How-To-Do-It

Making Pages That Move

Applications of Dynamic HTML in the Biology Laboratory

Ivan Gepner

Introduction

We are all familiar with the traditional web page containing static text and images. Load the page into your browser window on Friday and you will see the same page that appeared the previous Monday. In most situations, this is satisfactory and occasional updating by the author to bring material up to date is sufficient to keep the page relevant and useful.

There may be times, however, when we want the ability to provide dynamic pages to the user, such that the page can be updated continuously. Such pages may display information that is customized to the needs of the individual user, or information that changes as the user manipulates objects on the page (usually by clicking or dragging the mouse). The technology to do this turns out to have useful applications in the biology laboratory. I would like to present a brief overview of the mechanisms used to produce dynamic pages, and a few examples of applications in the laboratory.

Theory

Dynamic pages can be created in a number of ways. One powerful technique is called dynamic html (dhtml). Dynamic HTML itself rests on the foundation of three other technologies—the document object model, cascading stylesheets, and javascript. If you will bear with me for a few minutes, here is a brief description of each of these elements. The more interesting topic of applications will soon follow.

Each page may consist of a number of elements, some of which are contained within other elements. For example, if we have a page displaying a form we can consider the page to exist within the browser window, and the form to exist within the page, and the form controls (text box, etc.) to exist within the form. From this model we can define a hierarchy of elements which can be named. Many of these elements (objects) have defined properties which also have names, and some even have actions that they can take (methods). A form button, for example, can have an action that is triggered when the button is clicked. The specification of this list of objects, properties and methods is the document object model.

Ivan Gepner is an Associate Professor in the Department of Biology at Monmouth University, West Long Branch, NJ 07764; e-mail: gepner@monmouth.edu.

Stylesheets allow us to give values to the properties of these objects by specifying a list of rules. We could, for example, make rules that specify the value of color properties for various text elements of the page or the page background. More importantly, we can specify values for positional properties of our objects. This allows, for example, paragraphs or graphics to be placed at precise positions on the page. This extends page authoring beyond the capabilities of traditional html and is a boon to the people who care about more than the content of their pages but want control over design and layout as well.

Finally, we need a way not just to specify positional information for page elements, but to change those positions dynamically. The scripting language javascript is used for this. Javascript is intermediate in complexity between the macro languages you may be familiar with in Word™ or Excel™, and the sophisticated programming languages such as C and java. We can write code in javascript that constantly assigns new values to the x and y coordinates of page elements. When that code is run, the object appears to move across the page.

Actually implementing this technique in your own pages is not a trivial undertaking. The three component techniques take some time to learn. Even then, problems appear to confound us. For example, there are differences in the document object model used by Microsoft® in the Internet Explorer™ browser (version 5) and that used by Netscape® in the Navigator™ browser (version 4). So pages that may work fine in one browse may fail to work at all in the other.

Is it necessary to devote half of your waking day to mastering these techniques and overcoming the problems? Not at all. The key to success is to have the right tool to work with. Two commercial software products currently available, Dreamweaver™ from Macromedia® and Golive™ from Adobe®, will do most of the work for you. These programs allow you to design the effect you wish to achieve visually and then write the necessary code to achieve the effect.

Applications

How can these techniques be put to good use in the classroom or laboratory? I will illustrate with one application in genetics which is finished and running, and another in developmental biology that is still in development.

Genetics

In genetics, we examine the human chromosome constitution in both normal and diseased individuals to understand how chromosome defects lead to inherited abnormalities. The traditional laboratory exercise involves passing out karyotype sheets with enlarged images of human chromosome sets. Students cut out each chromosome, organize them into a standard arrangement, and paste them to a form for submission. The ability to organize the chromosomes correctly allows a determination of the sex and disease of their unknown test cases. Of course, the mechanical nature of the exercise leads to predictable problems. Students lose chromosomes, or change their mind after they were glued to the page. This year we ran the exercise on the web by using dynamic HTML to allow images of the chromosomes to be dragged and dropped onto a computerized form. Students could easily try different arrangements before deciding on a final result, and no chromosomes were eaten by the dog.

A partially completed page might look like Figure 1. To see this exercise in action go to:

http://www.monmouth.edu/~bio/karyotypes.htm.

Student feedback after this change has been very positive. They like the convenience of being able to access the materials from any computer connected to the Internet. They especially like the ability to experiment with different arrangements without committing themselves as they must do with glue or tape.

Pitfalls

To be successful in these exercises it is very important to control which browser the students will use to view the pages, or to test the material in all browsers that are available to them. In our first implementation of the lab we neglected to do that and paid a price.

Once the chromosomes have been moved to their place on the form, the student prints the page from the browser for submission. This works fine in Internet Explorer™, but in Netscape Navigator™ the browser insisted on printing the original locations of the chromosomes rather than their current locations. As a last minute work-around, we asked them to perform a Windows™ screen save operation if they were using Navigator™, and then to print the file that was produced by the screen save.

