END-USER TRAINING: AN EMPIRICAL STUDY COMPARING ON-LINE PRACTICE METHODS

Susan Wiedenbeck, Patti L. Zila, and Daniel S. McConnell

Computer Science and Engineering Department
University of Nebraska, Lincoln, NE 68588, U.S.A.
phone: (402) 472-5006
susan@cse.unl.edu

ABSTRACT

An empirical study was carried out comparing three kinds of hands-on practice in training users of a software package: exercises, guided-exploration, and a combination of exercises and guided-exploration. Moderate to high experience computer users were trained. Subjects who were trained with exercises or the combined approach did significantly better in both time and errors than those trained using guided-exploration. There were no significant differences between the exercise and the combined approach groups. Thus, it appears that the better performance of these groups can be attributed to the exercise component of their practice.

KEYWORDS: training, practice methods, exercises, guided-exploration, minimal manual, end-users, tutorials.

INTRODUCTION

We are interested in refining the understanding of the training needs of users of software packages. Since the mid-1980s, Carroll and his associates have documented problems which users face in computer training [5, 6, 7]. Their work indicates that learners of software packages experience numerous problems in the initial learning of software systems, including voluminous materials, lack of focus on real users' tasks, absence of error recovery information, and misleading analogies drawn from non-computer experience. Carroll and his colleagues have proposed a solution which addresses many of the training problems listed above, which is referred to as minimalist training [4, 7]. Minimalist training has been tested with positive results, as outlined below. However, there is a need for further studies which focus on individual parts of their larger strategy and attempt to determine in detail the role that they play. Here we investigate the role of different kinds of hands-on computer practice within the paradigm of minimalist training. We evaluate three kinds of practice which differ in the amount of structure provided to the learner.

PREVIOUS RESEARCH AND RESEARCH QUESTIONS

The minimalist training model has the following characteristics [7]: focus on real tasks, reduction in the verbiage of training materials, and support for error recovery and recognition. These features are meant to maintain high motivation, promote active learning, and make the environment "safe" for learners in the sense that they can try features out without fear of becoming hopelessly mired in errors.

Carroll, et al.'s experimental studies [6, 7] support the effectiveness of the minimalist training approach compared to commercial tutorial materials. A recent replication by Lazonder and van der Meij [12] supported their findings. Black, Carroll, and McGuigan [2] tried to isolate some of the many dimensions on which the minimal manual differed from traditional tutorial manuals and found two significant features: the shorter length in itself and the incompleteness of the materials, which encouraged inferencing by the trainee. Gong and Elkerton [11] showed that learning improved when subjects used a briefer manual with procedural instructions. However, in their study the effect of error recovery information was uncertain. On the other hand, both Olfman [14] and Davis and Bostrom [10] compared an exploration training approach to a more traditional instruction-based approach and found no significant differences.

An aspect of training that is closely related to the concept of minimalist training is the kind of hands-on practice provided in the training materials. Minimalist training advocates argue that people prefer active learning or learning by doing rather than by reading a manual. If that is the case, then hands-on practice may be essential for both learning and motivation in training. Carroll et al. [7] did not explicitly study practice methods, but it is clear that hands-on practice is essential to minimalist training. Charney and her colleagues [8, 9] studied practice methods and found that methods involving problem-solving practice were superior to merely reading or typing worked-out examples.

Among methods involving problem-solving, there are two major categories: exercise and exploration. Exercises leave the learner to decide on a solution strategy and the method for implementing it, and thus appear to be much less mechanical than typing the keystrokes of a worked-out example. Exercises provided by the manual designers can be structured to cover the full range of functions, illustrate important concepts, and correct misconceptions [9]. They may be particularly good for assuring coverage of the basic functionality of a system.

Exploration leaves the choice of the practice itself, as well as the solution strategy and method of implementing it, to the discretion of the learner. Allowing learners to set their own goals has been seen as a key to maintaining high learner motivation [7]. Another possible advantage of exploration is that it may facilitate meaningful learning. Assimilation theory [1] distinguishes between meaningful and rote learning. In meaningful learning the learner actively manipulates new information to connect it conceptually to prior knowledge, leading to a deep understanding of the new information and the ability to apply it in novel ways. Rote learning is memorization with little concern for its connection to prior knowledge. While rote learning may lead to success in tasks which require the repetition of known information, it is not likely to lead to success in novel tasks. Exploration-based training may promote meaningful learning by encouraging the learner to set goals which go beyond the simple procedures in the training manual. However, a possible problem with exploration training is that practice devised by learners may not be optimal for covering the capabilities and difficulties of the software.

