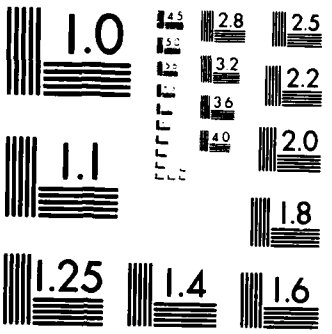
AD-A150 561 LEARNING TO USE AN OFFICE SYSTEM WITH AN ON-LINE TUTORIAL(U) IBM THOMAS J WATSON RESEARCH CENTER YORKTOWN HEIGHTS NY J M CARROLL ET AL. 26 JUL 84 1/1

UNCLASSIFIED RC-10644 F/G 5/9 NL



RC 10644 (#47740) 7/26/84
Computer Science/Cognition 12 pages

Research Report

LEARNING TO USE AN OFFICE SYSTEM WITH AN ON-LINE TUTORIAL

John M. Carroll
Sandra A. Mazur

Computer Science Department
IBM Watson Research Center
Yorktown Heights, NY 10598

IBM Research Division
San Jose · Yorktown · Zurich

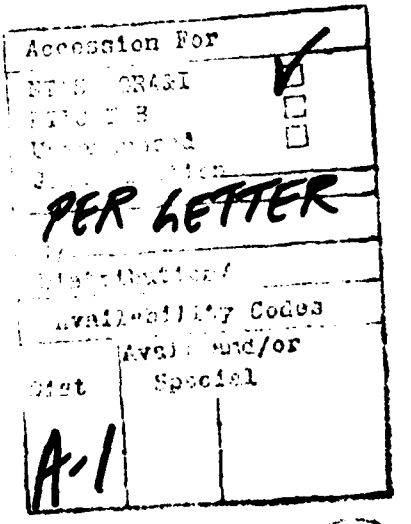
RC 10644 (#47740) 7/26/84
Computer Science/Cognition 12 pages

LEARNING TO USE AN OFFICE SYSTEM WITH AN ON-LINE TUTORIAL

John M. Carroll
Sandra A. Mazur

Computer Science Department
IBM Watson Research Center
Yorktown Heights, NY 10598

Abstract: On-line instruction is replacing traditional self-study manuals as the primary method of user self-instruction for office information systems (Seybold, 1981, 1984). The hope is that this method of instruction can be more effective than training via manuals by affording interactive response to learner actions. In this paper, we report an exploratory study in which six professionals learned to use a commercial office information system by means of an on-line tutorial. We found many of the same learner problems discovered earlier in the context of self-study manual training. Our goal is to describe and analyze this learning experience -- particularly the problems -- in enough detail to be useful to designers of on-line tutorials.



INTRODUCTION.

People who do not already know how to use computing systems often have a difficult time learning how to. Over the past three years, research in our laboratory and elsewhere has focussed on these learning problems and on finding solutions to them in better designs for interfaces and for training tools (Carroll and Carrithers, 1984a, b; Carroll, Ford, Smith-Kerker, and Mazur, to appear; Carroll, Mack, Lewis, Grischkowsky, and Robertson, 1984; Mack, Lewis, and Carroll, 1983). Our goal is to develop an understanding, with both scope and depth, of how new users interact with user interface elements, an understanding that can inform the design of future technology.

Up to now, our work has focussed on contemporary self-study manuals and on a particular population of users, namely office clericals who have had no prior experience with computer equipment. One direction in which to expand the breadth of this work is to consider other training approaches and other use groups. Indeed, much attention in the industry has shifted to "on-line" tutorial approaches, in which training lessons are presented by the system instead of by a manual, and to "professional" people as a user group. Professionals have more personal prerogative in determining which system they will use and what they will do with the system. The problems of getting started with a system are quite different for professionals: a system that fails to be easy to learn will not merely be an obstacle and an inconvenience for the user, it may simply be discarded or returned to the vendor for a refund! Professionals, unlike clericals, may take far more responsibility and initiative in deciding how to deal with their computer equipment, and in particular with problems in ease of learning.

This is the context for the present study of professionals learning to use a commercial office information system via an on-line tutorial. We have deliberately suppressed the identity of the particular system we studied, focussing our remarks on characteristics and problems that seem general in nature for current on-line tutorial approaches.

WHAT WE DID.

Our study is exploratory, and we begin with disclaimers. We studied only six individuals, staff members at a research laboratory. These people may very well be atypical examples of professionals. We did not perform any rigorous benchmark studies: we were not concerned with whether typing a mailing label is faster in the system we studied than on some other system. Rather, we tried to focus our attention qualitatively on the salient elements of the on-line tutorial, and on their implications for learners.

