User Learning and Performance with Marking Menus

Gordon Kurtenbach and William Buxton
Dept. of Computer Science
University of Toronto
Toronto, Ontario Canada, M5S 1A1
Phone: (416) 978-6619, Email: gordo@dgp.toronto.edu, willy@dgp.toronto.edu

ABSTRACT
A marking menu is designed to allow a user to perform a menu selection by either popping-up a radial (or pie) menu, or by making a straight mark in the direction of the desired menu item without popping-up the menu. Previous evaluations in laboratory settings have shown the potential of marking menus. This paper reports on a case study of user behavior with marking menus in a real work situation. The study demonstrates the following: First, marking menus are used as designed. When users become expert with the menus, marks are used extensively. However, the transition to using marks is not one way. Expert users still switch back to menus to refresh their memory of menu layout. Second, marking is an extremely efficient interaction technique. Using a mark on average was 3.5 times faster than selection using the menu. Finally, design principles can be followed that make menu item/mark associations easier to learn, and interaction efficient.

KEYWORDS: Marking menus, pie menus, gestures, pen based input, accelerators, input devices, multimedia

INTRODUCTION
Menus are used extensively in human computer interfaces. They provide critical information on what commands are available and a way to invoke commands. Some menus require substantial computing before display and this delays the user. Also, menus appearing and disappearing on the screen can be visually disruptive—a menu may obscure objects on the screen that are the focus of attention.

Some systems do provide methods to by-pass menus but the by-pass mechanism requires an action that is radically different than selecting using the menu. For example, in some systems, a user selects from a menu using the mouse but by-passes the menu using an "accelerator key" on the keyboard. The problem is that one has to learn two different protocols. Also, accelerator keys are not possible in keyboardless pen-based systems. Another problem is that accelerator keys may be awkward when hands are needed for other tasks (e.g., manipulating the mouse with the right hand and controlling a VCR with the left).

Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the ACM copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Association for Computing Machinery. To copy otherwise, or to republish, requires a fee and/or specific permission.

Permission is granted to distribute this material for use in non-commercial applications. CHI94-4/94 Boston, Massachusetts USA
© 1994 ACM 0-89791-650-6/94/0258...$3.50

Marking menus are designed to overcome these problems. Using a marking menu with a pen based computer works as follows. A user presses down on the screen with the pen and waits approximately 1/3 second (we refer to this as "press-and-wait"). A radial menu then appears directly under the tip of the pen. A user then highlights an item by keeping the pen pressed and making a stroke towards the desired item (see Figure 1a). The alternate way of selecting an item is by drawing a mark. This relies on the user recalling the location of the item in the menu. A mark is drawn by pressing the pen down and immediately moving in the direction of the desired menu item (see Figure 1b).¹

Figure 1: Selection using a radial menu (a) and selection by drawing a mark (b).

Drawing a mark avoids the problems with menus and accelerator keys described earlier. A user is not delayed by the display of the menu, and a mark obscures very little of the screen. Selection using the menu and using the mark are very similar protocols. No keyboard is required and this frees the other hand to perform other tasks.

There are other advantages to this approach. Unlike linear menus, marking menus can be operated "eyes free" because selection is based on direction of movement, not position. Hence, they are especially suited to tasks that require attention on other matters (e.g., transport control while watching video).

These advantages have motivated our study and development of marking menus. Previous studies have been

¹ Now at Xerox Palo Alto Research Center, 3333 Coyote Hill Rd., Palo Alto CA 94304, kurtenba@parc.xerox.com, (415) 812-4753

¹ Marking menus may also be hierarchic. This paper deals only with non-hierarchic menus. See [6] concerning hierarchic marking menus.
controlled laboratory experiments to address issues concerning the design and limitations of marking menus (e.g., how the breadth and depth of menu structures affect reliability of marking) [7][6]. In this study we wanted to gain experience and insights in how to design interfaces with marking menus and to determine if marking menus are used as designed in a real work setting. Our design is based on the user requirements that motivate the use of menus and menu accelerators methods. Specifically, the design features of marking menus are aimed at following user requirements:

**Requirement:** Novices need to find out what commands are available and how to invoke the commands. **Design feature:** pop-up menu.