In a previous empirical comparisons of exercise and exploration practice methods, Sebrechts and Marsh [16], studying UNIX novices, found that the exercise group performed better than the exploration group. It should be noted that their exploration practice was completely open-ended and occurred when the subjects had finished reading the manual. For half the subjects the manual was not even available during the practice.

As one part of a larger study, we carried out an empirical comparison of three kinds of hands-on practice: exercise, guided-exploration, and the combination of exercises plus guided-exploration. Unlike, Carroll et al. [6, 7], the content of our training manuals was the same across all conditions, and only the practice instructions differed. The exercise practice posed specific problems for the learner to solve. The guided-exploration practice asked learners to set their own goals and create their own problems to meet the goals. However, in doing so they were guided by suggestions and questions, which focused their attention on a set of possible goals appropriate to the software. We use the term guided-exploration to distinguish this kind of exploration from completely open-ended exploration. The combined practice consisted of an exercise followed by guided-exploration instructions. Below are the research questions which we addressed:

— **Does guided-exploration lead to more successful training outcomes than exercises?** Charney and Reder [8] did not include exploration practice in their study. Sebrechts and Marsh [16] included completely open-ended exploration practice and found that exercises led to better performance than exploration. We hypothesized that a more focused exploration-based practice would improve the performance of exploration trainees. As a result, we designed our guided-exploration practice to be a little less open-ended. We asked learners to practice at specific points as they worked through their manual and asked some focusing questions about the topic under discussion at that point. This was intended to guide them to choose appropriate goals, while still leaving them to choose the specific goals and the amount of practice. We hoped that giving a bit more guidance in the exploration would allow us to detect the advantages of exploration if they exist.

— **Is the combination of exercises followed by guided-exploration better than guided-exploration alone?** It has been suggested [16] that a possible reason for poor outcomes of exploration-based practice is that the learner does not know enough about the software to devise adequate hands-on practice. A remedy is to first give learners practice with exercises, then have them explore further. Our combined practice was designed to allow us to judge whether guided-exploration is more effective when it follows the more structured practice embodied in exercises.

— **Does goal-setting during practice appear to aid in learning software?** Carroll et al. [6, 7] see goal-setting by learners as fundamental in computer training. The essential difference between exercises and guided-exploration is that learners set their own goals in guided-exploration. The comparison of guided-exploration to exercises will give us insights about the role of goal-setting.

— **Which practice method aids meaningful learning?** Past research [3, 10, 13, 17] has used near and far-transfer tasks to operationalize the difference between rote and meaningful learning. Near-transfer tasks are very similar to tasks taught in the training. Far-transfer tasks require the learner to go beyond what was explicitly taught. This could involve combining several operations in a novel way or using tools and operations not described in the training, but suggested by analogy to others that were taught. Singley and Anderson [18] discuss procedural-to-procedural transfer as the transfer that occurs when productions acquired in training apply directly to a transfer task. We see our near-transfer tasks as being essentially of this type. Singley and Anderson [18] define declarative-to-procedural transfer as occurring when declarative knowledge acquired during training aids the acquisition of productions during transfer. We would classify our far-transfer as largely declarative-to-procedural. If exercises aid in the acquisition of productions for basic operations during training, they may aid near-transfer. Exploration practice, on the other hand, may aid far-transfer by encouraging learners to create problems that go beyond the training materials, leading to discovery of new concepts or novel combinations of simple concepts. A training method combining exercises and exploration could facilitate both near and far-transfer. In their study, Charney and Reder [8] evaluated the learning of easy and difficult commands, but apparently neither they nor Sebrechts and Marsh [16] tested subjects with tasks which required using commands in a novel way.

METHODOLOGY

Subjects

A total of 51 subjects participated in the experiment. Subjects were volunteers recruited among upper-level undergraduate and graduate students and were randomly assigned to the three training conditions. Seventeen subjects served in each training condition.

