

CS6763 – Design Environments
Assignment 3: Design Tool Analysis Plan
Maria Cordell
Markus Haas
Andriy Naborskiyy
Oriana Williams

Introduction

This document describes goals for evaluating the functional completeness, learnability, and pleasantness of two Web design products: Macromedia Dreamweaver MX 2004 and Mozilla Composer. Both products allow users to assemble HTML pages within a visual design environment much like that of a WYSIWYG word processor or presentation program. A rendered Web page is essentially a visual representation of the structure imparted by the underlying HTML code of the page.¹ The programs' visual/design controls allow the user to specify the size, color, and appearance of objects, and is especially useful for working with text and tables as well as for general page layout. External objects (e.g., images) or links to scripted or externally generated objects (e.g., JavaScript or Flash content) can also be added in both programs.

One major difference between these products is that Dreamweaver is commercially available, whereas Composer is available free of charge as part of the Mozilla suite. As a commercial and leading product, Dreamweaver has a broad tool set that supports the entire range of the Web design and production tasks, from conceptualizing an initial design to testing and uploading the entire site to a Web server.

At a high level, the functions supported by Dreamweaver are editing, testing, and publishing. Although not required, the product works best in a site-centric mode in which an entire site's file structure and assets (images, scripts, etc.) are cataloged. The product also supports a variety of methods for approaching these tasks as well as a variety of coding environments for HTML, XHTML, XML, and PHP, and other languages.

Composer is part of the Mozilla suite, which also includes a Web browser (called Navigator), a mail program (Mozilla Mail), and other minor applications such as a Java debugger. Although basic by commercial standards, Composer is a very capable HTML/Web page editor. Composer offers most of the functions users need to create and edit pages, including image placement and sizing, and table setup and modification. It can also import images and Flash content, though it cannot natively interpret or display the latter.

As a freely available application, Composer is a very usable tool for creating and editing HTML pages. It does lack the dedicated site management features of commercial products, but it contains the functionality needed for many Web page editing tasks.

The remainder of this document describes the assessment we will carry out to test and eventually compare Dreamweaver and Composer for

¹ A plain text HTML editor is also available in both products. This editing mode is useful in more complex coding situations where the user might need to make adjustments difficult to make in the visual/design mode. Our evaluation tests the products through the creation of a relatively simple design for which the code editing mode will not be necessary.

functional and design support effectiveness in the process of creating a simple Web page. The following pages describe our design tool quality assessment goals, test subject criteria, test metrics, design objectives, data collection methods, and subject questionnaires.

Design Tool Quality Assessment Goals

The eventual goal of this assessment will be to compare a group of users' experiences and accomplishments in Dreamweaver versus Mozilla Composer. To achieve this, each user's interaction with the design tools will be evaluated at a higher surface level, rather than a granular expansive level. To clarify, the effectiveness will not be measured since it focuses intensively on time needed to achieve subtasks. Likewise, completeness will not be a metric of the evaluation since intends to measure the success of a tool in several scenarios within multiple contexts. Instead, the evaluation is intended to concentrate on the basic functionality, learnability, and the user pleasantness (or frustration) in relation of the two webpage design tools.

- **Functional completeness** of a tool reveals how easily a task can be completed in simple contexts. It is not intended to be a complete coverage metric, but rather, a shallow, yet important assessment of the tool's basic capabilities. It also ensures that simple tasks may be done without confronting system bugs or other incorrect behaviors.
- **Learnability** is the degree of effort required to complete a task. Learnability assumes that the learner has some level of familiarity with the design domain, although no familiarity of the design environment is necessary. In the evaluation process, the user will be unfamiliar with both of the software design tools, so learnability can be more fairly measured.
- **Pleasantness** is another metric of a tool's success or quality. If users feel their experience with the design tool is pleasant, or free from much frustration, they may be more likely to use it. Although system power, learnability, and efficiency help determine a tool's success, a user's view of the type of experience the tool might create for them is important. It might become a factor in the tool's popularity.

Test Subject Criteria

Our test plan calls for evaluation a variety of common HTML editing tasks. We want to evaluate how effectively the tools support common tasks with which users should be familiar with in HTML, but which they do not necessarily already know how to execute with the tools. Our ideal test subject meets the following criteria:

- **Basic knowledge of HTML page composition** – A skilled novice, or someone who knows how to manually compose a basic HTML page, is the best subject. This is because we want to test the tool's facilities for building pages, not the user's knowledge of HTML. Hence, it is best if the user is already familiar with all of the web page terminology, so their focus may be on using the design tool.
- **Minimal prior exposure to Dreamweaver and Composer** – The less the users know about the tools, the better we are able to assess the tools' learnability and overall design quality. We may also include one more complex task, like adding a JavaScript rollover to a screen element, as a last task. The user will likely not know how to do this without the tool; in this case, we want to see how well the tool supports the user in adding functionality that the user doesn't already know how to code.