The system we studied is menu-driven. It presents menus and data in display windows on a high-resolution screen. Windows are activated and controlled by mouse-pointing and selection. The on-line tutorial material appears in its own display window, presenting direction to learners for carrying out various pointing and selection exercises in other concurrently displayed windows. The learner is to work through a series of tutorial units each consisting of several step-by-step exercises -- much on the same way learners are to work through typical self-study training manuals. However, the on-line tutorial manages the entire interface during these exercises, and can monitor the learner's success to correct errors and deliver feedback. Thus, the learner is provided with a far more interactive self-study environment.

Our six learners responded to an on-line request for volunteers that we placed on our local computer network. We excluded volunteers who had substantial programming experience, but all of the six we accepted had some experience with computers; a few had had some programming experience. We asked our learners to volunteer for two 3-hour sessions, promising free coffee and the opportunity to learn to use a "new integrated work station design". Our group included a lawyer, a graphics designer working in patents, a publications

librarian, a documentation designer, a researcher in automation design, and a research team manager.

At the beginning of the first session, we explained our ground rules: Learners were to work on their own; to imagine a real office situation in which they were learning to use their own new personal workstation (one that nobody else had yet); we were to be regarded as inert (albeit keenly interested) observers. The learners were presented with the system and two manuals (the basic training manual and the planning application book). They were asked to learn to use the system, and in particular to learn to use a planning application, selected because it seemed to be the most professional-oriented of the system's applications. We asked the learners to think aloud as they worked, and we prompted throughout the sessions as necessary to keep this self-disclosing monolog going.

The two of us kept independent notes on what was said and what was done during the six hours. Later, we compared and collated our observations. The inventory of critical and typical episodes and incidents culled from these notes on the 36 hours of learning we observed are the basis for our description. Our plan going into the experiment was to have our participants spend roughly the first 3-hour session covering basics (including the on-line tutorial), and the second covering the planning application and then performing a simple transfer of learning test (we asked them to plan the schedule for nine phases in the construction of a 3-bedroom house). Material accompanying the system we studied indicated that this would be more than enough time.

In Table 1 we have summarized the backgrounds of our six learners (L1-6) and the amount of time they apportioned to two major aspects of their task.

Table 1
Learner Background and Performance Summary

	L1	L2	L3	L4	L5	L6
Backgrounds.						
Prior experience with computers (years)	1	6	4	4	6	3
Number of different systems ever used	2	3	3	1	5	6
Average current computer use (hrs/wk)	9	3	7	28	15	30
Performance						
Use of on-line tutorial (hrs)	2.3	1.1*	3.1	0	1.2	1.1
Use of planning application (hrs)	0	2.3	0	0	4.3	.7
Attempted planning application task	N	Y	N	N	Y	N

*Skipped three units.

On average, the five learners who were able to try the on-line tutorial spent over one and three quarters hours on it, more than three times the half hour estimated by the system's designers (L4 made a single error that totally obstructed his progress). Moreover, two of the five were not able to use the tutorial until their second session (that is, after three hours of previous experience with the system); and one of the five skipped three units of the tutorial. Only three of the six managed even to try the planning application, and one of these was unable to accomplish very much at all. The two others spent an average of 3.3 hours to learn the basics and try the project we suggested to them.

In sum, we found a rather lower level of success than we had been led to expect. Although our learners had had experience (with an average of 4 systems over an average of 3 years), they did not all finish the tasks we suggested. Indeed, L4 and L6 -- who both spend more than half their work time using computers -- were among the learners who failed to get to our planning application transfer of learning task. But these performance benchmarks tell us little in and of themselves. In the balance of this discussion we focus on problematic aspects of the learning situation as we observed it in order to better understand why performance was poorer than we expected, and to understand how the work in on-line tutorial design might advanced with respect to these problems.

GETTING STARTED.

We have seen many of these problems over and over in prior work with other systems. For example, learners often fear that they can do something so wrong as to damage the system. As L1 said: "Will I damage this if I turn it off with the disk in? There is a terror of destroying things, so I'm still a bit tentative." This kind of fear was salient and typical at the start of learning, but in the case quoted L1 had already had 4 hours of experience with the system.