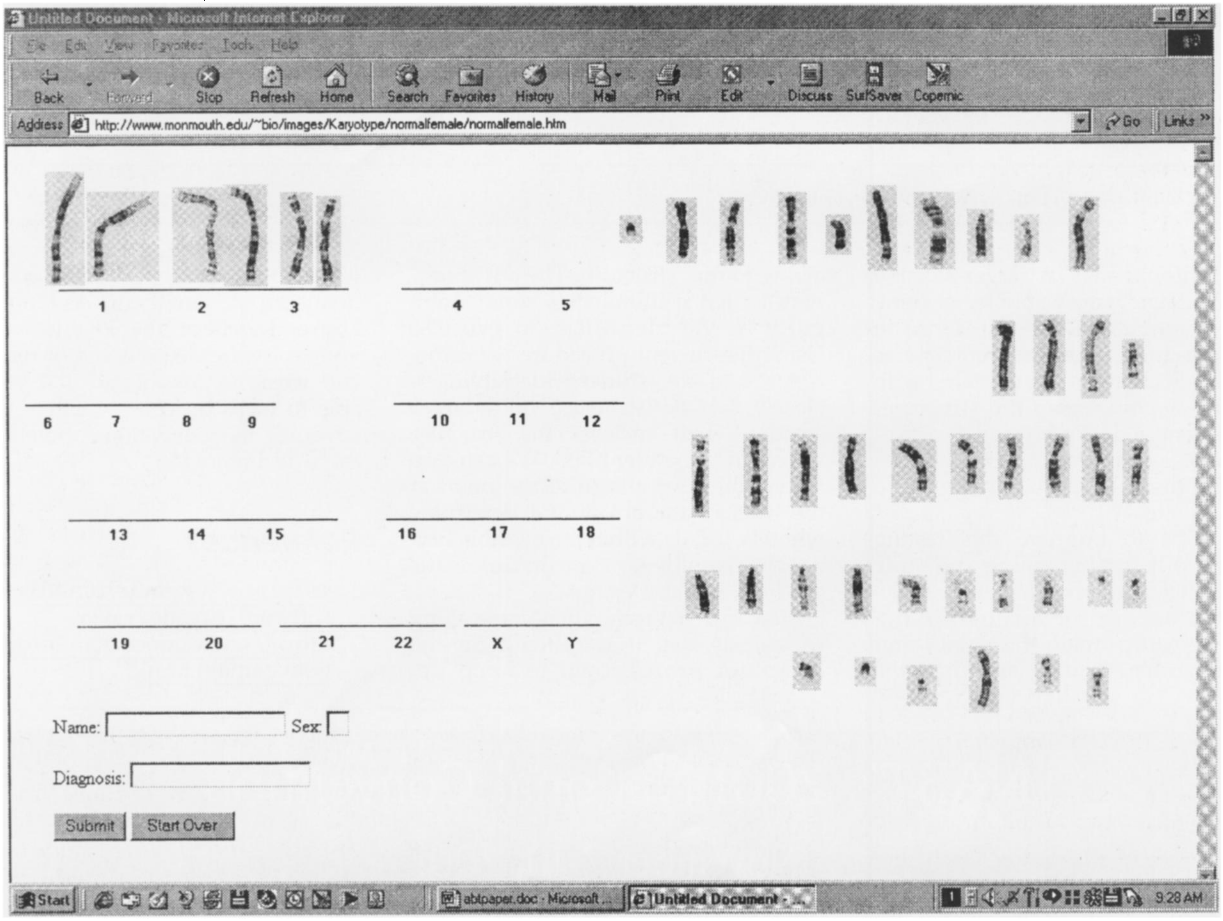
Enhancements

We hope to develop a technique that will automate the process of taking a screen capture and sending it as an e-mail attachment. This would allow the student to click on the submit button and have his/her work delivered painlessly to the instructor.

Developmental Biology

In developmental biology much time is spent looking at microscopic images of a diverse group of embryos at various stages of development. At the end of the semester students take a slide test in which they are asked to provide the names of structures in these embryos. This test used to be administered in a room full of microscopes. Students would march from scope to scope, writing down their answers at each station. Unfortunately, there were situations in which slides were moved from their original positions, leading everyone to view the wrong object. Additionally, students could not spend more time on the questions that gave them difficulty and move more quickly through the easier ones, and could not return to an earlier slide for a second look.

We now give this test on the web by displaying small samples of each



Untitled Document - Microsoft Internet Explorer

Address: http://www.monmouth.edu/~bio/images/Karyotype/normalfemale/normalfemale.htm

Name:

Sex

Diagnosis:

File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Print Edit Discuss SurfSaver Copernic
Go Links »
1 2 3
4 5
6 7 8 9
10 11 12
13 14 15
16 17 18
19 20
21 22
X Y
Submit
Start Over
Start
ablpaper.doc - Microsoft...
Untitled Document - ...
9:28 AM

Figure 1. A sample completed page.