Our test subjects will be fellow Georgia Tech students. We plan to solicit participants from the graduate program in Information Design & Technology, IDT, that have experience in the knowledge domain of Web design but little experience with the design tools being evaluated. The reasoning behind recruiting IDT students is that they come from a designer's perspective. This type of participant background will hopefully encourage them to use the design view of the tools (rather than the coding view) which is the main focus of our evaluation.

Test Metrics

Each person that goes through our evaluation process will be expected to compose a sample page with both Dreamweaver and Mozilla Composer. To minimize and counterbalance possible ordering effects, we will switch the tool use order for half of the test subjects. For this reason, it is important that we evaluate an even number of people given our small scale evaluation. Hence, we will test six to eight people on the set of tools. Also, we will enforce a twenty- five minute time limit after which we will stop the user from continuing with their task. By imposing a time limit, the tools' functional and learnability differences will become more apparent. The participants will be informed of the time limit and, of course at any time, they may choose to discontinue the experiment.

The participant will be asked to add a third Web page to an existing two- page site design. Since our experiment is set up to allow for a certain dependency on these existing Web pages, it is important that the user is given a comparable initial view of these pages in both Dreamweaver and Mozilla Composer. The subjects will be told the fully qualified directory path and name of the existing web pages of the site. They will first be asked to first open these files prior to constructing the additional web page. This way the participants will gain exposure to the other portions of the site and be given the opportunity to take advantage of the implemented site. We are interested to see if they choose to utilize this capability of the system or not.

As mentioned, after evaluating both Dreamweaver and Mozilla Composer, we hope to compare the functional completeness, learnability, and user pleasantness of the respective design tools. To accomplish this, some specific techniques will be invoked during evaluation.

- **Evaluating Functional Completeness** – Since both Dreamweaver and Mozilla Composer are products available on the market, our evaluation will not be concerned with analyzing possible system bugs. Rather, the functional completeness portion of the evaluation will focus mainly upon the ability of a user to complete a simple task in a simple context. Since the user is already familiar with the knowledge domain through their previous HTML experience, it is reasonable to believe that they will be able to understand the task expectations of the unfamiliar design tools. Functional completeness will be determined by the amount of tasks successfully completed to specification within the given time range. The amount of task partially or nearly completed will also be taken into consideration.
- **Evaluating Learnability** – The learnability evaluation will graze upon the users' initial learning curve, but will not be complete since it does not span a series of user experiences with the design tools. First, to determine the learnability of the software tools, the overall time of the user takes to complete the task will be measured. Also, the familiarity of users with certain menus or screens over the testing time will be monitored. The speed and comfort at which users revisit these

previously explored options will subjectively be evaluated. Learnability will also be measured by looking closely at the subjects' error rate. This is because if a high number of errors are committed during the task, it implies that the learnability curve of the design tool might be too steep.

- **Evaluating Tool Pleasantness** – The user's experience with both Dreamweaver and Mozilla Composer will be subjectively measured through a Likert scale survey. This scale will be able to provide quantitative results regarding the user's experience.

The details regarding the methodology of the test's data collection for functional completeness, learnability, and pleasantness will be further detailed in the "Data Collection" section.

Design Objective

Our testing protocol focuses not on specific low-level tasks, but on the larger goal of building a new page for an existing Web site. Like the case of the VLSI circuit layout task profiled in Chapter 10 of *The Psychology of Human-Computer Interaction* [4], our evaluation lets the Web page designer generate tasks as he or she proceeds. Working from an already designed page, it will be our subjects' job to decompose design elements into the design tool's functional elements for building the page.

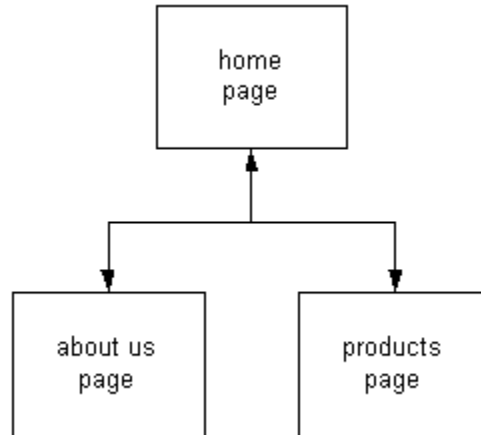


Figure 1: The basic site hierarchy within which the design tool evaluation is done. The “home” and “about us” pages already exist; the test subject is asked to create the “products” page.