Reading. Most of our learners did not want to read -- as they demonstrated by their behavior in the experiment. Nonetheless, most entered into the experiment by announcing that they were the type of person who carefully read everything before trying to do anything. Two of them brought paper and pencil with which to take notes on their reading. For example, L4 apologetically informed us that we would have to watch him read both the basic training manual and the planning application book before he would use the system. Two minutes later he had turned the system on. L6 began the experiment by picking up the basic training manual and announcing that his style was to first read everything thoroughly. Indeed, he actually read the manual for less than 9 minutes before switching his attention to the system. He next referred to the manual almost 2 hours later.

L5 was the only participant who was really very successful at reading. But there was an ironic twist to this. She was unable to load the on-line tutorial diskette on her first day, because she inserted it into the disk drive in the wrong orientation. Due to this problem, she turned to the manual, spending most of her first 3-hour session reading and following printed exercises. Curiously though, L5 was also one of the two participants to manage to get to the planning application transfer test. But her success was only relative. She had many problems coordinating the manual descriptions with events on the display, for example, only 20 minutes into her first session she exclaimed "I'm wondering as usual why it's not doing what the manual said it was going to."

Learners also found the manuals specifically lacking in some instances. L6 was afraid to simply turn the system off, thinking he might damage it. He checked the index of the basic training manual for help, and was directed to page 118. This turned out to be page 118 (that is, capital I -- 18). In his second session he wanted to refresh on some concepts but could not find any index entries for "cursor" or "pointer", nor any relevant entries for "mouse" or "moving".

Misreading. Often when learners did spend time and effort on reading they became snarled in confused interpretations and cross-references. In his 9 minutes of reading, L6 became troubled when he understood section B of the manual to have referred him to section D, only to have section D then refer him to section B. Later, he became tangled in a loop between two pages within a section. One page seemed to him to be saying that to create a document or folder (directory), you must make a paper pad, and the other page seemed to be saying that to make a paper pad you must create a document or folder.

L4 read a reference to a Figure 1, and associated the reference with an unlabeled figure of a calculator icon appearing on the same page. He never noticed that there indeed was a Figure 1 (labelled as such) on the very next page. He became momentarily confused, and then irritated, by the fact that page A-4 of the basic training manual suggests that the user read section A of the basic training manual. As L2 put it, "If I were a machine, I'd be in an infinite loop." L4 was also confused by references in the basic training manual to "controller cards" and his discovery of quick-reference cards under the keyboard. This initial confusion was unresolved for almost 50 minutes. At that point the learner seemed to see the difference and became irritated: "What's it going to help to know about controller cards. I'm not going to repair it; I want to use it!"

Skipping. When they did read, the learners tended to skip around in the manual. L4 simultaneously read from the basic training manual and the planning application manual, juggling the two in his lap, skipping between units and sections. L2 asked to see some of the other manuals, for example one for a data base application, and then skipped around in an even larger set of books. L5 was also juggling manuals. She was at first following the basic training manual, but failed to load the on-line tutorial diskette correctly. When the system booted, an initial desk-top layout was displayed -- which she recognized in a figure from planning application manual: "Oh! I have the wrong disk in. I thought I had <the tutorial> in but I have <the planning application> in." On the basis of this coincidence and its misinterpretation, she then skipped to the planning application manual for the remainder of the day.

Turning The System On. Problems associated with initially reading the manual were generally less troublesome than those associated with coordinating a reading of the manual with initial use of the system. L6 tried to follow an instruction that said: "when you hear a click, hold down the <Alternate Shift> key". He wondered whether he had heard a click (there were of course a variety of tiny ambient background noises emanating from the system as well as from adjacent rooms of our laboratory). He concluded that he had never heard a click. However, he went ahead and pressed the <Alternate Shift> key combination anyway, but by that time he had missed the cue, an error panel appeared: "<Alternate Shift> key combination not associated with available function". This series of problems led him to conclude that he could not "learn by doing", as he wished; he therefore decided to go through the on-line tutorial: "I was hoping to skip <the tutorial>, but I guess I need to go through it. I'm missing something".

The hard disk. L2 and L5 both turned on the system before starting the hard disk unit. L5 restarted from scratch when she saw her mistake, but L2 merely worried "That might make a difference or it might not make a difference." He was presented with a diagnostic error panel, but at that time was also reading in the book that the system took a long time to initialize, so he gave it more time. After several minutes, he began to consider other possibilities; he turned the system off, checked the orientation of a diskette he wanted to load, and then tried to start-up again. He read several troubleshooting sections of the basic training manual, and then tried to load his diskette in the lower disk drive. By failing to turn on the hard disk unit first, he had made unavialable the system code stored there. Finally, toward the end of a 47 minute period, he tried to load another diskette. Apparently by chance, he chose a backup of the system code. At that point, he was presented with several further error panels, which he did traverse successfully to initialize the system.