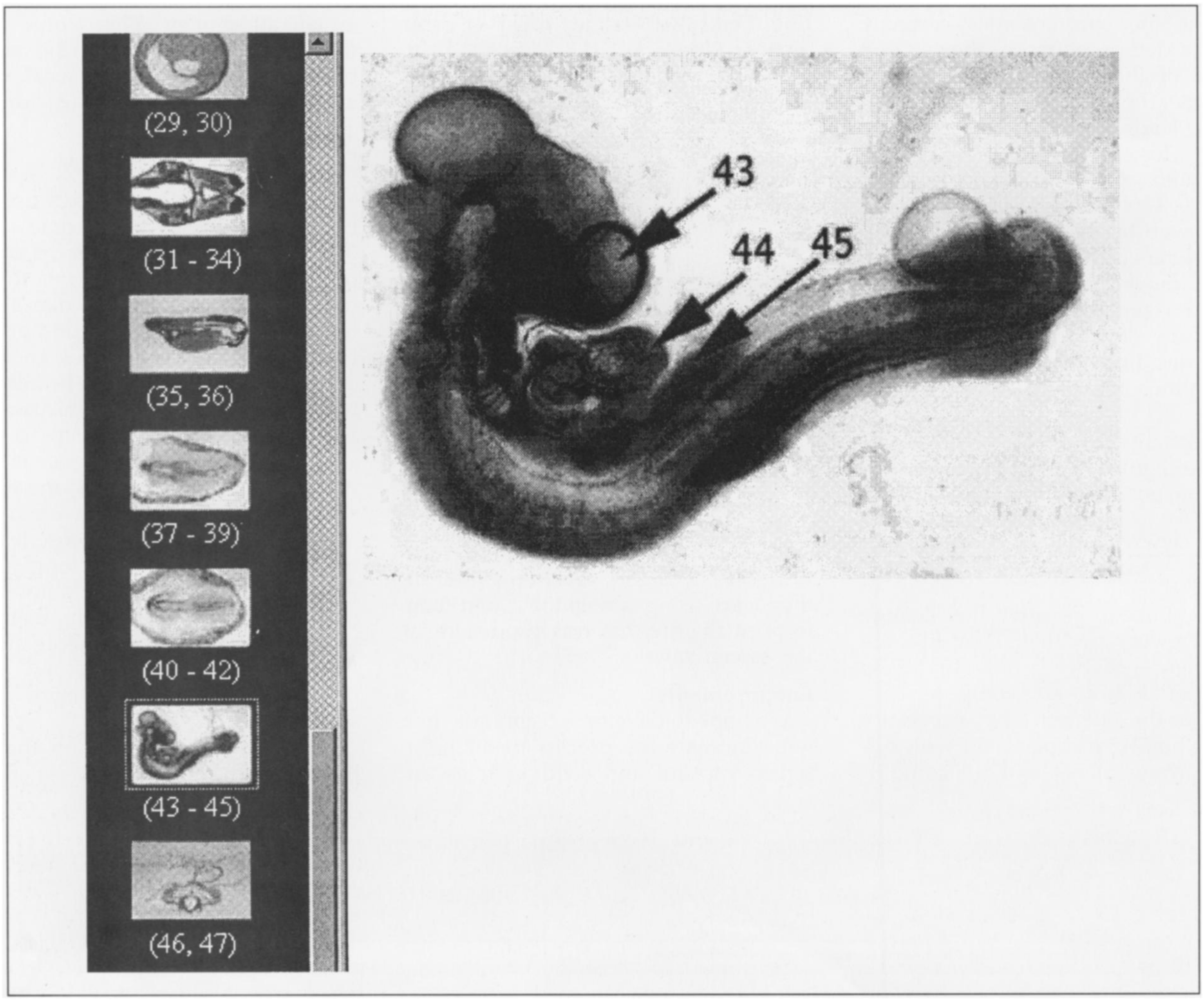


Figure 2. Web-enabled slide test in embryology.

image (thumbnails). A larger version of each image is viewable by clicking on the thumbnail. Students can go in any order and apportion their time as they like. Each image is annotated with red arrows pointing to the structures to be identified (Photoshop™ makes this easy). See Figure 2 or go to http://www2.monmouth.edu/Biology/ to view the site.

We hope to improve this testing process still further using dynamic HTML. Instead of pointing to a structure and asking for an identifying name we can provide the names and ask the student to drag and drop the arrow to that structure. The advantage here is that it illuminates some ambiguity in the identification process. Using the current procedure we sometimes find the student identifying a closely located structure or a larger structure that encloses the structure we mean to be identified. We can also record the coordinates of the point at the head of the arrow and determine whether it falls within permissible limits. This would allow for the automatic grading of these exams.

New Internet techniques come along so quickly that it is difficult for the computer professional to keep up. How then can the overworked academic profit from them? As I indicated above, I believe the key is to have your software do as much of the difficult work as possible so that you are able to focus on being creative in discovering the educational potential of these technologies.

References

1. http://www.macromedia.com/software/dreamweaver
2. http://www.adobe.com/products/golive/main.html