DYNAMIC VERSUS STATIC MENUS: AN EXPLORATORY COMPARISON

JEFFREY MITCHELL

BEN SHNEIDERMAN

Sixty-three subjects completed 24 tasks using a menu driven computer program. The menu items appeared in a fixed (static) order during 12 of the tasks. During the other 12 tasks the menu item order changed dynamically such that the most frequently selected items always appeared at the top of the menu. All the subjects tried both dynamic and static menus.

The subjects that used adaptive dynamic menus for the first set of tasks were significantly slower than those who used static menus on the first set of tasks. Subjects' performance during the second set of tasks was not affected by menu style. Eighty-one percent of the subjects preferred working with static menus to working with dynamic menus.

INTRODUCTION

Norcio and Stanley [1988] indicate that "An ideal computer system should adapt to the current user by compensating for weaknesses, by providing help appropriate to the context, and by decreasing the mental and physical workload of the particular user."

One means of decreasing a user's mental and physical workload is to automatically position the cursor at the next necessary screen location. In a menu driven system, for example, the next needed menu item could be highlighted instead of always highlighting the first item on a menu.

A system with a user interface which can adjust to meet a user's needs is called an adaptive system. "The idea of an adaptive interface is straightforward; simply, it means that the interface should adapt to the user rather than the user adapt to the system" [Norcio and Stanley 1988].

Adaptive interfaces fall into several categories, including those which support macro languages, user selectable interfaces, user definable interfaces, integrated user interfaces, and dynamic or self-adapting user interfaces. User interfaces can be classified on several dimensions including the amount of user intervention involved and the skill level required for user interface modification.

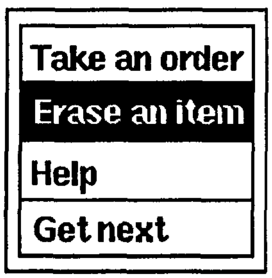
Some software applications allow users to setup, program, define, or select interaction styles and/or command sets. These user interfaces are called programmable adaptive interfaces. Sprint and EMACS are examples of applications which support this kind of user interface. The Sprint wordprocessor, allows the user to select from a menu of common command sets (WordStar, Microsoft Word, WordPerfect, etc.). The EMACS editor allows a user to define commands using a Lisp-like programming language.

Dynamic adaptive user interfaces automatically change in response to the user's behavior. Dynamic modification can be automatic and continuous, or periodic and at the users discretion.

This paper presents a comparative experimental analysis of a menu driven user interface operating in either a static or a dynamic adaptive mode. In the static mode menu item order is fixed. In the dynamic mode menu item order changes continuously such that the menu items most frequently selected by the current user appear at the top of the menu.

The menus consist of a vertical list of items located in a rectangular box in the upper left corner of the screen (see figure 1). Initially the first item on the menu is highlighted. To select an item, the user presses the cursor keys until the desired item is highlighted, the user then presses the enter key to make the selection.

Figure 1: Sample Menu with second item highlighted.



In the dynamic mode the menu items are continuously reordered so that the most frequently selected item is always the first item on the menu. The first item is followed by the remaining items in order of frequency.

ADVANTAGES & DISADVANTAGES OF ADAPTIVE USER INTERFACES

The trade-off between compelling a user to work within a static user interface or allowing the user to work with and perhaps modify a dynamic user interface is a significant design issue both in terms of cost and user satisfaction.

ARGUMENTS FOR STATIC USER INTERFACES

1. the user will become familiar with a fixed interaction style.

2. a static interface makes users skills portable across systems, assuming that the user has mastered the fixed interaction style.

3. a standardized static user interface for an architecture can reduce learning time if new applications adhere to the conventions of the existing interface.

4. implementing a dynamic user interface is more costly than a static user interface.

5. users do not want to take the time to configure their own interface.

6. multiple products sharing a single user interface will have the same "look and feel."

7. "the user may not be able to develop a coherent model of the system if the system changes frequently" [Norcio and Stanley 1988].

8. The users may experience a "loss of control or the feeling of loss of control" when using a dynamic adaptive user interface [Norcio and Stanley 1988].

ARGUMENTS FOR ADAPTIVE USER INTERFACES

1. the user is probably an expert in the task domain and hence may know more about what the interface should look like than the system designer.

2. the user probably knows best how he or she wants to use the application and does not want to use an arbitrary set of commands.

3. one interface cannot possibly meet every user's needs.

4. users frustrated by a user interface which does not meet their requirements will abandon the application in favor of another which has a user interface which more closely meets their needs.

5. a reduction in learning time will result if a user interface can be configured to work like an interface the user has already mastered.