Each subject will be supplied with a hardcopy of the page design and a set of specifications for the site. Figure 1 shows the completed page design towards which each subject will work. Many of the HTML elements required to build the page will be apparent from looking at the completed design; other elements, such as the page background color, will be provided in the printed specifications. Appendix A has a complete list of instructions and specifications that are not apparent from viewing the webpage visual as seen in Figure 2. In general, the page specifications include:

- the page background color, in its hexadecimal value
- the page title
- the maximum page content width
- image border specifications

Since this is a new page being added to an existing site, the test subject will have other site pages available for review. Part of our evaluation will determine whether the subject is able to identify these other pages to examine their content within the design tool. The subject will need to at least load images from the existing site in order to create the new page.

THE SNOW BOARD

AWESOME GEAR. AWESOME PERFORMANCE. AWESOME FUN.

Our Products | [About Us](#) | [Home](#)

Our Products

All our products are made of the finest Peruvian fiberglass and painted with the best Swedish lacquer.

- Unique designs sure to please novices and experts alike
- All designs available in three popular sizes
- All designs include nifty carrying bags
- Custom designs done for a special price!

Current Snow Board Products		
	The Challenger -- Our best board, for the most challenging runs.	\$800
	The Widowmaker -- An excellent board for pulling crazy stunts.	\$749
	The Wimp -- Our starter board. Get one if you dare.	\$499

Copyright © 2004, The Snow Board, LCC.

Figure 2: The layout and content design of the products page. This is the page users will build in both Composer and Dreamweaver.

The process of implementing the page will include at least the following tasks in the selected tool:

- launching a new page in the tool
- adding a page title
- changing the default background color
- adding text
- creating the layout within provided page specifications
- creating links
- creating a multi-column table with a border and a merged-cell header row
- adding images to the page and to the appropriate table cells
- changing text attributes in the appropriate places (bold text and/or “heading” selections)

There are multiple approaches to implementing a design objective in HTML. We do not expect all users to use the same approach to implementing our design in HTML. In fact, there will be some variation, not just between

the tools under evaluation, but within each tool. The important aspect to capture is the specific tool functions the user chooses to employ to carry out each objective and how well the selected approach served the user's needs. The results will directly inform the tool's evaluation in terms of effectiveness and learnability.

Data Collection

The model for collecting and analyzing data was taken from sources on design protocol analysis [1][2], which give examples of analyzing designing activities. Our objective differs, however, in that we aim to evaluate and compare two tools on the aspects identified above. We also give our subjects a concrete tasks with a limited (but varied) set of steps required for completion.

In the test metrics chapter we already discussed how we invite the subject to work with the tool. Each subject works alone at a computer while the researcher monitors the session passively from a distance. Camtasia studio is used to help capture user interaction and record a video of the screen. This setup allows registering several aspects of the subject's experience needed for our analysis. While passively monitoring the subject we look for obvious sings of anger or frustration.

To register the user interaction with the design tool, we take a video of the screen with Camtasia. The Camtasia studio, from TechSmith, is a popular screen capture utility. The most important feature of this tool is its screen capture codec, which is optimized for desktop recording [5]. Other researches [4] used Camtasia in combination with a remote control application to monitor the screen from another computer and videotaped the keyboard and mouse interactions. Since in this experiment, the researcher will be present in the same room, there is no need for the remote control portion of the application. Because of the low level of subjects' expertise with the development tool, we assume that the use of keyboard shortcuts will be limited. Otherwise, keyboard shortcuts and right- mouse menu shortcuts use can be deduced from the screen video.

After the session, we will play back the Camtasia video and analyze it in search of what operators were accessed, the method used to access them, and what objects were added or removed from the scene. We will note each time the subject creates an object. If an object is removed, or if the "undo" program function is used, this will signify an error in selecting an operation, and we will make a note of it. We will also note the way each user browses for operations; each time a subject opens a submenu or dialog box, browses the choices for prolonged time (greater than 5 sec) and closes submenu/dialog- box, this signifies idle browsing. We collect all this data in table similar to the following:

action	Comment
Open menu X	
Selection made	
Open dialog Y	
Close dialog	Subject couldn't find operation; time is lost
Picture added and removed immediately	Error is selecting operator; time spent on selection is lost
Subject was asked to stop	This shouldn't happen with any of our subject, should it?

Analysis of the data will consist of several categories. The solution will be measured for completeness to weigh the corresponding performance time parameter. The protocol is searched to identify user trial- and- error attempts, as well as to identify and time- measure idle browsing.