**Requirement:** Experts desire fast invocation. Once the user is aware of the available commands, speed of invocation becomes a priority. **Design feature:** easy to draw marks.

**Requirement:** A user's expertise varies over time and therefore a user must be able to seamlessly switch between novice and expert behavior. **Design feature:** menuing and marking are not mutually exclusive modes. Switching between the two can be accomplished in the same interaction by pressing-and-waiting or not waiting.

Our model of user behavior with marking menus is that users start off using menus but with practice gravitate towards marks. Users that are expert (i.e., primarily use marks) will occasionally return to using the menu to remind themselves of the available commands or menu item/mark associations.

In order to test this model of user behavior and verify that marking menus are used in a real work setting as designed, we collected data on the use of a marking menu in an application being used for a real task. Our analysis of the data answers the following questions: Do users, with experience, go from using menus to using marks? Do users ever go back to using menus? Are marks faster than menus and if so by how much?

**THE APPLICATION**

The application we used to study marking menus was a conversation analysis/editor program, named ConEd, developed at University of Toronto [10]. It is important to note that ConEd was an ordinary application for doing real work, not a concocted test vehicle for marking menus. By digitizing audio from a conversation among four people, data was collected to index who was speaking and when. The conversation analysis/editor program was then used to display this data in a “piano roll” like representation. The program runs on a Macintosh computer. Figure 2 shows a typical display of the data window. The y-axis represents the four participants in the conversation, and the x-axis represents time. A black rectangle indicates that a particular person is speaking for a duration of time (this is referred to as an event). The window can be scrolled to reveal different moments in the conversation. Besides displaying the data, the application can be synchronized to a video recording of the conversation. As the video plays, the application moves a horizontal bar across the window to indicate the current location in the conversation. If the bar moves past the right side of the display, the application automatically scrolls to the next section of conversation.

Typically, a user sits in front of the Macintosh and video monitor, watching the video and editing events in real-time. Such things as coughs and extraneous noises need to be deleted from the data. Other pieces of conversation, such as laughter, must be tagged for later analysis. Very often events must be added or extended because the automated speaker tracking system was not accurate enough.

A marking menu was used in ConEd for the six most frequent commands on events: laugh, delete, add, fill-in, ignore, and extend. The menu can be popped up by pressing-and-waiting in the “piano roll” window (see Figure 2). Alternatively, a mark can be drawn to select a command. See Figures 3 through 8 for a description of how these commands work.

**Discussion of design**

Based on our earlier experiments [7] and experience designing ConEd, we discovered several valuable design principles for placing marking menus in an interface. These can be summarized as follows:

- **Restrict menus to even numbers of items, up to twelve.** This enhances marking performance.
- **Place commands in marking menus that will be used frequently.** This speeds up the operation of the interface and helps users learn the marks.
- **Make use of the end points of marks.** This combines the operation of object selection and command specification.
Spatial commonalities between the representation being edited and the direction of menu items can be used to determine the assignment of directions to commands. The add and extend commands are examples of this (see Figures 4 and 5). Both these commands require a vertical time location value. A common way to indicate location along the horizontal is by a vertical “tick”. This serves as a mnemonic for the marks associated with these commands.

If possible, once a function in a menu is invoked, it is replaced by the corresponding inverse function. For example, the laugh and ignore commands can also be undone by repeating the laugh or ignore command on the same event. This allows extra functions to be associated with a menu without increasing the number of items in a menu (i.e., special “unlaugh” and “unignore” menu items were not needed). Furthermore, it provides a mnemonic to help recall the association between mark and function (e.g., “unlaugh is in the same spot as laugh”).

The basic idea of these design principles is to make menu item/mark associations easier to learn, and interaction efficient. We recommend following these principles when placing marking menus in any type of interface.

ANALYSIS OF USE

The behavior of two users using ConEd over an extended period of time was studied. We focused on only two users because our previous studies of marking menus focused on many users over a short period of time (i.e., an hour) in laboratory experiments. In this study we wanted to carefully examine individual behavior over much longer periods of time (i.e., hundreds of hours). Even with only two users the data analysis was substantial. The results of this small study can be used to direct future studies with larger numbers of users.