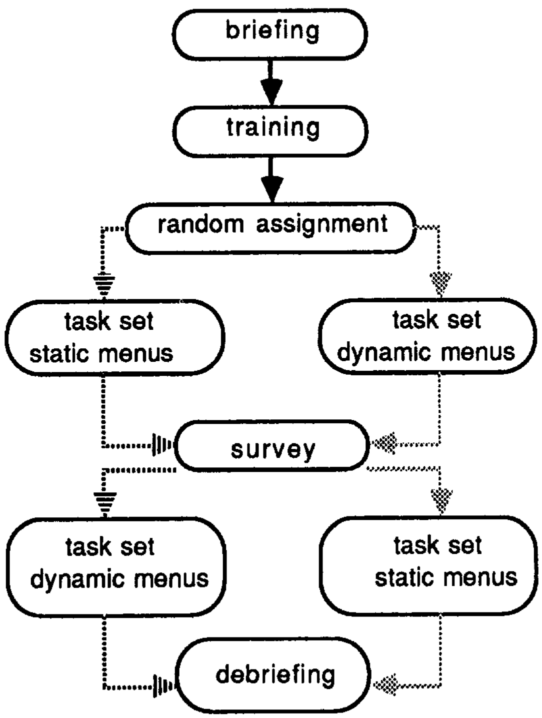
6. users will be more satisfied using a system which they can interact with in their own style.

7. an adaptive interface can increase performance by reducing a users mental and physical workload.

THE EXPERIMENT

This experiment is a two group design with repeated measures. Subjects are randomly assigned to groups. Subjects in one group perform 12 tasks using dynamic menus and then repeat the same 12 tasks using static menus. Subjects in the other group perform the 12 tasks using static menus and then repeat the tasks using dynamic menus (see figure 2).

Figure 2: Flowchart of Experimental Procedure



The experiment measures the impact of two independent variables on four dependent measures. The independent variables are menu style and presentation order. The dependent variables are elapsed time to complete tasks, number of errors, number of operations, and subjective satisfaction.

Menu style is a nominal level variable with two values: dynamic item ordering and static item ordering. Presentation order is also a nominal level variable with two values: dynamic menus first and static menus first.

The experiment assesses the effect of dynamic menus on the conceptual variable: user productivity. User productivity is measured using three operational indicators. They are: (1) elapsed time to complete each task, (2) number of errors made and corrected during each task (the system requires the user to continue working on a task until the task is completed successfully); (3) number of operations logged to complete each task. An operation is a menu item selection (a press of the enter key).

Information about subject's prior computer experience was collected during the experiment, and affective comments were solicited under program control at the end of the experiment (subjective satisfaction).

SUBJECTS

Seventy-three college students, recruited from the psychology department subject pool at a prominent university in the north eastern United States, participated in the experiment. Data for ten of the subjects was discarded because it was incomplete, or because of program or equipment failure.

Seventy-six percent of the subjects were freshman or sophomores with little or no computer experience. Only one of the subjects claimed to know how to write computer programs. On average, the subjects were familiar with less than eight of the 21 QUIS devices [Chin et. al. 1988]. The subjects were from diverse majors, primarily in the liberal arts and social sciences. Psychology was the most common major (over 20% of the subjects were psychology majors).

MATERIALS AND APPARATUS

The experiment was conducted in two small rooms equipped with four or five IBM-PS/2 Model 50 microcomputers. The experiment including briefing and debriefing was conducted automatically under the control of a computer program written specifically for this experiment.

Data for subjects was collected by the program and stored in files on the hard disk drives of each micro computer. Following the experiment the data was retrieved from the nine machines and stored on floppy disk for analysis.

The only additional materials required for the experiment were the informed consent form, which subjects signed and credit slips which were given to the subjects. Subject participation was voluntary, compensation was course work credit.

PROCEDURE

Subjects enter the laboratory and are seated in front of a micro computer. They sign a standard Psychology Department consent form and then proceed with the experiment under the guidance of the computer program.

The program has seven phases: briefing, training, task set 1, survey, task set 2, debriefing, and comments. (See figure 2)

The first phase (briefing) includes instructions to the subject and a description of the general type of tasks that the subject will complete during the experiment.

The second phase (training) allows the subject to become proficient at selecting menu items using a moving bar menu. As soon as the subject feels confident they terminate their own training and proceed to task set 1.

The third and fifth phases (task sets) include 12 menu selection tasks each. The same tasks are presented to the subjects twice, and subjects make selections once using dynamic menus and once with static menus. The program randomly assigns menu styles to the subjects.

A survey, based on the QUIS is administered between the two tasks (phase 4). Following the experiment an online debriefing is presented (phase 6) followed by a screen where subjects type in comments concerning their feelings about the menu styles.

Prior to the experiment a pilot study was conducted to evaluate the usability of the experimental procedure and apparatus and to collect preliminary data.

RESULTS

Subjects in the dynamic menus first condition required significantly more time to complete the first 12 tasks than did subjects in the static menus first conditions (F = 9.01, Prob. <= 0.01). Subjects in the dynamic menus first condition needed about three minutes more to complete the first 12 tasks than subjects in the static menus first condition (see table 1)

Table 1: Average total time required to complete 12 tasks broken down by presentation order and menu interaction style. (Time is in minutes; "D" stands for Dynamic; "S" stands for Static; standard deviation in parentheses; D->S means that the first task set was completed with dynamic menus, and the second task set was completed using static menus).