The subjects were recruited to represent moderate to high general computer experience. This group was chosen because in today's business environment most information workers routinely use one or more software packages in their work. Thus, learners of a package usually are experienced with other software. We wanted to represent this reality in our study. The mean age of the subjects was 28 years. Eighty percent were male and 20 percent female; however, a preliminary analysis of the data using an Analysis of Covariance showed that sex was not a significant covariate of performance. The subjects came from a wide variety of fields, but the majority were from technical areas, such as business and engineering. Many of the subjects had programming experience, ranging from introductory programming courses to an undergraduate degree in computer science. All of the subjects had considerable experience with applications programs on personal computers. The most widespread experience was with word processors and spreadsheets, followed by graphics and communications programs.

Materials

In this experiment subjects were trained on the Hypercard™ software program. Hypercard is a hybrid program which contains an integrated set of text and graphics tools along with an end-user programming language, Hypertalk™, which can be used to create advanced applications. Training was given on the basic text and graphics features of Hypercard. The Hypertalk programming language was not taught.

A self-study training manual was created for the experiment. Text-based learning materials were chosen because of evidence of better retention and transfer in text-based materials as opposed to alternate media [15]. Like Carroll et al.'s manual [7], our manual was brief, consisting of approximately twelve pages of text. It had a single introductory paragraph which motivated the use of Hypercard by mentioning several typical applications: address book, calendar, etc. Descriptions of commands and procedures were grouped into topic areas named to match the tasks of new users. Information was included to help the reader coordinate reading with what was appearing on the screen. The user was left to infer procedures by analogy as much as possible. Explicit error recognition and recovery information was included in the manual.

At various points in the manual practice opportunities were given. In all there were 14 practice opportunties. Three different versions of each practice were developed: exercise, guided-exploration, and combined. These three practice types were embedded in the manual to create three different versions of the manual with identical text, but differing in the statement of the hands-on practice.

Exercises gave specific tasks for the learner to carry out. They told what object to operate on and what to do to the object. Exercises were stated with the intention that it be easy for learners to evaluate their success. Guided-exploration asked the learner to create tasks for themselves in order to try out the procedures and commands detailed in each section of the manual. Thus, the learner was encouraged to practice the same procedures as the exercise subjects, but the specific object on which to work was not specified, nor was the specific result desired (where the object should be, what it should look like), nor how much to practice. In the guided-exploration practice instructions, focusing questions were posed to draw the learner's attention to important aspects of the interaction. Combined practice consisted of an exercise followed by a guided-exploration instruction. Figure 1 gives examples of the three kinds of practice instructions.

A set of evaluation tasks was also developed to use in measuring the subjects' performance after training. There were fourteen evaluation tasks in total. Nine of the tasks were near-transfer tasks, consisting of commands and procedures which were covered in the training manual. An example of a near-transfer task is the following:

> "On Card 2 create a field called Name. The field should be able to hold 2 lines of text and should be approximately 4" wide. Position this field towards the top and in the center of Card 2."

This was classified as near-transfer because creating fields, naming them, and physically manipulating

This practice was given following the manual section that discussed using the background. The manual section on the background consisted of two pages containing an explanation of the background, procedural instructions for using it, and related error recognition and recovery information. This section occurred near the end of the training, when the subjects had already created a stack of cards and put information on them.

Exercise

Go to the background. Create a button on the background and call it "Next". Move it to the bottom of the screen. Leave the background and return to the card. The button should now appear on every card in the stack.

Guided-exploration

Think of information that is needed in your stack of cards. Can you think of any information which each card would have in common? With this in mind, place the common information on the background, including buttons, fields, and graphics. When you feel confident that you can successfully work in the background, move on to the next section.

Exercise + guided-exploration

Go to the background. Create a button on the background and call it "Next". Move it to the bottom of the screen. Leave the background and return to the card. The button should now appear on every card in the stack.

On your own
Think of other information that is needed in your stack of cards. Can you think of any information which each card would have in common? With this in mind, place the common information on the background, including buttons, fields, and graphics. When you feel confident that you can successfully work in the background, move on to the next section.