Both users in our study were employed to edit conversation data. The edited data was used in a research project that was independent of marking menu research. Therefore, a user’s main motivation was not to use marking menus, but to complete the task of editing and coding the huge amounts of data as quickly as possible.

The first user (user A) was an experienced Macintosh user who was also familiar with video equipment and conversation analysis but unfamiliar with marking menus. The second user (user B) was a Macintosh novice. User B had to learn how marking menus worked, the many details of the Macintosh interface, and the correct way to edit the conversation data. Users had the interface to ConEd explained to them and some example edits were performed for their benefit. In particular, the commands in the marking menu were carefully explained and demonstrated.

Data on user behavior was gathered by recording information about a marking menu selection every time a selection was performed. A user only needed to register his or her name at the start of an editing session. The rest of the trace data was accumulated transparently. After completing the task, the users were asked to fill out a questionnaire on
their experiences using marking menus. The intention of the
survey was to reveal users' perception of marking menus
and gauge their level of satisfaction.

Hypotheses
We formed the following hypotheses about user behavior
with the marking menu in ConEd:

Marks will dominate: Use of the menu will dominate a
user's behavior at first. However, with experience, the use
of marks will dominate.

Marks for frequent commands: The more frequently a
command is executed the more likely it is to be invoked by a
mark.

Marks faster than menu: Time to select from the menu will
be greater than time to make a mark.

Marks get shorter and faster with time: With experience,
the average length of a mark and time required to make a
mark will decrease.

Results
User A edited for a total of 8.55 hours over approximately
six days. User B edited for 10.1 hours over a 29 day period.
Most editing sessions lasted one to two hours. The amount
of coding and editing required was extremely high. Over 18
hours of operation, the two users performed 5,237
selections.

We analyzed the data from the two users separately for
several reasons. First, we were concerned with individual
differences and combining the data would have masked
these differences. Second, this study was not a controlled
experiment. The data being edited varied, as did the amount
of time and number of sessions the users worked. Thus,
there was no logical way to merge the users' trace data.
Finally, our two users were very different in attitude and
expertise, and therefore combining the trace data would
have been inappropriate.

Menu versus mark usage
The hypothesis "Marks will dominate" was shown to be
true. Figure 9 shows the percentage of times a mark was
used to make a selection (as opposed to using the menu to
make a selection) versus the total number of selections
performed. Over time, marking dominated as the preferred
mode of selection. For user A, out of a total of 3,013
selections 6.6% used the menu. For user B, out of a total of
1,945 selections, 45.7% used the menu.

There are several interesting observations concerning the
usage of marks over time. First, when users returned to
using ConEd after a lay-off period, the percentage of
marking dropped. Figure 9 shows that several long lay-offs
from ConEd occurred during the study. Note the
corespondence between periods of inactivity and dips in
mark usage. This indicates that mark/command associations
were forgotten when not practiced. However, the amount of
fading reduced with the amount of experience (i.e., the dips
in Figure 9 become less pronounced with experience).
Second, note how user B's mark usage rises dramatically at
approximately 650 selections. We believe the reason this
happened was because user B was a very cautious and
inexperienced user. User B commented that it took her
several hours to get comfortable with the video machine and
the Macintosh interface before she could begin to think
about using marks.

The hypothesis "Marks for frequent commands" is based on

Figure 9: With experience, marking becomes the dominate method for selecting a command. Percentage of mark usage
was measured every 50 selections. Usage of ConEd spanned many days with "lay-offs" between sessions. Note that
after a layoff, a user had to resort to the menu to reacquaint oneself with the marks (especially user B).
the assumption that frequent use demands fast interaction and this motivates a user to learn the association between mark and command. This hypothesis is shown to be true by a strong correlation between the frequency at which a command was used, and the frequency at which that command was invoked by a mark (for user A, \( r^2 = 0.81, p<.05 \); for user B, \( r^2 = 0.88, p<.05 \)).

