TARGETING COMPUTER TRAINING: IMPROVING LEARNING BY MORE EFFECTIVELY ADAPTING TRAINING TO MEET USER NEEDS

Jeffrey Hsu
Information Systems
Silberman College of Business Administration
Fairleigh Dickinson University
jeff@fdu.edu

Murray Turoff
Information Systems
College of Computing
New Jersey Institute of Technology

Abstract

Training is an important means by which corporate staff and management can learn computing skills and gain knowledge in new technologies. However, despite the myriad of training methods and options which are available, frequently training does not achieve its desired result. While some of this may be due to individual differences or the ability for a particular subject to learn, the type of training/help can also play an important part. The role of three different forms of training (offline, online, visual online) in promoting or hindering learning was the focus of this empirical study, and the results varied by skill level: novices learned best with the offline manual, and worst with the visual online. Experienced users, on the other hand, learned best with online training and the worst with visual online. Also of significance is the finding that while experienced users have reasonably good perceptions of what kinds of training best contribute to their own learning, novices are largely erroneous in their perceptions. This may cause novices to choose training methods which are minimally useful, or possibly detrimental, in their quest to learn a new programming language. By better targeting the right form of training to a certain level of user, the effectiveness of training can increase, resulting in higher levels of productivity and efficiency.

Keywords: Training, human-computer interaction, markup languages, minimal manual, help

Introduction

Computer-related expertise and experience remain one of the most important and desired skill areas for organizations wanting to compete in our information-age society. However, whether it be doing computer language coding, working out problems on a spreadsheet, or using a firms' new information system, many people still struggle with using computing systems and software. One approach to improving the computer knowledge of staff and management in an organization has been to provide various forms of training (manuals, classroom, online, video, etc.) to those who need or desire it. In some cases, the training proves beneficial, however, in many cases, even after having undergone the training, the results turn out less than ideal, and more efficient computerized systems remain ignored or underused in favor of slower, manual methods. This results in wasted training expenses, lost productivity and time, and frustration on the part of the learner.

The inability for many existing training materials and methods to accomplish its main goal—to help potential users learn a new task or skill—is one of the most critical problems in the entire field of training. Some of this may be due to the problems that many professionals (Argyris, 1991) and also adults, particularly older adults, encounter in attempting to learn new skills (Prewitt, 1997), while another critical reason might be that the training itself is not designed to meet the needs of the learner. Some of the specific reasons given for these failures include the problems of learners recalling command syntax (Borgman, 1986; Michard, 1982),difficulties using complex training materials (Carroll, 1984), and problems with using applications software to accomplish specific tasks (Carroll and Mazur, 1986).

The entire area of training potential users of a system or language is often referred to end-user training (EUT), which is part of what is known as end-user computing (EUC). There is an entire body of literature which examines the issues, problems, and effectiveness of end-user training. In fact, end-user training has been identified as one of the key factors responsible for enabling the success of end-user computing (Davis and Bostrom, 1993). In addition, both efficiency and effectiveness can be enhanced when a strategy of both basic and advanced training is implemented in an organization (White and Christy, 1987; Rivard and Huff, 1988).

This paper describes the background, results, and summary/conclusions for a research experiment which examines learning and perceived learning as it relates to the learning of a survey questionnaire markup language. The language used, which has specific markup tag types designed for the creation of questionnaire question types, was an enhancement to markup language standards such as HTML and SGML (Hsu and Turoff, 1996).

Learning a markup language, like any new computer language, can often be a trying and difficult activity. Whether it be the complicated syntax, the myriad of commands, or unintuitive ways of accomplishing a task, the task of learning programming is often regarded as being laborious and aggravating. While subjective opinions and reactions to using/learning languages and software can vary widely from subject to subject, there are a number of key factors which can influence a subject's learning of a computer system. One of these factors is the type of training/help which can be provided to the learner. There are many different forms and variations of training/help, with varying results in terms of performance, effectiveness, and acceptance by users (Shneiderman, 1997; St. Laurent, 1997).

User background is another important factor which was examined in this experiment. The previous programming background of the subjects is an important factor which should be examined when it comes to evaluating help systems and learning. In general, experienced users tended to have a better "mental model" of a system or a language. There are a number of ways to obtain this kind of mental model--through usage, through analogy, and through training (Shih and Alessi, 1994). A person can learn how a system is put together by using it; it is a function of the user interface, knowledge of other, prior systems, and various individual traits. Analogies are also useful to help someone develop this kind of mental model, which can be heavily influenced by their use of similar systems or by previous experiences.

Payne and Green (1989) found that one of the main differences between novices and experienced users was found in the kinds of mental models which they have, with many novices having incorrect or ill-constructed models which lead to difficulty in actually conducting the task or writing the code. The use of help systems which use conceptual instruction methods will help to build effective mental models, which in turn will improve learning (Shih and Alessi, 1994). In short, the effect of help systems should be to enhance learning, both for novices and experienced users, but with a generally greater impact for novices.