Figure 1: Examples of three kinds of practice

them were topics described in the manual. Exercise and combined condition subjects were given exercises to practice most of these elements, although never in a form identical to the evaluation task. Since guided-exploration subjects chose their own practice, it is not certain which elements they practiced, but they did read the same descriptive text as the exercise and combined condition subjects.

In addition, there were five far-transfer tasks which required the subject to go beyond procedures explicitly described in the manual. Far-transfer was operationally defined as one of three things: using a tool that had not been taught in the manual, doing an operation taught in the manual in some different context from the original, or combining a series of separate operations in some novel way to achieve a goal. An example of a far-transfer task is the following:

> "Place a copy of the portrait on Card 1 onto Card 2. Place the copy to the left of the field Name."

In the manual the subjects were taught how to copy text. They were also taught how to select and move regular-shaped graphic objects, such as rectangles. However, the manual did not give instructions about copying an object such as the portrait mentioned in this exercise. The subject first had to recognize that the rather elaborate portrait was a graphic object, just like a rectangle, then conclude that it could be moved by a copy/paste operation as is text, then find a way to select an irregular-shaped graphic. Thus, this task was far-transfer because the subject had to infer how to do it by combining what they knew about copying text with their knowledge of manipulating graphics. In the task list the near and far-transfer tasks were intermixed.

Procedure

Subjects were run individually. The average time to complete a session was 2 hours, but some subjects took up to 2 1/2 hours depending on the amount of time they spent practicing and doing the evaluation tasks. First, the subject completed a questionnaire detailing his or her computer experience. Then the subject worked through the manual independently, carrying out the hands-on practice when instructed by the manual. Subjects were asked to work through the manual from beginning to end but were allowed to go back to earlier sections whenever they wished. Feedback was not given on the subject's work in order to simulate a real self-study environment where normally the only feedback is from the computer. Subjects were given a maximum of 90 minutes for training with the manual. If they finished working through the manual before the training period was over, they were allowed to continue their training, if they wished, in one of two ways, depending on their experimental condition. Guided-exploration and combined subjects were allowed to continue exploring

Hypercard on their own, reading and practicing on-line as they wished. Exercise subjects, on the other hand, were restricted to prevent unstructured exploration of the system. They were allowed to reread the manual, but the only practice they were allowed was to repeat exercises given in the manual. An experimenter was always present to monitor that subjects followed the instructions for their practice condition. The experimenter also kept detailed notes about the subjects' specific activities and their duration during the training phase. The training was stopped either after 90 minutes or when subjects indicated that they were finished. The total time spent on hands-on practice during training was recorded.

The evaluation phase of the experiment was conducted after a break, during which subjects filled out several questionnaires not reported on here. Subjects were given the set of 14 evaluation tasks to carry out. The manual was not available for use, but subjects were given a one page summary of the menu items and procedures taught in the manual to aid their memories. The tasks were timed and the subjects' work was saved. The evaluation phase lasted a maximum of 55 minutes, but subjects could stop sooner if they were finished or could make no further progress.

RESULTS

One subject completed a first pass through the manual but did not have time to go back to earlier sections for rereading and further practice. All other subjects had time to return to earlier sections of the manual. All but two subjects requested to go on to the evaluation tasks before the 90 minute training period was up. Thus, it appears that subjects had sufficient training time. Table 1 summarizes the means and standard deviations of the dependent variables, training time, near transfer time, far transfer time, near transfer correctness, and far transfer correctness for each practice condition. The following paragraphs describe the results of the multivariate and univariate analyses.

A MANOVA including all the dependent variables was run. The MANOVA showed that the effect of practice type was significant ($F(10, 90) = 5.62$, $p < .05$), showing an overall difference in performance among the practice types. After the significant MANOVA, ANOVAs were run for the individual time and correctness variables.