Selection time and length of mark
Selection time is defined as the time elapsed from the moment the mouse button is pressed down to invoke a marking menu, to the moment the button is released, completing the selection from the menu. This measurement applies both to using the menu and drawing a mark. Selection time, for both users, was substantially faster when drawing a mark than when using the menu. Figure 10 shows these differences. For user A, a mark was seven times faster than using the menu. For user B, a mark was four times faster. Thus hypothesis "Marks faster than menu" is shown to be true.

Even though using the menu and drawing a mark require the same type of movement, using the menu is slower than drawing the mark. There are several reasons why. First, a user must press-and-wait to pop up the menu. This delay was set to 0.33 seconds. However, as the fourth column in Figure 10 shows, even with this delay subtracted from the menu selection time, a mark is still much faster (i.e., user A is 4.2 times faster; user B is 3.0 times faster).

What is the source of this difference? The user most likely waits for the menu to appear on the screen. Displaying the menu takes the system about 0.15 seconds. The user must then react to the display of the menu (simple reactions of this type take no more than 0.4 seconds, according to Card, Moran, & Newell, 1983). However, when making a mark, the user does not have to wait for a menu to display and react to its display. Thus, a mark will always be faster than menu selection, even if press-and-wait was not required to trigger the menu. Figure 11 graphically shows this. The fourth column of Figure 10 provides evidence of this.

Selection time, using a mark, decreased with practice, however the decrease was very small. In view of the very fast times for marking performance, this is good news, since this means that, even early in practice, novice performance was very similar to expert performance. The decrease in selection time was less than 0.1 seconds. For this analysis we used the Power Law of Practice (performance time declines linearly with practice if plotted in log-log coordinates [11]). Linear relationships were found for both users (an analysis of variance of linear regression used: for user A, \( F(1, 1654) = 166.5, p<0.0001 \); for user B, \( F(1, 541) = 23.03, p<0.0001 \)).

The average length of a mark decreased slightly with practice for user B, but not for user A (an analysis of variance of linear regression used: \( F(1, 2813) = 10.82, p<0.01 \)). The average length of a mark was approximately one inch. The delete mark was excluded from this analysis because its length was used to indicate a range of events.

Given these results the hypothesis "Marks get shorter and faster with time" only holds for user B. User A's mark time decreases with practice, but length of mark did not. Further studies with more subjects are required to sort out individual differences concerning this hypothesis.

Users' perceptions
An important parameter not captured in the trace data was selection errors. The reason for this is that prior to a selection we did not know what item a user intended to

<table>
<thead>
<tr>
<th>average time to perform a selection (seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>mark</td>
</tr>
<tr>
<td>User A</td>
</tr>
<tr>
<td>User B</td>
</tr>
</tbody>
</table>

Figure 10: On average, marks were much faster than using the menu. For user A, a mark was seven times faster than using the menu. For user B, a mark was four times faster. Confidence intervals are at 95%.

Users’ perceptions
An important parameter not captured in the trace data was selection errors. The reason for this is that prior to a selection we did not know what item a user intended to...
select. Therefore, when a selection was made, we could not
tell whether or not the user had successfully invoked the
desired selection. Since users should be the judges of what
acceptable error rates are, our questionnaire simply asked
them how many errors they made with the marking menu:
no errors, few but acceptable, or too many? Both users
reported “few errors but acceptable”.

Users perceived marking menus as a tool that helped them
get the task completed quickly. Both users reported that
their performance with the marking menu was “fast”. User
B, however, reported she didn’t have enough regular
experience with the marking menu to be completely
comfortable drawing marks. Both users confirmed the
differences we found in performance between using the
menu and drawing a menu—both users reported a mark was
“much faster” than using a menu.

MARKING MENUS VERSUS LINEAR MENUS
The results from this study allow us to build on the
comparison between marking menus and linear menus.
When a user is familiar with the layout of a menu, selection
from a radial menu will be faster than selection from a linear
menu. Callahan et. al., [2] provide empirical evidence of
this for eight-item menus. It is possible that a linear menu
may be more suitable when there is a natural linear ordering
to the menu items and a user must search the menu for an
item before making a selection. Alternatively, a radial menu
may be more suitable when there is a natural radial ordering
of menu items. However, as shown by both Card [3], and
McDonald, Stone, & Liebelt [8], the effects of organization
disappear with practice. Callahan et. al., provide evidence
that, for eight-item menus, even when menu items have a
natural linear ordering, selection using a radial menu is still
faster and less error-prone than selection using a linear
menu.