The role of training methods in the learning and use of computer systems and software is an important one. There is constantly a need to provide assistance to users, however there has been a lot of research and attention given to the right kind and form of training. There are many factors which go into the development of an effective and productive training/help system.

A great deal of attention has been given to off-line training methods, or those kinds of help systems which are not computer-based. The primary form of off-line help is that of the manual, which often takes the form of thick volumes which accompany a system or a piece of software. A number of studies have concluded that very long and comprehensive manuals could actually have a negative effect, because they would overwhelm the reader with its technical emphasis and cause frustration on the part of the user.

Many of these manuals lead users through step-by-step tutorials and/or practice-drill exercises which are designed to teach the reader the material. However, it was found that many users preferred not to stick exclusively to these rote exercises, and instead followed a more "active" form of learning and problem solving (Mack, Lewis, and Carroll, 1983; Carroll & Mazur, 1986). One of the findings was that learners liked to work directly on practical and understandable tasks rather than just performing structured drill or practice sessions. This finding supports the assertion that learners preferred an "active" form of learning rather than a passive one . In addition, it is a logical follow-up to previous work which looked at the benefits of user-relevant goals and system-relevant topics, called guided exploration (Carroll et al., 1985). The guided exploration method reduced learning times, resulted in fewer errors, and showed better recovery from errors than using conventional training manual materials.

In summary, this research looks at the impact of training on learning of markup languages, with attention paid to the differences that these types of training have on both novice and experienced subjects.

Markup Languages

Markup languages have become increasingly important in the computing community, because of its wide use in both Internet web page design/development and electronic publishing. Simply put, a markup language consist of a set of tags, tokens, characters, or specialized command which are placed in a body of text in order to provide information about the text or other data being processed (Coombs et al., 1987; Goldfarb, 1991). Over the past several years, the importance of languages such as HTML (HyperText Markup Language), XML (eXtensible Markup Language), Standard Generalized Markup Language (SGML) , Dynamic HTML (DHTML) and others cannot be over-emphasized. HTML is widely used for creating hypertext-based documents which run on the World Wide Web (Darnell, 1997; Mullen, 1998). SGML is an international standard for describing marked-up electronic text (AAP, 1992; Wright, 1998). The definition of customized markup languages and tags for specific applications using XML is creating much attention and interest, for e-commerce and related applications (Holzner, 1997; Levelthal et al., 1998; Light et al., 1997; Wright, 1992).

Training

There were three different types of training which were developed and tested in this study. The types are as follows:

Offline Minimal Manual. A printed, hardcopy manual, which was a tutorial and reference guide prepared using Minimal Manual techniques. This used minimalist techniques proposed by Carroll. An important finding in the area of training and help systems concerns the concept that "less is more" (Scharer, 1983). In other words, briefer, more task-focused manuals which focused on learners' needs are preferred and worked better than standard manuals. Carroll et al. (1987) calls this the "minimal manual" and gave it certain specific characteristics:

(i) Focus is on real tasks and activities. Instead of providing practice examples and problems, the focus is on realistic tasks and on accomplishing a concrete goal. This might include creating a letter on a word processor rather than just learning how to input and edit text. This focus is designed to bring about greater satisfaction, meet the needs of learners, and provide the sense that something meaningful was accomplished (Carroll et al., 1987; Mack et al., 1983; Carroll and Mazur, 1986).

(ii) The amount of verbal material. Learners preferred more concise, shorter learning materials, which focus on tasks and on "action" rather than reading lengthy descriptive material. Aside from lengthy verbiage and descriptions, users did not find added control information to be very useful ("read this first," or "how to use this book") Instead, brevity and conciseness seemed to be most desired (Carroll et al., 1987).

(iii) Errors. Learners will make errors, and it is important to recognize this when creating help systems. An important factor is what to do to help a learner recognize errors and to allow him or her to recover from these errors. The ability to let learners know that an error has occurred, and also to allow them to properly recover from errors should be the goal of user documentation. In other words, a user should be taught to see when the error happened, and what to do when it happens.

In the research conducted by Carroll et al. (1987) which designs a minimal manual using a word processing application, it was found that the users of the Minimal Manual had learned faster and performed better than those who used conventional materials.

As stated before, the development of effective user manuals is an important task in developing computing systems. In order for a system to be effective it must be properly used in order to solve problems and do what the user has intended. However, there is often the problem that user manuals are not as effective as they should be. One problem is that users do not want to read them. Penrose and Seford (1988) found that only about 14% of users actually read reference/user manuals. Lazonder and van Meij (1993) and Redish (1988) found that an emphasis on tasks is important. Users like to read to "learn to do."