Training time consisted of the total time that subjects spent in hands-on practice during the training period, excluding the time spent reading the manual, but including time for looking up information in the manual during practice. Near-transfer time was the sum of the times spent doing the 9 near-transfer tasks. Far-transfer time was the sum of times for the 5 far-transfer tasks. The ANOVA for training time showed that there was a main effect of practice type ($F(2,48) = 7.82$, $p < .05$). Newman-Keul's test for specific differences was run and showed that the exercise condition was faster than the guided-exploration or the combined condition ($p < .05$), but the guided-exploration and the combined condition did not differ significantly from each other. The advantage of the exercise condition was about 12 minutes over the combined condition and 17 minutes over guided-exploration. The ANOVA for near-transfer time revealed that there was a main effect of practice type ($F(2,48) = 11.11$, $p < .05$). Newman-Keul's test indicated that the guided-exploration subjects were significantly slower than the exercises or combined subjects on near-transfer tasks ($p < .05$). The ANOVA for far-transfer time also showed a significant difference based on practice type ($F(2,48) = 7.88$, $p < .05$). The exercise and combined conditions were significantly faster than guided-exploration (Newman-Keul's test, $p < .05$). The exercise and combined conditions did not differ significantly from each other.

Variable	Exercise	Guided Exploration	Combined
Training time (minutes)	31.25 (11.71)	48.11 (11.43)	43.44 (15.05)
Near-transfer time (minutes)	10.24 (3.45)	17.35 (5.92)	10.44 (5.31)
Far-transfer time (minutes)	7.28 (3.56)	11.26 (4.29)	6.46 (3.40)
Near-transfer correctness (maximum = 27)	24.71 (1.65)	18.59 (4.69)	22.65 (4.57)
Far-transfer correctness (maximum = 15)	14.24 (1.20)	7.82 (3.80)	12.18 (3.19)

Table 1. Mean and standard deviation (in parentheses) of dependent variables by practice type

The evaluation tasks were graded for correctness on a scale of 0 to 3 as follows:

3 = task completely correct
2 = task mostly correct (over 50% correct)
1 = task mostly incorrect (50% or less correct)
0 = task completely incorrect or not attempted.

The work of the subjects was graded independently by two judges using a set of detailed grading criteria developed in advance. The score assigned to each task was the average of the grades of the two judges. The inter-rater reliability was .97. For the analysis, the sum of scores for all near-transfer tasks was used as the subject's near-transfer correctness score, and the sum of the scores for all far-transfer tasks was used as the far-transfer correctness score. The ANOVA for near-transfer correctness indicated a main effect of practice type ($F(2,48) = 11.08$, $p < .05$). Newman-Keul's test showed that the exercise and combined conditions had higher correctness on near-transfer tasks than the guided-exploration condition ($p < .05$). However, the exercise and combined groups did not differ significantly from each other. The ANOVA for far-transfer correctness also showed a significant difference based on practice type ($F(2,48) = 22.68$, $p < .05$). Newman-Keul's test revealed that the guided-exploration group performed more poorly than the other groups ($p < .05$), but exercise and combined groups did not differ from each other.

DISCUSSION

The results regarding the time spent on hands-on practice during training reflect differences among the three practice methods. The combined method took longer than exercises because there was more for the learner to do. We can reasonably assume that the extra time represents time that learners spent on exploration. Since the combined group spent almost 40 percent more time, the exploration component of their practice was substantial. The exploration group also spent significantly more time on practice than did the exercise group. There are several possible explanations of their additional time. Part of the time can probably be attributed to goal-setting, since these subjects were responsible for choosing their own practice. Another explanation could be that guided-exploration subjects tried out more functions or more advanced functions. However, if this were the case, we would have expected guided-exploration subjects to perform better than exercise subjects on far-transfer tasks, something which we did not see. A third possibility is that exploration subjects made more errors in training and spent more time in error recovery. Unfortunately, we do not have the data to evaluate this explanation. It should be noted that our result that training was slower for the guided-exploration group does not contradict the results of Lazonder and van der Meij [12]. They found faster overall training time for subjects using a minimal exploration manual, but they were comparing a minimal manual to a much longer standard tutorial manual, and they were focusing on the total training time, not just on the difference in time spent doing hands-on practice.

Our evaluation task results showed that exercise and combined practice tended to be superior to guided-exploration practice. Sebrechts and Marsh [16] had found that performance was poorer with completely open-ended exploration practice than with exercises. As a result, we made our guided-exploration moderately more structured to aid the learner in setting appropriate goals. In particular, we asked them to practice at given points in the manual, and we posed questions to them to focus their goal-setting and help them notice certain critical aspects of the interaction. Even so, exploration practice still led to poorer results. Since exercise subjects trained faster than guided-exploration subjects, the training time does not explain these results.