Response Time. Many miscoordinations devolved from the extremely slow response time of the system. All of the 5 learners who managed to get the system initialized at all complained about response time. And this was more than inconvenience. In more than one case it led to serious trouble. L3 had been trying to work on his own early in his second session, but had not been succeeding. He decided to reload the tutorial to refresh his memory, and in order to do this he had to turn the system off. He did that correctly but became impatient and

pressed the on/off button before the system had finished its shut down. In doing this he hung the system up. Since he had experienced relatively long response times before in the shut down situation, he just read on a bit in the manual. After 18 minutes, he was sure something was wrong and asked us to intervene.

L4 committed a fatal error in similar circumstances. He was able to start the hard disk unit, but became impatient during the long initialization period after pressing the on/off switch. This impatience led him to press the switch several more times before the system had booted, and indeed to press an assortment of keyboard keys. A side effect of all this was that an error panel appeared on the screen. Since he had not yet developed any skill in manipulating the mouse pointing device, he was unable to really progress with the error panel. However, unfortunately, he did try out the mouse button, clicking it several times. This also had an unintentioned side effect, namely to cause one disk drive to be specified as the primary input device (that is, instead of the hard disk, which is so specified by default).

At this point he was doomed as a new user. Each time he tried to start the system, he got error panels. Since the on-line tutorial diskette does not contain its own system code, he was unable to load it. Indeed, from the point of this error, this participant continued only one hour further. At that point, he gave up in despair and asked to be excused from the experiment.

THE ON-LINE TUTORIAL.

Typically, the learners tried to go it on their own -- at first. In every case, however, they made some obstructing error or became confused. For example, L6 was confused about the click he was supposed to hear while the system was booting, and L4 wondered why all the diskettes were stamped "Release 10", and what had happened to 1 through 9. These uncertainties convinced them to try the on-line tutorial. As L6 put it: "I guess I need to go through it. I'm missing something."

Five of the 6 learners tried to load the on-line tutorial after having already booted the system; this caused an error panel to appear on the screen telling them to turn off the system and then turn it on again with the on-line tutorial diskette already loaded. All five were surprised that the system had to be turned *off* in order to load a new diskette. L4, reading this in the basic training manual, convinced himself that there was a typo: "...'be sure <the system's> power is off'; it should say 'on'; it's a mistake". L5 and L2 had problems in loading the tutorial diskette which ended up diverting them from it. Both, however, made a fresh start in their second sessions, and at that time managed to load and run the tutorial.

Rote Learning With Reinforcement. The on-line tutorial consists of a series of unit lessons, each consisting itself of a series of exercise steps. One problem with this format is that the exercises are introduced, one after another, with no intrinsic rationale. They are to be followed by rote. When told to move the mouse in circles, L1 exclaimed: "Why are they telling me to do this?" The exercise seemed pointless to him.

Sometimes it happened that the method prescribed by the on-line tutorial was less efficient than one stumbled upon by the learner. While practicing replacement of text, several learners successfully and spontaneously employed the method of Backspacing and retyping (instead of selecting a text range with the mouse and then typing to replace). They were disturbed to find the system advising them to be inefficient. L5 simply couldn't believe that transactions with the Garbage Can (used for deleting data objects) could be as complicated as they seemed in the tutorial exercise (she was relieved later to learn that they can be simplified by short-cuts).

Sequencing. Another problem is that the exercise steps are written so that warnings occur *after* the opportunity to make a mistake; often information is provided *after* it would have been relevant. Throughout the early the tutorial units, L1 was surprised that the cursor changed shape (e.g., from an arrow to a pair of opposed braces) as it is moved about. He was certain there was an important reason for this, but could not find out what it was. He was finally quite annoyed that it was not until the end of the first unit of the on-line tutorial that he was actually *told* that the cursor can change shape (he was still not told why!). Subsequently, he was instructed to select a Demo, but only then warned that he need not have selected it if he had done alright on the immediately preceding exercise -- but he had already selected the Demo at that point.

A more costly example arose in the "Stopping <the tutorial>" unit. L1 executed the first step without reading ahead, and accordingly ejected the tutorial diskette: "Oh, I didn't mean to do that. Now I'll have to start all over." He did in fact start all over, going step by step through the tutorial a second time. The tutorial does provide a "Units" pop-down menu so that learners can accelerate through the training if they wish. L1 found the Units pop-down menu, and was able to use it, displaying a selectable index of all the tutorial units. However, he used this merely to confirm his current state, selecting the Unit he was already in (Unit 1). Instead of skipping back to the "Stopping <the tutorial>" unit, he kept re-initiating Unit 1: "I'm in a loop here."