In an attempt to create better user manuals, there has been attention given to the concept of minimalism. In other words, "less is more." The concept of thick, comprehensive manuals which discuss all of a program's commands has been found to be less effective that those which are briefer, focus on tasks, and reduce the verbiage of manuals. The concept of the Minimal Manual has been shown to improve performance over standard reference manuals (Carroll, 1990; Carroll et al., 1987; Carroll, 1991; Lazonder and van der Meij, 1993). The basic concept of the minimal manual is that focuses on real tasks, reduces the verbiage of training materials, and supports error recovery and recognition.

Lazonder and van der Meij (1993) went further to describe a Minimal manual as one which followed the guidelines of: (a) task orientation (b) adequate use of text, (c)provision of information to detect and correct errors, and (d) modularity. These are supported by Carroll (1990), Hallgren (1992), and Lazonder and van der Meij (1992). Task orientation refers to the focus which the manual has--which is that the manual is directed towards the goals which the software is designed to meet. Both the headings and the text are directed towards the main thrust of the program--such as "editing or creating documents for a word processor", rather than "installation or configuration details," or specialized "expert" commands.

In terms of the adequate use of text, there should be little text as possible, which means that summaries, indexes, and prefaces are deleted. The focus of the manual is on accomplishing tasks rather than "concepts" or "broad general; descriptions." The text should also be written simply, without jargon and as few technical terms as possible.

Handling errors is another important concept for a minimal manual. The manual should be able to meet the needs of beginning users who are likely to make mistakes and should not only help them know when an error has occurred, but also to help them in knowing what to do when an error occurs. Error recovery information, for example, is one important component of an effective minimal manual.

Being modularized (self-contained sections) is another important concept for a minimal manual. There is little cross-referencing during chapters, and most, if not all of the chapters should be modularized in such a way that someone can go to the section of greatest practical use and interest to them.

Other findings concerning minimalism and minimal manuals centers upon the ways in which minimal manuals differ from standard reference manuals. These include a shorter length, and "incompleteness" which encouraged the user to spend time exploring the system (Black, Carroll, and McGuigan , 1987), and the brief length and procedural emphasis (Elkerton, 1990). Other studies including Ramsay and Oakley (1992), Van derlinden et al. (1988), and Raban (1988) point to the benefits of minimalism over standard instruction manuals.

Another important concept of user training is the concept that users want to be active rather than passive when learning a system. Carroll et al. (1985) made the point that many learners needed to set goals of their own during learning in order to keep their motivation during learning. The concept of "guided exploration" is in line with the "goal setting" concept, where learners are encouraged to go off and explore their system, and also in effect create their own objectives for learning. This can be contrasted to the approach which emphasizes the use of exercises to teach skills. The user is given a problem to solve, and then lets the learner find an appropriate solution and the right method for getting to the solution. Charney (1988) found that the use of exercises can teach a wide range of functions and also help users to learn important concepts. In studies contrasting these two methods, Sebrechts and Marsh (1989) found that those given exercises performed better than those who were asked to do guided exploration. In another study, by Wiedenbeck et al. (1995), there were three conditions: exercises, guided-exploration, and combined (exercise, then guided exploration). It was found that the combined approach brought about the best performance, and that the exclusive use of exercises was close in terms of performance results. The guided exploration group had the poorest results. The reason may be that the open-ended approach of the guided exploration did not focus the users towards the tasks to be learned as well as using exercises.

Another concept in regards to training users of computing systems is that of providing users with a kind of "mental model" of the system. There are a number of ways to provide this kind of mental model--through usage, through analogy, and through training (Sein and Bostrom,1989). A person can learn how a system is put together by using it; it is a function of the user interface, knowledge of other, prior systems, and various individual traits. Analogies are also useful to help someone develop this kind of mental model, which can be heavily influenced by their use of similar systems or by previous experiences. Finally, there can be a kind of conceptual model which is provided to the user, which can help him or her to learn the system.

In order to develop this mental model, one approach is to use an advance organizer (Ausubel 1960; 1963) which can provide the naive user with a knowledge of the relationships between the displays and components of the system. In other words, it explains the organization and structure of the system, in practical terms, to a novice user. Another method is to provide a global "prior to learning" display which is designed to bring about a more effective level of learning. Patrick and Fitzgibbon (1988) showed in their experiment that a structural display of the relationships (procedural and functional) helped to improve learning. It is especially effective when presented prior to the administration of the learning materials and actual work on the task.

Additional research conducted on the minimal manual since Carroll's original study have also pointed to better results (learning, performance) than using conventional reference manuals (Ramsay and Oatley, 1992; Olfman and Bostrom, 1988; Gong and Elkerton, 1990).

Additional work includes that by Lazonder and van der Meij (1993). They examined the work done since Carroll's original paper and not only replicated and extended the experiment, but also provided some additional guidelines, which in their opinion might improve the research. It was also noted in previous research that having too much information is not always a positive factor, and that a brief, concise format is preferred to a narrative style or an overly comprehensive coverage of the help material (Borenstein, 1985; Grimm et al., 1987). These all support the concept of minimalism, which emphasizes a smaller amount of text and less information overall.

