
<!ELEMENT article  (title , name+ , journal , year , (terms | audience | volume | page | focus )* , abstract )>
<!ELEMENT title (#PCDATA)>
<!ELEMENT name (last, first, m?)>
<!ELEMENT last (#PCDATA)>
<!ELEMENT first (#PCDATA)>
<!ELEMENT m (#PCDATA)>
<!ELEMENT journal (#PCDATA)>
<!ELEMENT year (#PCDATA)>
<!ELEMENT volume (#PCDATA)>
<!ELEMENT page (#PCDATA)>
<!ELEMENT terms (#PCDATA)>
<!ELEMENT audience (#PCDATA)>
<!ELEMENT focus (#PCDATA)>
<!ELEMENT abstract (#PCDATA)>
]>
```

The next step was to create a CSS (cascading style sheet) to define the style that is
applied to the XML file when opening it up directly in a web browser. The reference to
this file is defined in the beginning of the XML file:

```xml
<?xml-stylesheet type="text/css" href="abstract.css" ?>
```

The abstract.css file specified how each tag should be displayed in a web browser. Here
is the content of that file:

```css
title {
    display: block;
    color: blue;
    font-size: 13pt;
    font-weight: bold;
    margin-top: 1.5em
}

volume, year {
    display: block
}
Limitations
There is a method in the latest CSS spec that allows you to add content that can be used to preface data, such as adding the word “Focus” in front of the content within the `<focus>` tag. This functionality did not work correctly so the preface content was omitted. It may be a syntax error or a browser limitation that could be resolved in a later release.

Achievements
The XML file can now be directly opened and viewed in both IE and Netscape on PC and MAC machines.

Next Steps
Look at some of the other options for displaying the data in this XML file. There are several different options available including XSLT, which should allow users to select fields, or different options on how they want the data to appear. A XSLT file will manipulate the data and generate an HTML file with the user-defined content.

Description of Design
The metaphor for the Center for Research in Technical Communication design is the computer mice and the different areas to which they lead. Since researchers will be viewing the page from their computers, the mice work to reinforce that “technical communication” feel. Within each mouse, then, the text tells where that link goes, and
that particular mouse is repeated on that second-level page. The dominant color from the mouse, then, is used in the text of the second-level page.

The color scheme begins with the background blue, which plays off the blue in the dominant image. The blue would work well with the yellow, green, and purple mice that are present in the dominant image. Again, the links along the bottom of the secondary pages are also presented in this same blue, with the visited link being red to provide a sharp contrast.

We feel researchers will come to the sight and want to search first and foremost, thus, we need to make the links to the secondary pages as obvious as possible. Wording, therefore, was extremely important, as was placement on the dominant image and the links as they were repeated below. We envision easy navigation from the home site to the secondary pages and among the secondary pages. We also view the site as a “meeting place” for researchers from academia and industry, so it was important to include links to their sites. Also, professional organizations are important to researchers, so we wanted to link to them (Society for Technical Communication, Association of Teachers of Technical Writing, E-Server Technical Communication Library, etc.)

Key to the design, and the entire site, is the search capability for each area (researcher, abstract, institution.) Linking to the off-site search pages gives some concern for security and compatibility issues, and these will need to be addressed. The search function is working, however, and that is most important.
Also, the listserv (techcomm-l@mtu.edu) will be very important in beginning conversations among and between scholars and professionals. Housing the listserv on the Center for Research in Technical Communication site will also help drive traffic to the site, we believe.

**Evaluation of Design in Light of Motivation**

Problems arose when the site was moved from one server to another. The images became broken because of the new naming schemata. The Site Folder feature in Macromedia Dreamweaver, which has been created for the current site, will take care of future moves and name changes (in theory).

Given more time (than nature of every Web-beast), more content could have been developed before the site was created. Instead of creating a site that we then needed to fill with content, we could have better planned the look, feel, and metaphor of site, not to mention its usability.

The concept of “design first, content later” is not new. James Souttar addressed it in his article, “The Myth of Content and the Encyclopedestrianization of Communication,” in the book *Graphic Design and Reading*. He said the Web is running the risk of becoming a dumping ground of information in an attempt to become an encyclopedia, and not a very good one at that. More and more, designers are being asked to design Websites before the content is developed. Souttar said this arrangement exists in no other media,
and the best scenario for Web design is enough time (and other resources) to develop meaningful designs, each unique to the specific job at hand. Instead, we are more often than not left with “one size fits all” (p. 178).

Considering this was just a mock-up, we probably did take it as far as it could have gone in the three-week time frame. Like every other Web job, it always takes much longer than originally conceived. Time estimates are not really applicable here, but it would take a good portion of a fifteen-week semester to do a better job of development. As it was, we barely had enough time to bounce ideas off ourselves.

**Conclusions/Recommendations**

Time is the most important resource a Web team can have (and good computers and servers are helpful, too). Under the two- to three-week deadline, we were able to at least assemble a mock-up that was in working order. We were also able to work across a couple of different platforms (Mac and PC) and in different browsers (Netscape and Internet Explorer) with different software (Dreamweaver, Flash, XML language, ASP, Xcel).

There were some glitches in the communications via e-mail. Timely responses didn’t always happen, and confusion reigned a couple of times. But, the team did prove they could work together, and they proved you could, indeed create a site like this.
The next step would be securing the “techcomm.org” domain name (or something similar) and finding people to maintain the current site and to continue to develop the database of researchers, abstracts, and institutions. Such a site would be a great meeting place for technical communication researchers, and it would also reflect very well on Michigan Technological University.
**XML Data Model for Article Abstract Entries**

Article Abstract Entry Form

Active Server Page

abstract.xml

Abstract Search Form

Web Browser

CSS

XML file viewed directly in browser

POST Variables

Adds Markup Tags to Variables

Appends entry to existing abstract.xml

Macromedia Flash

Dynamically generated output

Macromedia Flash

Terms

**SGML:** Standard Generalized Markup Language

**XML:** Extensible Markup Language

**DTD:** Document Type Definition

**XSLT:** Extensible Style Language Transformation