The printed material in the basic training manual (accompanying the on-line tutorial) also presents information that is out of order with respect to the needs of the learner who has to execute it. As L1 put it: "I've already loaded the disk -- *now* they tell me not to touch the shiny plastic!" L2 said it would have helped him coordinate sequences of instructions if the tutorial screens had been printed in the basic training manual. The planning application manual also seemed to have ordering problems, although as we have noted only two of the learners got to use it very much. In one example, L5 was following the planning application manual to draw a particular diagram from a figure in the manual. But upon turning the page she was shocked to find that there was a step by step procedure for drawing that diagram, out of order and too late to help her.

Step complexity. The individual exercise steps in the tutorial were often quite complex. Even when problems of out-of-order instructions and warnings were avoided, learners could get lost within the exercise steps. For example, L2 named the same object twice in immediate succession. He was following the manual and simply overlapped himself. L5 re-read instructions for opening a folder icon immediately after having completed them and became confused about whether she had in fact completed them (the instructions, after all, were still there on the screen directing her to follow them). She tried to comply, by opening another folder icon (one with a name similar to the one she had already opened in following the instruction the first time). However, the system had disabled the menu choice to open *because* she had just opened an icon. Menu selection in the system is context dependent; menu choices inappropriate to the current context are dimmed and cannot be selected. Thus, if an icon has just been opened, the Open choice is dimmed, the Open function is not available. L5 knew something was wrong in this episode, for she had used the menu before to open folders. But she could not figure out *what* was wrong or what she could do about it.

L1 and L2 had trouble with a tutorial panel containing five steps. Just prior to the first step there was a heading saying to read both instructions before doing anything. L2 merely read the first two steps and then focussed on a second heading just prior to the third step, a heading that began with words "Do this ..." He then executed the next two steps. Because he had skipped the very first two steps, he was unable to properly execute the subsequent steps. Problems following seemingly arbitrary instruction sequences, like the foregoing, are very similar to those we have seen in learners following self-study manuals (Mack, Lewis, and

Carroll, 1983). They underscore the generality and importance of such problems, and call into question an on-line tutorial design approach that merely transliterates self-study training manuals into the interactive medium.

Learner control. The tutorial provides positive reinforcement messages to the user each time an exercise step is correctly completed ("Excellent!", "Right!", "On we go."). L6 found these to be "fatuous"; "It's silly to say Excellent, when they just told you that you did it wrong!" L2 felt it was insulting to be commended for following an instruction that was not very difficult. L1 became snarled in an error loop of repeated Get Paper operations, but when he did finish the step the system said "Excellent!" -- to which he replied "the Hell it is!"

Fundamentally though, the learners seemed to resist the rote learning track whenever possible. For example, L1 and L6 routinely executed the exercise summaries (which are intended to be "read only"). L1 detested the Demos (which allow a learner to watch an exercise step performed before attempting to do it): "I can't stand the Demo because it's set up for a 4th grade reading speed". L3 expressed impatience with the tutorial, focussing on the summaries: "It might be nice to give you the opportunity to do more, without running through this trivial example." On his second transit through the tutorial, L3 often jumped the gun on the exercise of particular skills, for example, he practiced scrolling prematurely. In many of these cases, the tutorial obstructed his practice since he was out of sequence for the tutorial. L1 resented error panel messages implying that although he had done something wrong, the system would let him go on anyway. In these cases, he tended to prefer to go back and redo the exercise step -- and thereby take control away from the tutorial.

It seemed that the learners wanted to do something more concrete than to follow a series of rote exercises. L2 balked at instructions to read a passage but not to do anything: "I'm tempted to do it anyway and then see if I can get out." Indeed, he moved the mouse during a Demo, when instructed explicitly not to do so -- wondering why the tutorial didn't reprimand him for this. Learners often tried to practice on their own within the on-line tutorial, L6: "I just read how I could make a paper pad from the menu, but I went to the menu and I couldn't do it. They should let you make a note here." After an hour of the tutorial, L1 complained "I'm getting impatient. I want to do something, not learn how to do everything." After a hour and a half, he said "I could have typed 3000 words by now".

Error Shielding. In many places the tutorial permits small user errors, correcting them automatically as the user passes on to the next exercise. Sometimes the user is informed that such and such a response is being altered to whatever. At other times, no feedback -- or incomplete feedback -- is provided. The response is altered surreptitiously, as it were. A typical example was that participants used names they had created in their practice memos instead of the names suggested by the tutorial. At the end of the exercise step, the tutorial puts in the name it had suggested and go on, briefly putting up a panel that explained that the user's name had been replaced by the name the system had originally suggested. Usually this caused only minor surprise and/or frustration.