The method by which help text is structured and presented is very important in terms of the overall effectiveness of the help system. This "quality" of the help text is indeed an indicator of help effectiveness. In fact, often lists were preferred to blocks of text (Borenstein, 1985). The level and amount of coverage of the markup structure ands commands manual was similar to that of the online help, however the structure of the information was presented in a printed form rather than using it through an online delivery format.

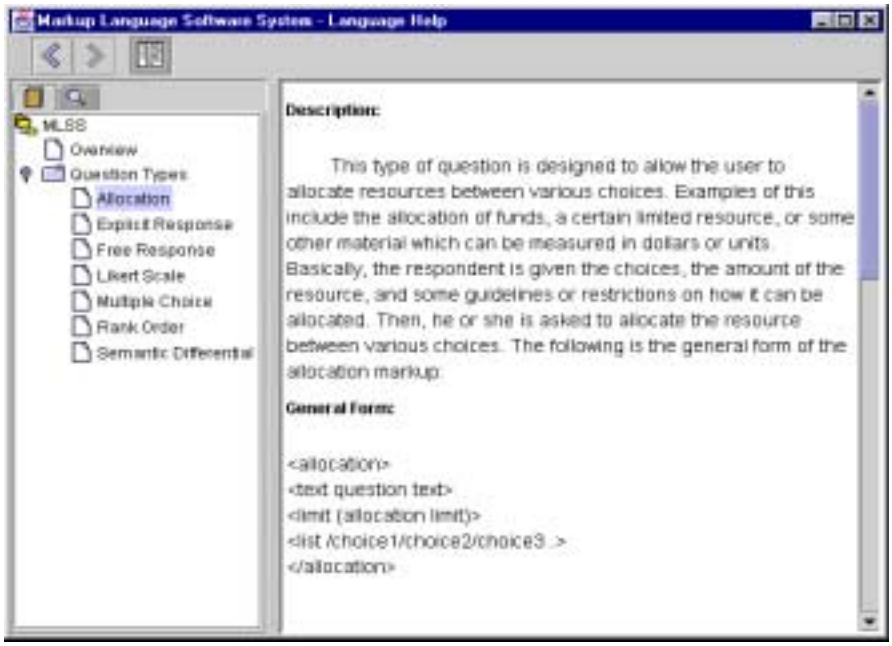

Figure 1. Standard Online Training

Standard Online Training. This form of training is of the traditional online format, with headings outlining major language concepts, examples, and descriptive explanations, all presented using an online help facility. Standard online training corresponds to the model of online help which offers reference-oriented information, including specifications on language syntax, examples of code, and examples of output generated. (Hsu and Turoff, 1996; Shneiderman, 1997, 1992). This training was designed to be used while online with the system. Standard online training was hypothesized to provide better learning for experienced users, who can benefit from the detailed information and having it available on the screen (see Figure 1).

Visual Online Training. This visually-structured training is designed to enhance learning through the visual identification of markup language code as it relates to the finished output. In effect, it includes a minimum of text, and instead features graphics depicting the code and the interpreted output. This is a more "experimental" form of training which emphasizes the relationship between the code and the generated question (see Figure 2). While not illustrated in the screen capture, the same level and kind of reference information (detail on commands and structure) is contained within the Visual Online as the Standard Online, however the examples are illustrated with visual/graphical examples rather than written examples. Only the graphical part of this training system method is shown, due to technical and space limitations.

Aside from the offline manual, which was written using Microsoft Word and then printed out in hard copy form, the other two versions of training were incorporated into the Java-based MLSS (Markup Language Survey System) which was the main tool used in the study. Versions of the software were built with manual only (Minimal Manual), standard Online training (Online) , or online visually structured training (Visual Online). As such, subjects were only allowed to access one form of training while completing the task.