The task given to the subject is simple and well defined. A designer with working knowledge of HTML should have a clear idea of what components constitute the solution and proceed on executing it without extended pauses. We exclude the possibility that idle browsing is due to task acquisition. So we look for the idle browsing as indicative of learning the tool.

In addition to idle browsing, we are interested in finding how many selection errors the user makes. To find those errors we feel that removing an object or undoing an operation is a good indicator. The same rationale is applied here as to the idle browsing. The task is simple enough that for a user experienced in the domain of web design that the “undo” operation cannot be used to address the exploration of the solution domain, but rather, only can be used for the exploration of the set of operators available in the tool. We compare learnability of the tools based on the number of operator selection errors and the total time spent on idle browsing for operators.

To measure completeness, the solution will be compared with the provided template page according to the list of tasks specified in the Design Objectives. Also, each constraint specified in Appendix A will be checked to contribute to the completeness grade.

Test Subject Questionnaires

Pre-Experiment User Questionnaire

1. What is your age?
2. What is your major?
3. How familiar are you with Dreamweaver? (Likert 7- very familiar/not familiar at all)
4. How familiar are you with Mozilla Composer? (Likert 7- very familiar/not familiar at all)
5. How much experience do you have with creating a Web page? (Likert 7- no experience/a great deal of experience)
6. How much experience do you have with creating Weblog pages? (Likert 7- no experience/a great deal of experience)
7. How comfortable are you in writing HTML code (with no graphical editor or tools)? (Likert 7- very comfortable/not comfortable at all)

The rationale behind a pre-questionnaire is to ask questions whose answers might be affected by the experience of doing the experiment's tasks. For instance, if the subject has a difficult time with the experimental task, he may discount his level of expertise at that type of task. An objective measurement of a subject's skill is valuable because it lets the researcher evaluate the difficulty of the experimental task. Questions 4- 8 below are of this type. Other questions are not sensitive to being asked before or after the experiment. These include standard questions characterizing the demographics of the subject group (Questions 1, 2, and 3). They are usually asked at the beginning to allow the subject to concentrate fully on the task he has just completed through questionnaires administered during and after the experiment. We observe this convention in our experiment.

Our user group is small and expected to be relatively uniform (college students recruited from one of the LCC programs), so the need to characterize the subject population is diminished, hence only the three questions. Questions 4 and 5 are designed to characterize the subject's experience with each design tool. Since we are testing learnability, it is essential to know how much of a background the person has. Questions 6 and 7 are designed to assess the subject's general knowledge of composing Web pages with any product. Weblogs fall into this category because they do produce a Web site, although the HTML code is almost completely abstracted from this user, a situation similar to using Dreamweaver or Composer without taking advantage of the advanced features or writing native HTML code. (A weblog, or blog, is an online diary maintained in most cases by an individual.) The final question assesses a different aspect of the subject's experience with HTML-- writing it directly in a text editor in a similar fashion to a traditional programming language. It is important to measure this as well because a subject who has a lot of experience in this style of HTML composition may entirely bypass the graphical tools that Composer and Dreamweaver provide; in this case, the results would be somewhat difficult to assess because we would not know whether the user switched to directly editing the HTML because of frustration with the tool's usability,

lack of functional completeness, or experience and preference for that style of editing.

The scoring for most of these questions (and most in all the questions overall) is a Likert scale. Ease of scoring is the main reason for writing the questions in this manner. There is no need for the subjects to detail a great amount of information in these questions. The final questionnaire gives opportunities for users to reveal a greater depth of information about their experiences during the experiment. Those questions are harder to score but may reveal elements of the data that might otherwise not be reported.

Intermediate User Questionnaire

1. Were you able to complete the task given? (Yes/No)
2. Did (Dreamweaver/Composer) provide all of the functionality you needed?
3. How difficult did you find the task? (Likert 7- (very difficult/not difficult at all))
4. How enjoyable did you find this task? (Likert 7- enjoyable/not enjoyable)
5. Do you have any comments about the program you just used?

The subject is given a brief test between the two parts of the experiment for two reasons. First, the experience of using the first program is still very fresh in his memory; one can easily forget or confuse details when the exact same task is repeated twice in a row. Second, once the subject has used both tools, he can compare them and subjectively answer questions about one tool in a comparative manner with the other. For example, if the subject finds the first tool moderately easy to use, but finds the second one extremely easy to use, after completing both parts, the person is likely to assign a lower ease of use score simply based on the comparative ease of use of the second tool.