Over-Shielding. After such an error and error correction, the system offers several choices: Practice, Demo, Back, and Continue. L6 initially found these to be too telegraphic: "'Practice' what? 'Demo' of what? 'Back' to where?"; he picked Continue, the only choice he had not eliminated. Still, he complained "It doesn't tell you to go to Continue at the end of an exercise." Later, he did try the Back option, to find out why a name he had typed in had been replaced. However, the tutorial did not take him back to the explanation about names, but rather back to the prior exercise step (the point at which he was originally asked to type in a name). Indeed, the only way he could truly go back to the point he wished to go to was to recommit the same naming error! L5, who made very few mistakes in the tutorial, wondered

why the system couldn't just go on and spare her the trouble of explicitly selecting Continue after each step.

L1 tried to rename an object, but misspecified the object (by miscoordinating a selection via the mouse). He typed in the new name without consequence -- there was no beep, no error message, no feedback at all. At the end of the step, when the system asked if he was sure that he understood (intended as tactful and positive reinforcement), he laughed but went on anyway. The tutorial, however, put in the name he had been unable to put in, and several minutes later he was quite surprised to find it there. In a converse example, L1 (as well as L2 and L5) typed in several fields of a practice memo that he had not been told to type in. This time the tutorial deleted his extra material at the end of the exercise step. He was also surprised to find this material gone, and retyped all of it. (Of course, the tutorial again removed it on the next exercise step.)

In another incident, L1 selected "set aside" instead of "save and put away" in the Main menu. At the end of the exercise step, the tutorial informed him of this error and gave him the option of going back and redoing the step. He did this, and correctly used "save and put away", but ignored the rest of the step -- the operations he had already correctly practiced on his first time through. Now having finished the step the second time, he was again informed that he had incorrectly performed the step, namely the other operations, the ones he had had correct on his first try. Again, this suggests that the tutorial exercise steps are at too coarse a grain.

L2 encountered a similar problem with the unit of selecting multiple icons. He correctly completed the tutorial exercise step on this, but began to wonder *why* one would ever want to select multiple icons in the first place. He backed up to the relevant step and re-read the panel. Satisfied, he moved on again, but at that point the tutorial put up an error panel -- he had not actually done the exercise step, he had only re-read it. He felt compelled to work through everything again. Accordingly, he backed up and selected a group of icons, but this time decided to select a different group than he had on the first pass (for variety). But of course, he had again failed to do *just* what the exercise step said, and the tutorial treated his new effort as an error too.

The tutorial's error shielding also prevented learners from practicing individual skills they had mastered. L2, having just learned about scroll arrows tried to operate them on the tutorial window. However, the tutorial window is arranged so that the scrolling operations do not work in it, preventing this sort of practice. This caused frustration and confusion since it made the scrolling operations appear to work unreliably.

Under-Shielding. The foregoing examples suggest that error shielding can create some problems for new users, but there were also cases where correcting errors only at the ends of exercise steps provided too little shielding. In the second unit of the tutorial, L1 spontaneously decided to practice icon selection. In the course of this he positioned an icon so that it partially occluded another icon. He was then unable to coordinate the selection of the occluded icon, which was required for the next exercise step. He tried to correct this in several ways, including performing the exercise step on another icon. But all of his attempts failed. When he gave up and tried to go on, the tutorial automatically corrected the error, but in a sense it was too late.

L2 resized a window such that it was wholly within (i.e., on top of) the tutorial window. Later, when he activated the tutorial window, this other window became wholly buried under the tutorial window (the currently active window moves to the "top" of the display area). Hence his other window became inaccessible for further use. L3 was practicing skills for moving icons around the desktop and accidentally released a folder icon while it was superim-

posed over another folder icon: "I lost Supplies!". By releasing the icon over a folder, he had placed that data object into the folder, hence it became invisible on the desktop.

Error Diagnosis and Recovery. The feedback the tutorial provides for learner errors is in many cases too general to help the learner adequately understand what the problem was. For example, most learners made errors coordinating the selection of icons from the display. The on-line tutorial delivers identical error feedback for the error of naming the correctly selected icon with the "wrong" name as it does for the error of selecting the wrong icon but naming it with the "correct" name (i.e., the name the tutorial suggests). In the second case, the error feedback is all but uninterpretable: L1 paraphrased it as "I blew it."