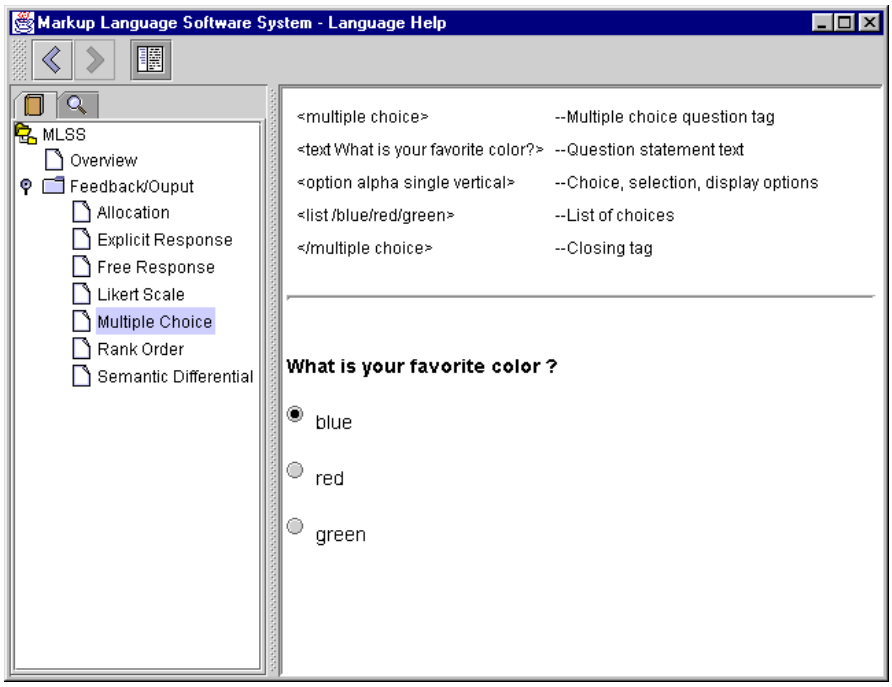

Figure 2. Visual Online Training
(excerpt from display section with example of output)

Materials

The following materials were used in this study:

- *Pre-Test Questionnaire*. Filled out by all subjects who participated. A key section of the questionnaire addressed previous background, and was used to determine skill level.

- *MLSS Software*. Each participant was provided with a customized copy of the experimental software, which was a custom-designed Java-based PC application which incorporated the following:
  - Editor for writing the markup code.
  - Interpreter which reads and creates output from the markup code.
  - Offline Manual (if applicable).
  - Online or Visual training system (if applicable).

- *Experimental Task* . The task entailed taking a printed 15-item survey questionnaire, and coding it using the markup language. The resulting code, if written correctly, should produce an online version of the questions to be displayed on the screen. The questionnaire used is an adapted (shortened and simplified) form of the EIES Virtual Classroom questionnaire, which had been used in previous research to tests users' reactions and responses to online Virtual Classroom courses and systems.

- *Post-Test Questionnaire*. This questionnaire included a quiz which required the subject to recall from memory the markup types, coding and syntax, and purpose for each markup type. The quiz was graded on a basis of 70 points, with 70 being a perfect score.

Subjects

A total of 268 undergraduate students participated in this experiment. The students were volunteers recruited among students majoring in computer science, engineering, architecture, and accounting. The students were given credit for participating in the study. Subjects were categorized as either a "novice" or an "experienced" subjects according to their previous background,

reflected in their responses to items in the Pre-Test Questionnaire, administered to the subjects prior to conducting the actual task. The subjects who participated in the study were primarily undergraduate students from both the New Jersey Institute of Technology and Queens College in New York. The number of subjects in each of the training conditions was approximately equal, and among these, very close to half were novices, while the other half were experienced users.

Procedure

This experiment, which examines the learning of markup languages as it relates to help systems and user experience level, is in effect a 3x2 factorial design, with 3 forms of training and 2 types of experience level. The procedure for this experiment was conducted in three main stages: Pre-Test, Test, and Post-Test.

The Pre-Test stage involved the distribution of Pre-Test Questionnaires to prospective subjects, with the intention of analyzing them and categorizing them according to skill level. The skill level was determined according to the following formula:

- *Experienced* were defined as those subjects who rated themselves as having EITHER:
  - Medium or high or very high programming experience
    OR
  - Medium or high or very high markup knowledge/experience.

- *Novices* were defined as those subjects who did not meet either of the above conditions.

The Task Stage required subjects to build an online questionnaire using the markup language, using the markup tags/code. In essence, the subject would build an "electronic" version of the paper questionnaire, which displays survey questions onto the screen after interpretation of the markup language code.

The Post-Test stage involved submittal of all task-related information, followed by the completion of a Post-Test questionnaire which included a quiz and items focused on subjects' perceptions of learning.

The scores for the three training methods (Offline, Online, and Visual) and two levels of experience (Novice, Experienced) were analyzed using GLM ANOVA, using SPSS v9.0 at a 0.05 level of significance.

Results

The testing and ANOVA analysis produced the following results:

Symbol	Condition	Novice	Experienced	All Conditions
OFF	OFFLINE	36.69	27.95	32.32
ONL	ONLINE	27.16	47.54	37.35
VIS	VISUAL	17.02	21.13	19.08
	All Subjects	**30.07**	**31.28**	**30.68**

GLM Results			**Significance**
Model	F = 52.09	p = .000	***
SKILL	F = .213	p = .645	
TRAIN	F=22.88	p= .000	***
SKILL * TRAIN	F = 3.87	p= .023	**

* = Significant at p < .1 ** = Significant at p< .05 ***= significant at p< .01
Note: The above mean values are test scores out of 70 maximum.

Figure 3. GLM Analysis of LEARNING: Means by Condition

Learning, or more specifically, retention, was a measure of how well subjects learned through studying the materials and also conducting the task. The learning score was determined by grading the POST-TEST QUIZ, which had maximum score of 70 (perfect score) and a minimum score of 0 (no responses).