Performance in the combined condition was very similar to the exercise condition. This implies that the exercise practice was the essential factor leading to successful hands-on training. We had speculated that combined practice might be optimal because the exercise would give subjects a basic understanding of each concept, and further exploration would then allow them to expand their understanding. However, this expectation was not supported. We had also speculated that practice methods with an exploration component would lead to success on far-transfer tasks because they would encourage meaningful learning. Instead, we found that for far-transfer performance of subjects in the exercise condition equaled that of subjects in the combined condition and exceeded that of subjects in the guided-exploration condition. From this it appears that exercise practice allows the trainee to work with the material sufficiently to later apply it in novel tasks. Guided-exploration practice, with its goal-setting component, was not an aid to far-transfer.

Why did guided-exploration practice lead to unexpectedly poor results? Other researchers [9, 16] have speculated that novice learners have difficulty creating adequate practice on their own. Based on our results, we believe that the inability to formulate adequate practice may also apply to more experienced learners. It seems most likely to occur when learners are working with software packages dissimilar to those they already know. We believe that this was the case in our experiment. While most of the subjects had experience with both text processing and graphics software, they did not have experience with text and graphics in the integrated combination of Hypercard. We dealt with this in the exercise and combined conditions by providing a set of exercises that integrated text and graphics to build a very rudimentary address book-type of application. The guided-exploration practice instructions continually asked subjects to think of realistic information that they could group using the features of Hypercard, but this apparently was not successful in encouraging subjects to create a realistic application. In fact, our observations recorded during training showed that most exploration subjects (13 out of 17) created a series of practice problems that were discrete and unrelated to one another. Thus, they may not have gained a sense of how Hypercard could actually be used. This may have hurt their performance on far-transfer tasks, in which they needed to put separate pieces of knowledge together in novel combinations. Our observations during training also suggested that guided-exploration subjects did not practice basic skills as thoroughly as the exercise and combined subjects. They tended to skip practicing some functions described in the manual and to minimally practice others. This may have hurt them on near-transfer tasks. For example, the exercise given for putting text in a field required the subject to type the text, erase part of it, change the font, and change the style. Guided-exploration subjects usually did not practice all of these basic skills, even though they were all described in the manual. Also, guided-exploration subjects tended to be attracted to and spend proportionately more time on creating and manipulating graphics than fields or buttons. This is a poor distribution of effort in terms of Hypercard functionality. Buttons and fields are the basic elements used in information storage and manipulation, while graphics affect the look of a Hypercard application but not its functionality. Exercise and combined subjects tended to practice the basics because the set of exercises stressed them. Thus, the goal-setting of the guided-exploration subjects did not aid learning.

CONCLUSION

Based on these results, we recommend the inclusion of well-designed exercises in training materials. We see two main advantages to exercise practice. First, exercises can assure that the learner is exposed to the software's most important functionality. Second, a set of exercises can suggest typical uses for which the software is well-suited, thus helping users to see how it could be used to advantage in their own work. Of course, these advantages can only be realized in exercises which are carefully designed and tested. The requirement of careful design and testing applies not just to individual exercises but to the exercise set as a whole. A series of exercises must be well-integrated if it is to effectively suggest appropriate uses of the software.

We do not see our results as a challenge to the minimalist training model. In fact, our experiment was firmly situated in the minimalist tradition by our minimal manual which promoted active, hands-on learning and inferencing on the part of the user. Charney and Reder's results [8] supported the importance of problem-solving practice. This work suggests that it is important that the problem-solving practice be well-conceived. Even relatively high experience learners, such as ours, may experience difficulties in devising useful practice on their own, and this seems particularly likely when the software domain is unfamiliar. Carroll et al. [6, 7] argue that it is important for learners to set their own goals in order to maintain motivation during training. We agree that doing real work is highly motivating. However, many times learners do not begin with real work to accomplish which can only be done using the new software. Rather they begin with questions about how the software can benefit them in their own work. In this situation, the user's goal is less immediate and concrete. The user wants to learn what the software offers and how to carry out typical operations. For these learners, the structuring inherent in a well-conceived set of exercises appears to be optimum. It should be noted that exercises in a self-study manual are only suggestions. Users whose goal is to carry out a specific piece of work should still be able to use a minimal style manual to advantage for that purpose.