Drawing from data in an experiment by Nilsen [9], we can
directly compare six-item marks and six-item pop-up linear
menus. However, we caution that the data being compared is
from two different tasks and user populations. Nevertheless, a comparison of the data and possible explanations is interesting. In Nilsen’s experiment, a
selection from a six-item linear menu required on average
0.79 seconds. In our study, user A and user B required, on
average, 0.18 and 0.40 seconds respectively to perform a
selection using marks. Furthermore, in Nilsen’s experiment
the subjects’ only task was to select from a linear menu.
Therefore, one would expect selection speed to be
artificially fast. In our study, in contrast, the users were
performing selections in the context of other real world
tasks. Thus, we can conclude, with caution, that if menus
contain an even numbers of items and less than ten of them,
and users frequently use the menus, marking menus will
have a distinct advantage over linear menus.

As a practical example of the impact of this speed-up, we
can consider the performance of another real user using

CONCLUSIONS
This study demonstrated several things:

- A marking menu was a very effective interaction
technique in this setting. Its effectiveness was contingent on
applying the technique to an appropriate setting—
specifically, using a marking menu to invoke a few
commands that are used frequently, and require a screen
location as a command parameter. Also, despite the
simplicity of the mark, features of the mark, such as the start
and end points, and the orientation of the mark, can be used
to make interactions more efficient and easier to learn.

- A user’s skill with marking menus definitely increases
with use. A user begins by using the menu, but, with
practice, graduates to making marks. Users reported that
marking was relatively error free and empirical data showed
marking was substantially faster than using the menu. The
data strongly indicates there is an advantage in drawing
marks even if menus could pop-up instantaneously.

- The ability to switch back and forth between menus and
marks is important. When a user’s skill depreciates during a
long lay-off period, the user utilizes the menu to reacquire
skills. Our results lend evidence to the notion that for
interfaces in general, support for switching between novice
and expert mode at the interaction level is a utilized and
important feature.

Marking menus could be appropriate in many other
applications besides ConEd. For example, Microsoft Word
has seven groups of function icons that appear in the
“ribbon” and “ruler” display area. These icons could be
grouped into seven marking menus containing four or less
items. Each group of icons could be replaced by a single
icon which when pressed displays a four-item marking
menu. The elimination of icons would allow space to
display more text, or other or larger function icons (larger
icons make pointing to them easier). The graphics editor in
Microsoft Word already has tool pallet icons that work this
way but uses pop-up linear menus. The popular Macintosh
drawing program called Canvas also uses a similar scheme.

Marking menus, however, are not appropriate when the list
of items changes dynamically. In this situation, users can
still use the menus but will never graduate to using marks
since menu item locations change.

1 A third user used ConEd extensively over a long period of
time but she was not included in this study because she
assisted in the design of the marking menu used in ConEd
and ConEd itself. Therefore, we felt she would not be an
unbiased user of marking menus.
FUTURE RESEARCH
We have been developing hierarchic marking menus [6] that use hierarchic pop-up radial menus and "zig-zag" marks to select from the hierarchy. Based on the results here, we expect marks for hierarchic marking menus to exhibit the same performance improvements as marks for non-hierarchic marking menus. However, the magnitude of the speed-up is an open question. We believe that the speed-up may be even more dramatic than with non-hierarchic marking menus.

We are continuing to experiment with combining marking menus with two-handed interaction techniques such as floating pallets controlled by the left hand and toolglass widgets [1].

ACKNOWLEDGMENTS
We thank the members of the Input Research Group at the University of Toronto who provided the forum for the design and execution of this project. We would also like to thank Karen Weber and Tom Moran for their comments on drafts of this paper. This work was performed in the Dynamic Graphics Project laboratory at the University of Toronto, to whom we are grateful. We gratefully acknowledge the financial support of the Natural Sciences and Engineering Research Council of Canada, Digital Equipment Corporation, Xerox PARC and Apple Computer.

REFERENCES