When the effects of training are examined (see Figure 3), it was found that novices learned the most with the Offline manual (mean=36.69), and the least with visual training (mean=17.02). The novice required the detail inherent in, and the offline nature of the manual in order to learn best. Experienced users learned best with the online and the worst with visual online. Experienced users can easily grasp information from an online training source, and can more efficiently use the information found there and apply it, as opposed to the offline manual. Visual online training brings about poor learning for either skill level. Together with actual learning, it is also interesting to examine perceptions of learning: how much do subjects feel that they have learned while examining the materials and doing the task? Learning perception is measured using a learning perception scale incorporated into the Post-Test Questionnaire. The questionnaires and scales used in this research were generally consistent with those used in past NJIT research studies, which include both research projects and dissertation research (Benbunan, 1997; Fjermestad, 1994; Dufner, 1995; Ocker, 1995).

Symbol	Condition	Novice	Experienced	All Conditions
OFF	OFFLINE	6.08	5.77	6.31
ONL	ONLINE	4.67	5.06	4.77
VIS	VISUAL	4.62	6.12	5.54
	All Subjects	**5.13**	**5.95**	**5.54**

GLM Results			**Significance**
Model	F = 272.90	p = .000	***
SKILL	F = 17.68	p = .000	***
TRAIN	F=20.44	p= .000	***
SKILL * TRAIN	F = 6.31	p= .002	***

* = Significant at p < .1 ** = Significant at p< .05 ***= significant at p< .01
Note: Lower mean values above represent higher subjective rating of learning.

Figure 4. GLM Analysis of LEARNING PERCEPTION: Means by Condition

The results (see Figure 4) indicate that novices perceived to have learned more than experienced users overall. In terms of the impact of training method, online training had the highest learning perception, followed by visual training, and then offline training. Online training is comprehensive and provides a great deal of information. Its online nature also makes it more accessible to users and connects it together with the task rather being separate (as with the offline manual). The manual brought about the lowest learning perception, perhaps due to requiring more effort and time in order to use and learn from it properly. In fact, the offline manual, overall, produced the second highest learning, slightly below online.

One observation which can be made right away is that both novices and experienced could not appreciate (in terms of their perceptions) the benefits of offline training. While both novices and experienced subjects rated the offline training rather low in terms of learning perception (6.08 and 5.77, respectively) they actually learned a considerable amount in actual learning (36.69 for Novice/ Offline and 27.95 for Experienced/ Offline).

It yields some interesting insights to examine the relationship between the subjects' learning and learning perception. Do the training methods which a subject perceives as having learned from, actually bring about learning? This can be examined by taking novices and experienced users separately and considering the role of skill * train in particular. For novices, the perception was that both visual and online training brought about the best learning, with offline the worst. This is entirely contrary to actual learning, where novices learned the most from offline help! Visual online training actually resulted in the lowest learning, and online was roughly halfway between the two other types. Experienced subjects, on the other hand, had much better perceptions of their own learning. This is likely due to their knowledge and experience in programming and/or working with markups. Citing the same examples for novices, experienced perceived that visual help would result in the worst learning (6.12) , and that offline and online would be better (5.77 and 5.06, respectively). In examining actual learning, online was best (47.54), followed by offline

(27.95), and visual was definitely lowest at 21.13. So, it is clear that experienced subjects have a much better perception of their own learning.

Discussion and Conclusion

It has been said that perceptions do not always reflect reality. This cannot be more true when it comes to training and learning markup languages. The results indicate that the kinds of training most resisted by novice users are exactly those which would bring about the best learning, while those which were perceived to be extremely useful turned out not to be that useful after all.

Studies have reported that only about 14% of users actually read reference manuals (Penrose and Seiford, 1988). Novices in particular tend to resist manuals, and felt they learned more with the concise, "visual" form of training. However this form of training may not be the best for new users with limited computer and programming experience. In fact, offline training, while rated low in terms of system satisfaction and learning perception, actually brought about high levels of learning for novices. The fact that there is a completely inaccurate perception of learning on the part of novices may be part of the reluctance of novices to embrace the offline manual.

A well-designed, "minimal" manual would work out well for those who are first learning the system, even if they are experienced, since offline training seems to help learning over online forms of training. In terms of training, once a user becomes familiar with a particular system, online training may be substituted for the manual. The concise visual online training which has real examples together with the code, would be ideal for providing advanced users a reference to refresh one on a certain command or tag.

Manuals still have an important place, since some users prefer a written manual in place of online training. And, as reported in this study, some people, especially novices, with limited computer experience, would learn better with a printed manual instead of navigating through online training. In fact, the current practice of not including printed manuals and enclosing a CD-ROM or referring someone to web-based help may be detrimental to novices who are trying to learn the system for the first time.