When directed to put a folder into the Garbage Can and then to recover it, L1 put it first into the Garbage Can, then into a diskette window, and then back into the Garbage Can. The error panel said "you put it into the diskette instead of onto the desktop" -- an inadequate description of what had actually transpired. As in the case of the selection/naming errors, the tutorial provides some support for error diagnosis and recovery, but not enough. However, in both cases the system *purports* to be providing complete support, which leads to confusion for learners. It would be better to make it clear that only "hints" about error recovery are being provided.

In dealing with error recovery, the tutorial often employs somewhat "cute" checkpoint questions, e.g., "Doesn't say Set Aside 'Clipboard'? Redo Step 2" (perhaps to maintain learner motivation). This message, for example, means that if the main pop-down menu does not list as a choice "Set Aside 'Clipboard'", the user should redo Step 2 in the exercise. A few words were saved and a small amount of cuteness was achieved in the wording that was in fact used, but at a price in clarity. L3 read and reread this message several times. Even though he had *successfully completed the exercise*, he backed up a step and began repeating part of his work. At that point he did make a mistake (he selected "Cut" from the edit pop-down menu before selecting text to be cut). In response to this, he elected to back up another step. At that point he chanced to notice that "Set Aside 'Clipboard'" was listed as a choice in the main menu. From this sequence of events he drew this conclusion: "Maybe they're telling me that it really *doesn't* mean set aside Clipboard." Unfortunately, this conclusion is entirely wrong.

Summary. The tutorial was far less successful than we had either hoped or expected. The learners did not like it, did not cooperate with its rote learning approach, and worst of all did not seem to gain much. Most of them spent most of their time running and rerunning the tutorial, but when they went to apply what they had "learned" they could not. Minutes after finishing the tutorial, L1 became puzzled by the fact that there were icons at the bottom of the screen: "What's all this junk at the bottom?" After two further transits through the tutorial, he complained that he hadn't learned about any of the pop-down menu headings on the menu bar (he had in fact used some of them). He summed up: "I am tremendously frustrated. I've done a lot of work and I can't do a damn thing."

L3, who had finished the tutorial 16 minutes earlier, tried to create his own document and simply could not get started: "I have completely forgotten how to open, enter, to create a document!" He seemed only to be able to open and close icons, and to experiment with choices in the pop-down menus. After several cycles, and as if in desperation, he just typed a burst of characters in an open window. Fortunately, the window was a blank document, and he succeeded in typing. Ironically the only two learners who actually succeeded in making serious progress with planning application (recall that this was the overall goal that we gave the learners) were L2 and L5 -- who initially inserted the tutorial diskette in an incorrect orientation, and were therefore unable to use it at all during their first sessions.

REMARKS.

Our study suggests some major problem areas for the design of on-line training. One of these is the overhead of getting started at all. The designers of the on-line tutorial focussed their efforts on the tutorial itself and overlooked the fact that just getting the tutorial loaded and started was nontrivial: half of our learners were stymied by this obstacle. Whatever good ideas the tutorial might have had were initially wasted for these people by being inaccessible. Merely getting started at all should never be this difficult.

Many other specific problem areas can be related to a theme we have developed in our prior studies of training by self-study manuals: learners do not necessarily do what training designers want them to; instead they tend to get actively involved and to think and plan and solve problems; they try to learn by doing (Carroll and Mack, 1984). This active learning thread permeates the description we have given of professionals learning to use an office information system via an on-line tutorial.

Rote learning designs don't work. Many of the specific problems our learners encountered sprang from the rote learning design employed in the on-line tutorial. The tutorial strikes an awkward compromise between truly attempting to provide interactive coaching and merely placing pages from a self-study training manual up on the display. It relies in the end on the learner's passive compliance in reading material from the screen and then in carrying out practice exercises -- for demeaningly cute verbal rewards. Yet as we have seen, our learners did not want to read and avoided doing so. They did not want to practice narrowly prescribed -- and often trivial -- exercises, and they avoided doing so.

In contrast, ad hoc active learning strategies often led to successful outcomes. For example, L2 had trouble understanding the Garbage Can and conceived of several experiments in which he tried to throw away objects, and then opened the Garbage Can to check on his success. He noticed that popping down a menu temporarily halts the printer, and he then tested this several times to be sure he was drawing the right conclusion. He was worried about the automatic dimming of the screen, and tested its connection to mouse movement by experimenting. In each case causes and effects were systematically correlated as the learner actively created his understanding of the system. Making the operation of system functions visible facilitates active learning.