Mean (Std. Dev.)	Dynamic	Static
D->S n=35	19.67 (4.51)	11.07 (2.37)
S->D n=28	11.14 (1.87)	16.49 (3.75)

There was no significant performance time difference between menu ordering styles on the second 12 tasks (F = 0.02). Subjects in both groups were able to increase their performance significantly, and needed only about eleven minutes to complete the 12 tasks the second time around.

Menu style (dynamic or static) did not have any significant effect on the average number of operations required to complete the 12 tasks (F = 0.10). The first task set required about 260 operations on average. The second time the tasks were completed the subjects became more efficient requiring about 35 fewer operations to complete the same 12 tasks (see table 2).

Table 2: Average number of operations (menu selections) required to complete 12 tasks broken down by presentation order and menu interaction style. ("D" stands for Dynamic; "S" stands for Static; number of subjects per cell is given in parentheses).

Mean (Std Dev)	Dynamic	Static
D->S n=35	260 (27.46)	224 (18.14)
S->D n =28	223 (32.21)	262 (14.34)

Menu style did not significantly (F = 0.25) influence the average number of errors made by subjects while completing tasks (see table 3). Subjects did make fewer errors on average during the second task set independent of the menu style.

Table 3: Average number of errors made while completing 12 tasks broken down by presentation order and menu interaction style. ("D" stands for Dynamic; "S" stands for Static; standard deviation in parentheses).

Mean (Std Dev)	Dynamic	Static
D->S n=35	9.94 (7.55)	4.89 (4.28)
S->D n=28	4.82 (3.86)	11.07 (10.41)

In summary, menu ordering style did not significantly impact the average number of operations required to complete the 12 tasks. Nor, were subjects more or less likely to make errors using one of the menu ordering styles.

Subjects on average, made about ten errors and required about 260 operations to complete the first 12 tasks independent of the menu ordering style. They were able to reduce the number of errors to about five, on average, for the second set of 12 tasks, which required about 225 operations. Again, no significant difference existed between groups for errors or number of operations.

During the debriefing phase of the experiment subjects were asked if they preferred using the dynamic adaptive menus or the static menus. 81% of the subjects preferred static menus over dynamic (this finding was not influenced by presentation order).

At the end of the debriefing subjects were asked to make further comments (free form text response). About half of the subjects made comments but only 13 of the subjects directly addressed the menu style issue. Of those, ten preferred static menus and three preferred dynamic menus.

DISCUSSION

In spite of the fact that the dynamic menus in this experiment were organized to assist the menu user by moving frequently selected items to the top of the menus, subjects still paid an initial time penalty learning to use the dynamic menus.

Casual observation during the experiment indicates that subjects almost instantly memorize the relative position of menu items even on menus with 15 items. Some subjects were disoriented and experienced strong negative affect when the menu ordering changed, lending credence to Norcio and Stanley's assertion that users may be disoriented and have feelings of loss of control using an adaptive system.

The findings of this study suggest at least three things. First, dynamic menus, as described here, can slow down first time users, at least until they become accustomed to this interaction style. Second, after even a small amount of practice there is no performance difference between dynamic and static menus. Third, most of the subjects did not like dynamic menus and the affective component was strongly felt.

Many of the problems associated with dynamic adaptive user interfaces may be reduced by giving the user control over the dynamic aspects of the interface (i.e. the user should be able to *turn on*, *turn off*, and adjust the granularity (frequency of adaptation) of the dynamic part of the interface. Also, the interface should *suggest* changes when changes are indicated rather than adapting the user interface automatically).

This study demonstrates that users of a dynamic adaptive menu-driven user interface only achieve performance characteristics similar to users of an equivalent static menu-driven user interface with training. Furthermore our subjects expressed a preference for a static user interface.

Further research is necessary to determine the effects of a dynamic adaptive user interface on expert or experienced users. Those who believe dynamic adaptive menus are beneficial must try other strategies to promote their case.

REFERENCES

Chin, J. P. "A Dynamic User Adaptable Menu System: Linking it all together". *Tech. Report CS-TR-2120* Department of Computer Science, University of Maryland at College Park (October 1988).

Chin, John P., Virginia A. Diehl, and Kent L. Norman. "Development of an Instrument Measuring User Satisfaction of the Human-Computer Interface." *ACM SIGHI Proceedings* (May 1988) pp. 213-218.

Norcio, A. F. and Stanley, J. "Adaptive Human-Computer Interfaces". *NRL Report 9148* Naval Research Laboratory (Sept. 30, 1988).

Shneiderman, B. *Designing the User Interface Strategies for Human-Computer Interaction.* Addison-Wesley Publishing Co., Inc., Reading, MA, 1987.