The questions are identical for both tools. The first two questions are designed to assess the functional completeness of the tool. The questions are scored yes/no because they are measuring an absolute value: whether the user able to complete all of the individual subtasks in the given project. The next question tests the learnability and usability of the tool. The task given is sufficiently simple that all of the subtasks do not require extensive knowledge of page composition, or even of the advanced parameters that tailor the behavior of the HTML editor in use.

Post- Experiment User Questionnaire

1. Which tool allowed you to do the task more quickly?
2. Which tool allowed you to do the task more easily?
3. If you had to add more complicated features to the web site, which tool would you choose?
4. Was the task clearly worded? (Likert 7- very clear/not clear at all)
5. Did the illustration of the page you were to create provide all of the information necessary to create it? (Likert 7- strongly agree/strongly disagree)
6. In your experience with making web pages, have you done tasks similar to the one you just did? (Likert 7- strongly agree/strongly disagree)

In this questionnaire, the questions are not all straightforward in what they are measuring. The first question is designed to test the user's perception of which one took longer, not to make an absolute measurement. This is operating on the principle that a user will not perceive the passage of time as much if he is making progress in the task, but time spent hunting for a solution will register in the person's memory. In effect, we are making use of the "time flies when you're having fun" principle. The result provides a bit of insight into which program was more fun to use, and to a lesser degree, which one is more usable.

In the next question, we see which tool the subject perceived to be easier. This question is a much more direct way of getting similar data. By comparing the results of these two questions, we will be able to better understand the user's level of confidence that one tool is easier to use than the other. The third question builds on the first two by asking the subject in a very roundabout way which tool he liked better and found easier to use. The subject is most likely going to choose the more learnable tool, which likely will also be the one that is more enjoyable to use.

Questions 4 and 5 enable the subject to explain away difficulties in understanding the tasks. It is important that the researchers know of such difficulties so the subject's results can be discarded if the outcome is tainted by lack of understanding of the tasks are to be accomplished.

The final question is not entirely necessary, but may reveal some information. If the subject has done something almost like what he did in this experiment, the results may be somewhat affected because the person is no longer a novice at the overall task. If the person used one of the programs being evaluated here, or one with similar layout, he will likely show a lower completion time or less hunting for tools in the interface during while he is working. Knowing a cause for a shorter time score can be a justification for the data to be disregarded in the statistical analysis. However, some documentation of why that is being done is necessary, and can be provided by this question.

References

1. Eastman, C (1970) On the analysis of intuitive design processes, G. Moore, ed. Emerging Techniques in Environmental Design and Planning, MIT Press.
2. Suwa M, and Tversky B, (1997), What do architects and students perceive in their design sketches? Design Studies
3. Juarez, O.; Hendrickson, C.; Garrett, J.H., Jr.(2000) Evaluating visualizations based on the performed task, Information Visualization, 2000. Proceedings. IEEE International Conference
4. Card, Stuart K., Thomas P. Moran, and Allen Newell, The Psychology of Human- Computer Interaction. Lawrence Erlbaum Associates, 1983.
5. A. Seffah, R.Naghshin, R.Kline, "Using walkthroughs to profile software developer personas"
http://hci.cs.concordia.ca/www/hcse/projects/humancase/Using_Walkthroughs_To_Profile.pdf
6. Camtasia Studio 1.1 review. http://www.designer-info.com/master.htm?http://www.designer-info.com/Photo/camtasia_studio.htm

Appendix A: Instructions

Each test subject will be provided with a brief set of instructions designed to set the context and define the object for the session, as follows. Recall that only the specifications not apparent through the Figure 2 visual will be detailed in the in the requirement list.

Instructions

- You are to add a page to the Web site for The Snow Board, a manufacturer of snowboards and snowboarding gear. The page you're adding is the company's Products page.
- All site files are located within the "My Documents" folder, in a subfolder (or working folder) named "SnowBoard."
- The existing site pages in the SnowBoard folder are the company's home and "about us" pages: **index.htm** and **aboutus.htm**.
- The attached printout shows the completed Product page design. Your task is to translate this design into a Web page in two separate passes: once in Macromedia Dreamweaver and once in Mozilla's Composer. The file name for the new page you will create each time must be **products.htm**.
- Keep in mind the following page specifications as you create the page. (Note that the following does not fully describe the page. Refer to the attached design hardcopy for other requirements.)
 - The page background color is CCCCCC.
 - The page title is "The Snow Board -- Products" (without the quotes).
 - All page content should have a maximum width of 800 pixels.
 - Product images should have a 1-pixel black border.
- The images you need for the Products page are in the site's working folder.