How to design for active learning. In many respects the system we studied was well-designed from the perspective of active learning. Relatively many interface transactions had consequences that were visible to the user and simple enough to afford interpretation. However, the pitfalls of active learning with this system are perhaps more instructive. L1 and L6 were both attracted to the system's Calculator because it seemed familiar and perhaps simpler than the planning application. However, L1 became absorbed in the fact that some of the buttons didn't look as much like buttons as did others. L6 used the calculator but failed to try any memory functions, commenting, "So that's all the calculator can do." Clearly, one can actively rummage around in a domain but not get to the important content. The functional relations of the Calculator were not as familiar as the learners expected and not visible or simple enough to be reasoned out. In both cases, the exploratory learning had a less than inspiring outcome; neither individual mastered the calculator.

When learners notice some feature, they seek an explanation for it. They can become frustrated and/or confused if no explanation is obvious, as L5 indicated in her reaction to an inscription she found in the border of a document window, "4692 blocks free out of 9690". As she said, "It makes me feel that I did something wrong." Worse perhaps, they may concoct an inappropriate explanation for what they believe they have observed. L2 got an error panel message "Before you type make a selection". The error panel contained a button labelled

Cancel. L2 reasoned that this was the selection referred to, and he clicked the Cancel: "I guess they were trying to save me from an error I was about to make, by typing Cancel I averted it." Cancel was not the selection referred to in the error panel, and his Cancel action did nothing more than dismiss the error panel. Interface designs need to call the user's attention to key features and to make these features simple enough that the user reliably tumbles to the correct interpretation of them.

What about errors? If rote learning approaches really worked, learners would never make errors in training. To its credit, the on-line tutorial we studied acknowledged the pervasiveness of errors and employed error shielding techniques to reduce the potentially distracting and frustrating consequences of errors. Nevertheless, our study exposed problems with the particular approach to error shielding that the tutorial employed. The approach seemed too rigid; one small error could spoil an entire effort on an exercise. Errors were sometimes so over-shielded that learners failed to detect them and therefore could not learn from them. For example, experimenting with conceptual distinctions between objects ("icon" versus "window") and actions ("click" versus "double click") can be fruitless when incorrect possibilities evoke no system response. Finally, when the tutorial did not shield errors, it often provided too little feedback for successful error diagnosis and recovery.

A better approach to error shielding might be to diagnose every learner action as either an error or a correct performance. This increases flexibility, for at worst, making an error necessitates repeating a single action (not an entire exercise). Feedback should accompany each error (whether shielded or not) so that the learner can interpret the experience. We developed and studied this approach with a "training wheels" system in which error shielding was employed to help learners keep on track in the course of learning by doing (Carroll and Carrithers, 1984a; see also Carroll, 1984).

The future. On-line training is not just a better medium than self-study manuals for presenting tutorials. It is a different medium entirely. It offers an opportunity to break with self-study approaches; to radically move toward an active learning-by-doing approach to user training. We feel that the system we studied did not go far enough in this regard, and that this is indeed typical of current approaches to on-line training. The challenge in this area is not to migrate existing approaches to user training into a new medium, but to discover the unique strengths of the new medium and the uniquely appropriate techniques for exploiting them in the design of user training.

Acknowledgments

We are grateful to Robert Gordon, Don Hatfield, Clayton Lewis, Mark Martin, and Mary Beth Rosson for comments on an earlier version of this report.

References

Carroll, J.M. Designing MINIMALIST training materials. *Datamation*, 1984 in press.

Carroll, J.M. and Carrithers, C. Training wheels in a user interface. *ACM Communications*, 1984a in press.

Carroll, J.M. and Carrithers, C. Blocking learner error states in a training wheels system. *Human Factors*, 1984b in press.

Carroll, J.M., Ford, J.R., Smith-Kerker, P.A., and Mazur, S.A. Learning to use a word processor with the Minimal Manual. Psychonomics Society, 1984.

Carroll, J.M. and Mack, R.L. Learning to use a word processor: By doing, by thinking, and by knowing. In J.C. Thomas and M. Schneider (eds.), *Human factors of computing systems*. Norwood, NJ: Ablex, 1984.

Carroll, J.M., Mack, R.L., Lewis, C.H., Grischkowsky, N.L., and Robertson, S.R. Learning to use a word processor by guided exploration. *IBM Research Report*, 1984.

Mack, R.L., Lewis, C.H., and Carroll, J.M. Learning to use word processors: Problems and prospects. *ACM Transactions on Office Information Systems*, 1983, *1/3*, 254-271.

Seybold Report on Word Processing, 4/1, 1981.

Seybold Report on Office Systems, 7/3, 1984.

END

FILMED

3-85

DTIC