The results clearly show that offline manuals, despite the less than perfect perceptions about them, are important tools for learning and using the language and the system. An important point to mention is that the format and organization of the manual is critical. Providing 10 volumes of primarily reference manuals, with page after page of commands listed one after the other, may not produce the best results in terms of learning, satisfaction, and performance than a well-designed single volume tutorial . Instead, using a minimalist approach which emphasizes real tasks and activities, and which are geared to the needs and interests of users, would work best. Reference volumes still have their place, however they should probably be separated from the main tutorial materials or be searchable from a CD-ROM, so that the concise "learning" and "use" materials can be made distinct from the reference resources. This will help to improve perceptions and satisfaction in terms of offline help.

Online training seems to be everywhere. Most new software packages have some kind of tutorial or online help facility, and online training/help is an important component of a software designer's arsenal, since some users (especially experienced) prefer it, and having it available online helps to improve accessibility and can include such advanced features as searching and indexing. However, "too much information" presented in an on-screen, online format may have a detrimental effect. Many users do not like to read lengthy explanations off a screen, and having it on the screen, partially or fully obscuring the main system, may prove to be distracting and increase "cognitive load." It might be useful to produce a kind of hybrid system which combines the best features of online and visual online. Someone can use a concise form when needed, and obtain more detailed information when desired.

In general, it is recommended that *online training*, if used the primary tutorial or learning facility, be directed towards experienced users, who can make use of its online nature and specific features. The need to navigate through screens and learn to use the system may be added complications which may be daunting to the novice. Online training, by its very nature, takes up space on the screen, and may be distracting from the task at hand.

From the results of this study, it appears that *visual online training* is useful in increasing satisfaction. However it does not contribute much in the way of promoting learning, especially for novices. As such, it seems best suited to experienced users, who already have a more solid foundation in programming, and can use such a concise form efficiently. As mentioned in the previous paragraph, it might be useful to consider if some kind of "hybrid" system that could be created, which combines both the visual and detailed forms of help so that it can serve a wide range of users. This might prove to have greater utility than using stand-alone visual online. While graphical/visual presentations maybe more attractive and therefore pleasing to the user, they may not compete effectively with the use of semantic memory as the main learning facility of the human mind. The key to effective learning may

still be the ability to trigger the right semantic metaphors in which the user may develop new understandings within the context of things they already understand. In addition, the poor performance of visual training may be due to the fact that it provides no conceptual structure for the learner to classify what they are using and to aid retention. Facts are remembered better when there is a conceptual map to file them in memory.

From the perspective of organizations and management, there are a number of recommendations and conclusions which can be made with regards to training. More specifically, they are:

- *Do not ignore the benefits of well-designed offline training materials*. Manuals, especially those using minimalist approaches, are useful for both novices and experienced users to really learn the language or software package.

- *Novices may not know what helps them to learn best, and may choose inappropriate training approaches.* It is far better to tailor a program for a learner based on their previous background and skill level. This is in contrast to the approach in some firms to provide the same type of training to all employees. While this approach may be highly beneficial to some, it provides only minimal help to others.

- *Encourage meaningful learning rather than rote learning*. Implement training programs and methods which produce meaningful learning (reflecting a fundamental understanding of concepts and applying them to new situations), as opposed to rote learning (memorizing information with no connection to previous knowledge). The ability to combine and integrate knowledge is key (Ausubel, 1986; Wertheimer, 1959).

- *Online and visual online-based training may be better suited to more experienced users*. Due to the complexities of the online form of training, as well as the visually -oriented format of visual online training, both of these seem better suited to learners with previous computing experience.

- *Learners of a new package or system could benefit more from printed rather than electronic manuals/training materials.* A lot of commercial software is being provided to learners with no real paper manuals. This is probably a significantly unproductive decision on the part of organizations to save some direct costs and which can result in significant indirect costs.

These recommendations and conclusions should be taken into account when designing or implementing a training course or program for computer-based skills and applications. If properly delivered, training could play an important role in effectively helping corporate staff and managers make use of the computing resources in their organization.

References

Argyris, C. Teaching Smart People How to Learn, *Harvard Business Review*, May-June 1991.

Association of American Publishers. Electronic Manuscript Project, Reference Manual on Electronic Manuscript Preparation and Markup, 1992.

Ausubel, D.P. *Educational Psychology:; A Cognitive View*. New York: HRW, 1968.

Benbunan, R., Effects Of Computer-Mediated Communication Systems On Learning, Performance And Satisfaction: A Comparison Of Groups And Individuals Solving Ethical Scenarios, Ph.D. dissertation, Rutgers University, 1997.

Black, J.B., Carroll, and McGuigan. What kind of minimal manual is most effective? *ACM CHI+GI Proceedings*, 1987, pp. 159-162.

Borgman, C. L. The User's Mental Model of an Information Retrieval System. *Intl. Journal of Man-Machine Studies*, 1986, 24, 1, 47-64.

Carroll, J.M. Minimalist Training. *Datamation*. 1984, 30,18, 125-136.

Carroll, J.M., *The Nurnberg Funnel: Designing Minamalist Instruction for Practical Computer Skill* (Cambridge MA: MIT Press, 1990).