ACKNOWLEDGMENTS.

This work was supported by NSF grant CDA-9200230 and by funds from the University of Nebraska. It was completed while the first author was a visitor at the INRIA. The authors would like to thank Jack Kant for his work in developing the training materials.

REFERENCES

1. Ausubel, D. P. Educational psychology: A cognitive view. Holt, Reinhart and Winston, NY, 1968.

2. Black, J.B., Carroll, J.M., and McGuigan, S.M. What Kind of Minimal Instruction Manual Is the Most Effective, in Proc. CHI+GI 1987 Human Factors in Computing Systems and Graphics Interface (Toronto, April 5-9, 1987), ACM Press, pp. 159-162.

3. Borgman, C.L. The user's mental model of an information retrieval system: an experiment on a prototype on-line catalog. International Journal of Man-Machine Studies, 24 (1986), 47-64.

4. Carroll, J.M. The Nurnberg Funnel: Designing Minimalist Instruction for Practical Computer Skill. MIT Press, Cambridge, MA, 1990.

5. Carroll, J.M. and Mack, R.L. Learning to Use a Word Processor: By Doing, By Thinking, and By Knowing. In Human Factors in Computer Systems, J.C. Thomas and M.L. Schneider, Eds., Ablex, Norwood, NJ, pp. 13-51.

6. Carroll, J.M., Mack, R.L., Lewis, C.L., Grischkowski, N.L., and Robertson, S.R. Exploring exploring a word processor. Human-Computer Interaction, 1, 3 (1985), 283-307.

7. Carroll, J.R., Smith-Kerker, P.L., Ford, J.R., and Mazur-Rimetz, S.A. The minimal manual. Human-Computer Interaction, 3, 2 (1987), 123-153.

8. Charney, D.H. and Reder, L.M. Designing interactive tutorials for computer users. Human-Computer Interaction, 2 (1986), 297-317.

9. Charney, D.H., Reder, L.M., and Wells, G.W. Studies of elaboration of instructional texts. In Effective Documentation: What Have We Learned From Research, S. Doheny-Farina, Ed., MIT Press, Cambridge, MA, 1988, pp. 47-72.

10. Davis, S.A. and Bostrom, R.P. Training end users: an experimental investigation of the roles of the computer interface and training methods. MIS Quarterly, 17 (1993), 61-85.

11. Gong, R. and Elkerton, J. Designing Minimal Documentation Using a GOMS Model: A Usability Evaluation of an Engineering Approach, in Proc. CHI'90 (Seattle, April 1-5, 1990), ACM Press, pp. 99-106.

12. Lazonder, A.W. and van der Meij, H. The minimal manual: is less really more? International Journal of Man-Machine Studies, 39 (1993), 729-752.

13. Mayer, R.E. The psychology of how novices learn computer programming. Computing Surveys, 13 (1981), 121-141

14. Olfman, L.A. A Comparison of Applications-Based and Construct-Based Training Methods for DSS Generator Software, unpublished doctoral dissertation, Indiana University, Bloomington, IN, 1987.

15. Palmiter, S. and Elkerton, J. An Evaluation of Animated Demonstrations for Learning Computer-Based Tasks, in Proc. CHI'91 (New Orleans, April 27-May 2, 1991), ACM Press, pp. 257-263.

16. Sebrechts, M.M. and Marsh, R.L. Components of computer skill acquisition; some reservations about mental models and discovery learning. In Designing and Using Human-Computer Interfaces and Knowledge-Based Systems, G. Salvendy and M. J. Smith, Eds., Elsevier Science Publishers B. V., Amsterdam, 1989, pp. 168-173.

17. Sein, M.K. and Bostrom, R.P. Individual differences and conceptual models in training novice users. Human-Computer Interaction, 4 (1989), 197-229.

18. Singley, M.K. and Anderson, J.R. The Transfer of Cognitive Skill. Harvard University Press, Cambridge, MA, 1989.