cartographic techniques

MOUSEOVER MAPPING
by James M. Swanson

While Hypertext Markup Language (HTML) works well as a standard for describing page layout and hypertext linkages, it has kept the World Wide Web from becoming a truly interactive medium for maps. Although the multimedia capabilities of today's Web browsers are are comparatively advanced, watching a movie or hearing a soundbite can hardly be considered direct interaction. New developments such as Java and VRML have the potential to change this. This article will concentrate on JavaScript and a technique for making maps more dynamic, called *mouseover mapping*.

JavaScript is Netscape's cross-platform, object-based scripting language for the Web. Do not confuse it with Java. While they are syntactically similar, Java is a compiled programming language used to create small applications or applets that are downloaded from the server and executed on the user's machine. JavaScript, on the other hand, is a scripting language that is actually integrated into the HTML and interpreted at runtime by the browser.

To use JavaScript, it is important to have some understanding of object-oriented programming. In the object-oriented paradigm, an object is a construct with properties that are either variables, other objects, or functions known as methods. When you look at a Web page, think of everything except the text as an object: the forms and buttons are objects, the images are objects, even the page itself is an object. These objects exist in a hierarchy that reflects the structure of the HTML document.

JavaScript is event-driven. Events are actions that occur as a result of something the user does; for example, clicking on a button, selecting text in a field, or moving the mouse over a hyperlink. In many cases, object properties can be dynamically changed within the HTML document. This is very powerful because it liberates the Web page creator from a severe restraint imposed by HTML. Before, you were limited with what the browser was told to display. Now, without having to link to a new page or execute a server-side script, the user can instantly change an image or validate the data in a form.

In mouseover mapping, the user's movement of the mouse over *hot buttons* changes the displayed map or some aspect of it. These buttons are simply hyperlinks which can exist within the map or outside of it. The actual linkage, however, has been disabled and a JavaScript *onMouseOver* event handler has been included.

```
<A HREF="javascript:ignoreClick()"
onMouseOver="swapMap(imageName)"></
A>
```

The first part of this tag tells the browser to do nothing if the user clicks on the hot button. The second part makes a call to a map-swapping function embedded in the document HEAD. Between the open and close of the anchor tag, the hot button can be defined as text, an image, or an image map SHAPE. Remember that the base map image must be inserted into the HTML code before the mouseover links.

```
<IMG NAME="mapName" SRC="path/
fileName" WIDTH=imageWidth
HEIGHT=imageHeight>
```

Since this map image is a JavaScript object, the source property associated with it can be accessed with the following notation:

```
document.images[mapName].src
```

The routine that actually does the swapping of map images is really very simple.

```
function swapMap(imageName)
{
    document.images[mapName].src =
    eval(imageName + ".src")
}
```

The *eval* function performs the concatenation of the two strings and returns the value. If the entire map image does not need to be changed each time, HTML tables can be used to break up your image into discrete parts. However, each section of the map must be given a different map NAME. The original image can be swapped back in when the mouse leaves the hot button by including the *onMouseOut* event handler in the anchor tag above. The syntax will be the same as the onMouseOver event handler, but the two must be separated with a semicolon.

Also, the browser does not access the server to download new images. All of the images that will be used have been preloaded by another short script.

```
{
    imageName = new
    Image(imageWidth,imageHeight);
    imageName.src = "path/fileName"
}
```

Finally, the pieces are put together:

```
<HTML>

<HEAD>
<TITLE>MouseOver Mapping</
TITLE>
<SCRIPT LANGUAGE="JavaScript">
<!—

// TASK: PRELOAD IMAGES
{
imageName = new
 Image(imageWidth,imageHeight);
imageName.src = "path/fileName"
}

function swapMap(imageName)
// TASK: SWAP IMAGES
{
```

```
document.images[mapName].src =
eval(imageName + ".src")
}

function ignoreClick()
// TASK: IGNORE HYPER-REFER
ENCE
{
}

// END—>
</SCRIPT>
</HEAD>

<BODY>

<IMG NAME="mapName" SRC="path/
fileName" WIDTH=imageWidth
HEIGHT=imageHeight >
<A HREF="javascript:ignoreClick()"
onMouseOver="swapMap(imageName)">Change
Map</A>

</BODY>

</HTML>
```

Now the dynamic behind mouseover mapping has been explained and the code foundations are in place, the uses of this form of mapping can be examined. The most evident use is for highlighting. When the user moves the mouse over a geographic region, the area changes shade or color. In the same manner, when the mouse arrives at a particular point on a map, the name can appear next to it. This is especially useful considering the complexity of some maps. *Live legends* would be another use. Depending on the type of map, moving the mouse through the classification hierarchy might display all categories, only bodies of water, or just rivers. For choropleth mapping, hot buttons across the bottom of the map could represent different ways of classifying the data. These buttons could be units of time, too, if maps with temporal components such as weather or erosion are to be displayed.

Finally, mouseover mapping provides a unique solution to the limited resolution of today's monitors. Detail is often sacrificed in order to fit the whole map on your screen. With mouseover mapping, a region of interest can pop up in greater detail on the map as a *close-up bubble* or in another image area, frame, or window. This technique can also be used to display attribute data that corresponds to the mouse's position on the map.

With an understanding of digital image manipulation, a willingness to experiment with JavaScript, and a little imagination, a lot can be done with mouseover mapping. Hopefully, it will be helpful in making maps on the World Wide Web more interactive.

cartography bulletin board

University Cartographic Laboratory Homepages: Marketing Tool or Marginal Presence

by Donna G. Schenström and David C. Wilfahrt
Cartographic Services Laboratory
University of Wisconsin-Milwaukee

Over the past two years the popularity of the World Wide Web has exploded. Colleges and universities throughout North America are placing information on departments, faculties, and research facilities in cyberspace. Many university cartographic laboratories fall into these offerings. This report explores some of the cartography lab sites examining their subject matter, focus, scope, and goals.

The breadth of most pages are greatly influenced by the prominence of Cartography as an area of study within the university, be it housed within Geography or some other Social Science or Earth Science. Page breadth is also affected by each laboratory's mission. Who the lab serves, or who the target client is, greatly influences how the lab's homepage is marketed.

Laboratory pages are also influenced by the role GIS, GPS, Remote Sensing, and multimedia play within the Department and the University. Many cartography laboratories do not stand alone, but rather are combination labs offering a variety of spatial analysis services with technical, digital and informational aspects.

Common Cartographic Laboratory Purposes

Professional map/graphic design and production work for:
• Home department (Geography) faculty
• College faculty
• University faculty
• Outside clients and contracts
• Research
• Student Recruitment
Student map/graphic design and production access for:
• Coursework
• Research
• General experience

Cartography Laboratory Homepage Themes

• Opening page
• Mission statements
• Ongoing research, completed research, work samples (in text or graphic description)
• Facilities listings
• Faculty Use Policy
• Fee Schedules - Pricing
• Guidelines for facility use
• News Articles & Press Coverage
• Faculty & Staff
• WWW Listings, Virtual Libraries
• Map Ordering

Cartography Lab Sites

Memorial University of Newfoundland http://www.mun.ca/GEOGRES/MUNCL.HTM
Ohio State University http://www.cfm.ohio-state.edu
The Pennsylvania State University http://www.deasy.psu.edu
Syracuse University http://www.maxell.syr.edu/geo/cartolab.htm