Carroll, J.M. and Mazur, S.A. LisaLearning. *IEEE Computer*, 1986, 19, 11, 35-49.

Carroll, J.R., P.L. Smith-Kerkler, and S. Mazur-Rimetz, . The Minimal manual, *Human-Computer Interaction*, 3, 2, 1987. pp. 123-153.

Coombs, J.H., A.H. Renear, and S. J. Rose. Markup Systems and the Future of Scholarly Text Processing, *Communications of the ACM*, 30, 11, 1987, pp. 933-947.

Darnell, R., *HTML 4 Unleashed*, Indianapolis: Sams, 1997.

Davis, S. and R. Bostrom. Training End Users. *MIS Quarterly*, March 1993

Dufner, D., The Effects of Group Support (Listing and Voting Tools) and Sequential procedures on Group Decision Making, Using Asynchronous Computer Conferences, Doctoral Dissertation. Ph.D. in Management Program, Graduate School of Management, Rutgers University, Newark.

Fjermestad, J. Group Strategic Decision Making in a Computer-Mediated Communications Environment: A Comparison of Dialectical Inquiry and Constructive Consensus Approaches, Doctoral Dissertation. Ph.D. in Management Program, Graduate School of Management, Rutgers University, 1994, Newark.

Goldfarb, C., *The SGML Handbook*, New York: Oxford University Press, 1991

Holzner, S., *Xml Complete*, New York: Mc Graw-Hill, 1997

Hsu, Jeffrey and M. Turoff, A Markup Approach to Surveys and Questionnaires, *Journal of End User Computing*, Vol. 8 1996, No. 4, 20-27.

Lazonder, A.W. and H. van der Meij, The Minimal manual: Is Less Really More?, *International Journal of Man-Machine Studies*, 39, 1993, 729-752.

Leventhal, M. And D. Lewis, *Designing Xml Internet Applications* (Charles F. Goldfarb Series), Upper Saddle River: Prentice Hall, 1998.

Light, R., and T. Bray, *Presenting Xml,* Indianapolis: Sams, 1997.

Mack, R.L., C.H. Lewis, and J.M. Carroll, Learning to use word processors, *ACM Transactions Office Info Systems*, 1, 3, 1983, pp. 254-261.

Michard, A. Graphical Presentation of Boolean Expressions in a Database Query Language, *Behaviour and Information Technology*, 1, 3, 1982, 279-288.

Mullen, R. , *The Html 4 Programmer's Reference*, RTP: Ventana, 1998.

Ocker, R. Requirements Definition using a Distributed Asynchronous Group Support System: Experimental Results on Quality, Creativity and Satisfaction, Ph.D. dissertation, 1995, Rutgers University.

Olfman L. and R.P. Bostrom, The influence of training on use of end-user software. *Proceedings of the ACM Conference on Office Information Systems*, 1988, pp. 110-118.

Payne, S.J. and T. Green, The structure of command languages. *International Journal Man-Machine Studies*, 30, 213-234.

Penrose , J.M. and L. Seiford. Microcomputer users' preferences for software documentation: an analysis. *Journal of Technical Writing and Communication*, 18, 1988, 355-366.

Prewitt, E. . What Managers Should Know about How Adults Learn. *Harvard Management Update*. January 1997.

Ramsay, J.E. and K. Oakley, Designing minimalist tutorials from scratch, *Instructional Science*, 21 ,1992 , pp, 83-87.

Redish, J.C. Reading to learn to do. *The Technical Writing Teacher*, 15, 1988, pp. 223-233. Reiser, Anderson, and Farrell, 1985.

Rivard, L.M. and Huff, S.L. . Factors of Success for End-User Computing. *Communications of the ACM*, 1988, 31,5, 552-561.

Scharer, L. L. User training: less is more. *Datamation*, 29, 1983, pp. 175-182.

Sebrechts, M. and R.L. Marsh, Components of Computer Skill Acquisition: Some Reservations about Mental Models and Discovery Learning Designing and Using Human-Computer Interfaces and Knowledge Based Systems. *Proceedings of the Third International Conference on Human-Computer Interaction* , 1989, v.2 ,p.168-173.

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

Shih, Y., and Alessi, S.M., (1994), Mental models and transfer of learning in computer programming. *Journal of Research on Computing in Education*, 26(2), 1994, 154-7.

Shneiderman, B., *Designing the User Interface 3/E*, Reading MA: Addison Wesley.

Shneiderman, B., *Designing the User Interface 2/E*, Reading MA: Addison Wesley, 1992.

St. Laurent, S., *XML: A primer*, IDG Books, 1997

Wertheimer, M. *Productive Thinking*. New York: Harper and Row, 1959

White, C.E. and Christy, D.P. The Information Center Concept. *MIS Quarterly*, 11,4, 451-458.

Wright, H. , SGML Frees Information, *BYTE*, June 1992, pp. 263-268.